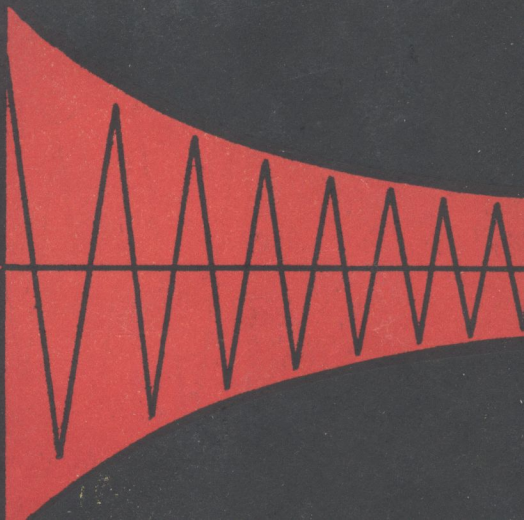


МЕТОДИ ОБЧИСЛЕНЬ

ПРАКТИКУМ
на ЕОМ



519.6/075)
MSY

МЕТОДИ ОБЧИСЛЕНЬ

ПРАКТИКУМ
на ЕОМ

2862-13

Затверджено
Міністерством освіти України
як навчальний посібник
для студентів вузів,
які навчаються
із спеціальності
«Прикладна математика»



Київ
«Вища школа»
1995

ББК 22.19я73

М54

УДК 518.5 (07)

Автори: *В. Л. Бурківська, С. О. Войцехівський, І. П. Гаврилюк, Л. Д. Греков, О. О. Клуннік, М. П. Копистира, А. В. Кузьмін, В. Л. Макаров, М. М. Москальков, А. І. Риженко*

Рецензенти: *Я. Й. Бігун, М. М. Дріль, В. Г. Маценко, І. М. Черевко* — кандидати фіз.-мат. наук доценти, *М. Ф. Кириченко* — д-р фіз.-мат. наук проф. (Чернівецький державний ун-т); *А. Ф. Кошій* — канд. фіз.-мат. наук доц. (Харківський державний ун-т); *В. Я. Скоробогатко* — д-р фіз.-мат. наук (Ін-т прикладних проблем механіки і математики НАН України)

Редакція технічної літератури

Редактор *В. Ф. Хміль*

Методи обчислень: Практикум на ЕОМ: Навч. посібник / М54 *В. Л. Бурківська, С. О. Войцехівський, І. П. Гаврилюк та ін.* — К.: Вища шк., 1995. — 303 с. : іл.

ISBN 5-11-004030-3

Розглянуто чисельні методи розв'язування основних задач обчислювальної математики. Подано методичні вказівки до використання цих методів, запитання для самоперевірки з метою кращого засвоєння теоретичного матеріалу, тексти підпрограм на алгоритмічній мові ФОРТРАН з прикладами використання їх, а також контрольні завдання.

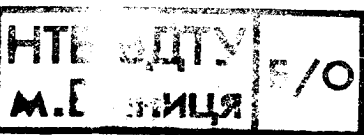
Для студентів вузів, які навчаються із спеціальності «Прикладна математика», а також аспірантів і наукових співробітників, що застосовують чисельні методи для розв'язування задач прикладної математики за допомогою електронних обчислювальних машин.

М $\frac{1602012000-144}{211-95}$ 113-95

ББК 22.19я73

ISBN 5-11-004030-3

© *В. Л. Бурківська, С. О. Войцехівський, І. П. Гаврилюк, Л. Д. Греков, О. О. Клуннік, М. П. Копистира, А. В. Кузьмін, В. Л. Макаров, М. М. Москальков, А. І. Риженко, 1995*



<i>Вступ</i>	4
§ 1. Графічне відображення інформації	6
§ 2. Розв'язування нелінійного рівняння	14
§ 3. Прямі методи розв'язування систем лінійних алгебраїчних рівнянь	23
§ 4. Розв'язування систем нелінійних рівнянь	34
§ 5. Задачі на власні значення	40
§ 6. Інтерполювання многочленами	48
§ 7. Інтерполяційні та згладжуючі сплайни	64
§ 8. Чисельне інтегрування	79
§ 9. Найкраще середньоквадратичне наближення	92
§ 10. Розв'язування задачі Коші для системи звичайних диференціальних рівнянь методом Рунге — Кутти	100
§ 11. Задача Коші для жорстких систем звичайних диференціальних рівнянь	110
§ 12. Метод стрільби розв'язування крайових задач для звичайних диференціальних рівнянь	125
§ 13. Методи Рітца та Бубнова — Гальоркіна розв'язування звичайних диференціальних рівнянь	142
§ 14. Метод сіток розв'язування крайових задач для звичайного диференціального рівняння другого порядку	149
§ 15. Метод скінченних елементів для розв'язування рівняння Пуассона	158
§ 16. Розв'язування одновимірного рівняння теплопровідності	169
§ 17. Економічні методи розв'язування двовимірного рівняння теплопровідності	180
§ 18. Різницева схема для розв'язування одновимірних рівнянь гіперболічного типу	187
<i>Додаток</i>	194
<i>Показчик розміщення підпрограм</i>	297
<i>Список використаної та рекомендованої літератури</i>	300
<i>Предметний показчик</i>	302

Спеціаліст з прикладної математики має володіти широким колом знань, щоб ставити та розв'язувати складні практичні задачі на електронних обчислювальних машинах (ЕОМ). Методика, яка при цьому застосовується, почала розроблятися з появою перших ЕОМ і зараз основні її моменти об'єднані в понятті обчислювального експерименту (ОЕ), який включає такі основні етапи:

1) вибір математичної моделі у вигляді алгебраїчних, диференціальних рівнянь, задачі мінімізації тощо;

2) створення на цій основі дискретної моделі, реалізація якої зводиться до виконання найпростіших арифметичних і логічних операцій на ЕОМ;

3) складання обчислювальної програми;

4) розрахунок за програмою на ЕОМ;

5) обробка й аналіз знайдених чисельних результатів.

При цьому на другому етапі найчастіше доводиться мати справу з такими стандартними задачами чисельного аналізу, як розв'язування систем лінійних і нелінійних рівнянь, мінімізація функцій, чисельні диференціювання й інтегрування та ін. Методи розв'язування цих задач розглядаються в університетському курсі «Методи обчислень». На третьому та п'ятому етапах використовуються готові програми, що реалізують методи розв'язування перелічених вище стандартних задач. Ці програми об'єднуються в пакети прикладних програм (ППП).

Практикум на ЕОМ призначений для засвоєння чисельних методів, які найчастіше застосовуються на практиці при розв'язуванні стандартних задач обчислювальної математики, вивчення особливостей програмної реалізації їх, розширення досвіду практичної роботи на сучасних ЕОМ. При цьому передбачається, що в загальному проектуванні на початкових курсах студенти в основному вивчили:

1) компоненти математичного забезпечення ЕОМ і практику використання їх, включаючи алгоритмічні мови та відповідні транслятори, редактори програмних текстів налагодження програми, засоби програмування;

2) методику налагодження програм;

3) основні прийоми введення інформації в ЕОМ і зручного для осмислення її виведення;

4) структуру та вигляд звітних документів.

Цей посібник відповідно до програми університетського курсу «Методи обчислень» містить лабораторні роботи з основних його розділів. З метою зменшення часу на розробку програми реалізації кожного методу, що вивчається, в книзі наводяться тексти підпрограм на алгоритмічній мові ФОРТРАН з відповідними коментарями. Це допоможе студентам не лише скоротити дефіцитний навчальний час на розробку програм, а й дасть змогу їм краще засвоїти основні особливості чисельних методів і специфіку програмної реалізації їх. За рекомендацією викладача ці підпрограми можна використати і цілком як «чорні ящики» або, з погляду педагогіки, як «виносну пам'ять».

Підпрограми частково запозичено (з певною переробкою) з відомих літературних джерел, а решту розроблено авторами. Добре відомо, що якість і ефективність програм залежать від загальних принципів, яким підпорядкована розробка їх (наприклад, таких, як мінімальний час роботи, мінімальні витрати ємності пам'яті, сумісність з іншими програмами тощо).

При розробці підпрограм автори виходили з принципу «ієрархичності», згідно з яким у кожній підпрограмі повинні максимально використовуватись інші. Цей принцип, по-перше, має переваги з методичної точки зору, а по-друге, скорочує програмні тексти, хоча в деяких випадках призводить до збільшення часу виконання програм.

Для зручності користування всі підпрограми розміщено в додатку, а ілюструючі програми з прикладами застосування підпрограм разом з результатами обчислень наведено у відповідних параграфах книги.

Порядок виконання завдань студентами такий:

1) вивчення того чи іншого методу розв'язування конкретної задачі, для чого в методичних вказівках до кожної теми зазначено необхідну літературу;

2) вивчення алгоритму та підпрограм, які реалізують методи обчислень;

3) оволодіння навичками використання підпрограм тестуванням їх і розробка власних програм;

4) розв'язування конкретного варіанта завдання на ЕОМ;

5) аналіз знайденого результату й оформлення звіту.

§ 1. Графічне відображення інформації

Ефективність використання ЕОМ як засобу проведення ОЕ при розв'язуванні різних прикладних задач залежить від можливості швидкої оцінки здобутої інформації. Це можна зробити, наприклад, за допомогою графічного відображення інформації.

Для перегляду графічної інформації можна застосовувати спеціальну апаратуру: пристрої для побудови графіків (плотери), графічні дисплеї. Використання такої апаратури потребує спеціального математичного забезпечення. Прикладом комплексу програм для виведення графічної інформації, орієнтованого на застосування алгоритмічної мови ФОРТРАН, може бути система ГРАФОР [4].

Проте велика кількість ЕОМ (СМ-4, ДВК-2М та ін.), які використовуються за нашого часу (в тому числі і в начальному процесі), не мають цих пристроїв. Стандартними пристроями відображення інформації для них є алфавітно-цифровий дисплейний пристрій (АЦДП) або принтер. Тому в цьому параграфі наведено досить прості програми, що дають змогу будувати графіки функцій однієї змінної та лінії рівня функції двох змінних на цих поширених пристроях виведення інформації.

Побудова графічного зображення на АЦДП складається з заповнення кожної позиції рядка, що виводиться, деяким символом, який відображує точку графіка, чи прогалиною. При цьому у зв'язку зі скінченністю поля зображення потрібно визначати мінімальне та максимальне значення таблично заданої функції, будувати координатні осі й наносити на них шкалу значень, обчислювати кроки відображення символів уздовж осей та ін.

Підпрограма GRAPH. Ця підпрограма дає змогу побудувати графік кількох (до дев'яти) функцій $y = f_i(x)$, $i = 1, M - 1$, заданих таблицею в декартовій системі координат. Значення всіх функцій мають бути заданими на одній сітці (краще рівномірній) незалежної змінної.

Координати сітки та значення функцій повинні розміщатися в одному масиві V розмірності $MN = M * N$, де M — кількість функцій плюс 1; N — число точок сітки вздовж осі абсцис. Всі ці значення заносяться в масив V групами з N елементів: спочатку — всі координати вузлів сітки, потім — усі значення першої функції (причому k -те значення функції відповідає k -му вузлу), далі — значення другої функції і т. д.

Опис інших вхідних параметрів підпрограми розкривається в коментарях і додаткових пояснень не потребує.

З'ясуємо тепер інформацію, що виводиться підпрограмою GRAPH. У першому її рядку зазначається символ, яким відображаються значення функції відповідного номера. Далі під заголовком «АБСЦИСИ» в стовпцях виводяться значення координат вузлів сітки. Праворуч зверху виводиться номер графіка. Потім у полі з 50 стовпців і N рядків відповідними символами відображаються значення функцій, причому якщо значення двох або більше функцій збігаються чи близькі одне до одного, то у відповідній точці зазначається символ останньої з цих функцій. Нижче виводяться поділки шкали по осі ординат, значення кроку STEPУ цієї шкали, мінімальне та максимальне значення всіх функцій, які виводяться, що дає змогу наближено знаходити значення функції для кожної точки її графіка. Крім того, в полі графіка символом «:» відображається нульове значення ординат (вісь абсцис), якщо виконується умова $V_{MIN} \times V_{MAX} < 0$.

Цю програму можна застосовувати, наприклад, для графічного відокремлення коренів рівняння $f_1(x) = f_2(x)$. Нехай значення функцій $f_1 = \cos x$, $f_2 = x^2$ обчислюються на рівномірній сітці з кроком $H = 0,04$ на відрізку $[0, 1]$, який має $N = 26$ точок уздовж осі абсцис. Програма, що реалізує цей приклад, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ДО ПІДПРОГРАМИ GRAPH: C
C ПОБУДОВА ГРАФІКІВ ДВОХ ФУНКЦІЙ ДЛЯ ВІДОКРЕМЛЕННЯ C
C КОРЕНІВ РІВНЯННЯ F1(X)=F2(X) C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DIMENSION V(100)
C ЗАДАВАННЯ ІНТЕРВАЛУ ЗМІНИ АБСЦИС
TYPE 100
ACCEPT *,A
TYPE 101
ACCEPT *,B
C ЗАДАВАННЯ КІЛЬКОСТІ ТОЧОК СІТКИ
TYPE 102
ACCEPT *,NS
C ЗАДАВАННЯ ПРИСТРОЮ ВИВЕДЕННЯ ГРАФІКА
TYPE 103
ACCEPT *,NW
H=(B-A)/FLOAT(NS-1)
C ОБЧИСЛЕННЯ ЗНАЧЕНЬ ФУНКЦІЙ ТА ФОРМУВАННЯ МАСИВУ V
X=A
DO 10 I=1,NS
V(I)=X
V(I+NS)=F1(X)
V(I+2*NS)=F2(X)
10 X=X+H
C ЗВЕРНЕННЯ ДО ПІДПРОГРАМИ ПОБУДОВИ ГРАФІКА
CALL GRAPH(NW,1,NS,3,3*NS,V)
STOP
100 FORMAT(1H$, 'A=')
```



```

101  FORMAT(1H$, 'B = ')
102  FORMAT(1H$, 'NS = ')
103  FORMAT(1H$, 'NW = ')
      END

```

```

C    FUNCTION F1(X)
      F1 = COS(X)
      RETURN
      END

```

```

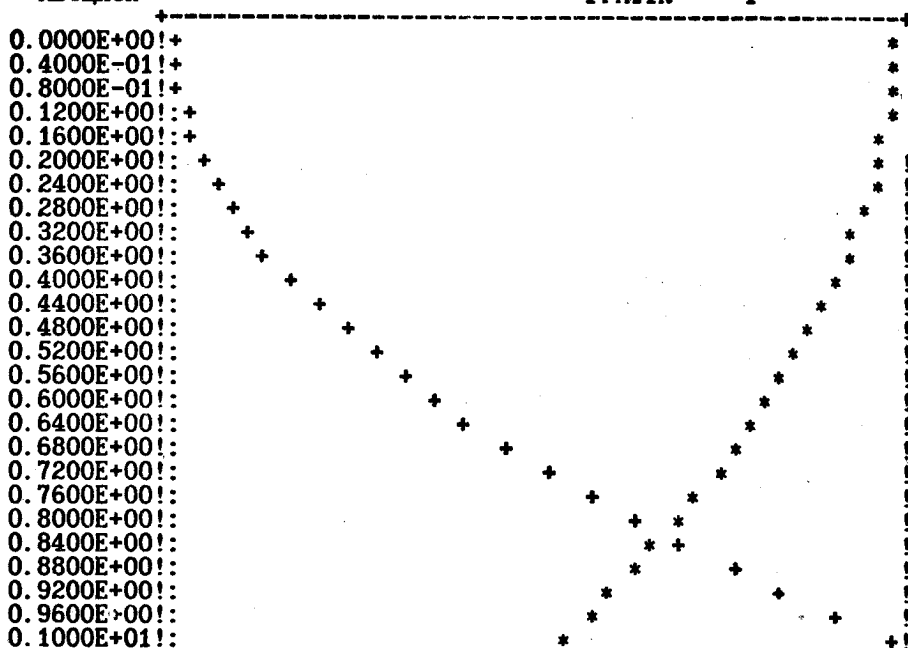
C    FUNCTION F2(X)
      F2 = X*X
      RETURN
      END

```

Після введення з терміналу значень $A = 0$, $B = 1$, $NS = 26$, $NW = 6$ на принтері дістаємо такий результат:

ФУНКЦІЇ: 1 - * 2 - +
 АБСЦИСИ

ГРАФІК 1



ШКАЛА ОРДИНАТ

1 2 3 4 5 6 7 8 9 10 11
 STEPУ = 0.10000E+00
 VMIN = 0.00000E+00 VMAX = 0.10000E+01

Як бачимо, корінь рівняння $\cos x = x^2$ знаходиться в інтервалі 0,8—0,84, а відповідне значення функцій становить приблизно 0,78.

Підпрограма GRAFIK. У разі неоднозначності однієї чи кількох функцій використання підпрограми GRAPH може призвести до ділення на нуль. Ця ситуація виникає, наприклад, при побудові графіків функцій, заданих параметрично, або при використанні полярної системи координат для обчислення абсцис і ординат функцій.

Для побудови графіків функцій, заданих довільним набором значень абсцис і ординат, можна застосувати підпрограму GRAFIK, яка дає змогу побудувати також графіки кількох функцій, заданих на різних нерівномірних сітках. На відміну від підпрограми GRAPH у підпрограмі GRAFIK вісь абсцис орієнтується горизонтально. Тому кількість символів NX для відображення графіка функції вздовж цієї осі не може перевищувати 70.

Кількість символів NY для відображення графіка функції вздовж осі ординат може бути довільною. Проте для відображення графіків функцій, що виводяться, без зміни масштабу NX і NY треба вибирати так, щоб відношення кроків $HX = (XMAX - XMIN)/NX$, $HY = (YMAX - YMIN)/NY$ було близьким до 0,65 для принтера та 0,43 для дисплея. Ці значення можуть бути уточнені ітераційним підбором NX і NY при введенні їх з терміналу, щоб коло на графіку відображалося без деформацій.

Підпрограма GRAFIK може бути використана для відокремлення коренів системи двох рівнянь. Так, якщо необхідно відокремити корені системи рівнянь

$$f_1(x, y) = 0; \quad f_2(x, y) = 0,$$

де перша функція описує коло радіусом 1 з центром у точці $x = 2$, $y = 0$, а друга задається рівнянням у полярній системі координат (кардіоида), то програма, що реалізує побудову графіків цих функцій, має вигляд:

```
С ГОЛОВНА ПРОГРАМА ДО ПІДПРОГРАМИ GRAFIK:
С ПОБУДОВА КАРДІОЇДИ ТА КОЛА
DIMENSION X(101),Y(101),N(5),H(5)
LOGICAL *1 SYM(2)
С ЗАДАВАННЯ НОМЕРА ПРИБОРУ ВИВЕДЕННЯ
TYPE 100
ACCEPT *,NW
С ЗАДАВАННЯ ПАРАМЕТРІВ ПІДПРОГРАМИ GRAFIK
NM=0
M=2
N(1)=31
N(2)=31
NX=47
NY=25
NM=N(1)+N(2)
С ОБЧИСЛЕННЯ КООРДИНАТ ТОЧОК КОЛА
```

HC=8.*ATAN(1.)/FLOAT(N(2)-1)

FI=0.

DO 10 I=1,N(1)

X(I)=2.+COS(FI)

Y(I)=SIN(FI)

FI=FI+HC

CONTINUE

10

C ОБЧИСЛЕННЯ КООРДИНАТ ТОЧОК КАРДІОЇДИ

FI=0.

DO 20 I=1,N(2)

J=I+N(1)

R=1.+COS(FI)

X(J)=R*COS(FI)

Y(J)=R*SIN(FI)

FI=FI+HC

20

C ПОВБУДОВА ГРАФІКА

CALL GRAFIK(NW,M,N,NM,X,Y,NX,NY)

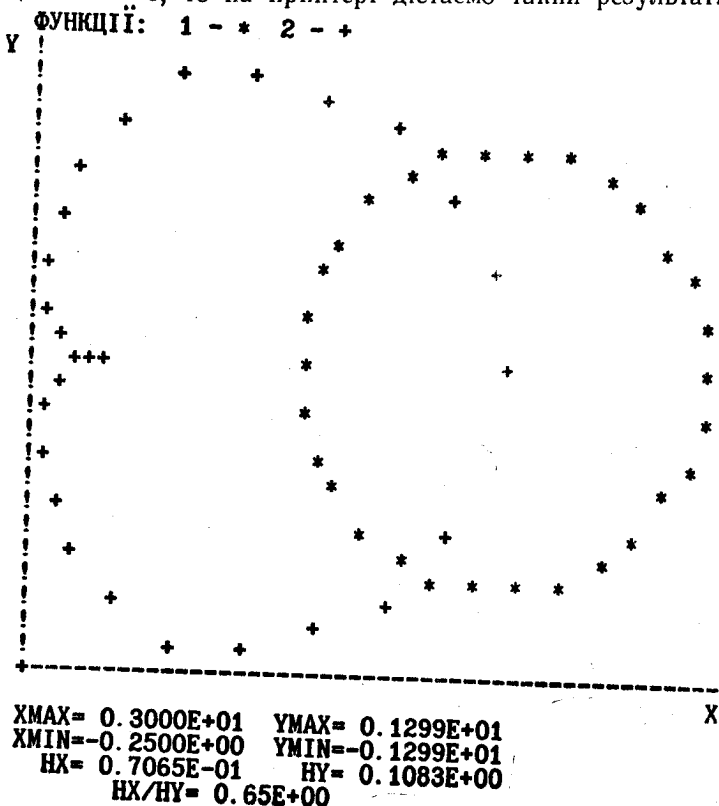
STOP

100

FORMAT(1H\$, 'NW=')

END

Якщо $NW = 6$, то на принтері дістаємо такий результат:



Зауважимо, що нанесені тут осі абсцис і ординат не дають інформації про місце знаходження початку координат. Цю й іншу інформацію можна дістати, використавши інформацію про максимальні та мінімальні значення абсцис і ординат функції Y_{MAX} , Y_{MIN} , X_{MAX} , X_{MIN} , а також кроки H_X , H_Y уздовж цих осей. Так, з наведеного графіка випливає, що наближеними розв'язками рівнянь є $x \approx 1,8$; $y \approx -0,8$ і $x \approx 1,8$; $y \approx 0,8$.

Підпрограма LINUR. Одним із способів графічного подання функції двох змінних є побудова ліній рівня, тобто ліній, що відповідають у межах зміни незалежних змінних однаковим значенням функції

$$f(x, y) = C.$$

Змінюючи значення C , дістаємо серію ліній, які характеризують поведінку функції. Такий спосіб зображення на площині тривимірного рельєфу застосовується в картографії.

Оскільки на АЦДП важко зобразити довільну лінію, фактично відображаються смуги рівня, тобто значення функції з деякого інтервалу.

При побудові ліній рівня функції двох змінних, заданих таблично, доводиться використовувати інтерполяцію. В підпрограмі LINUR для побудови лінії (смуги) рівня функції, заданої значеннями у вузлах нерівномірної прямокутної сітки, застосовується білінійна інтерполяція.

Зробимо кілька зауважень стосовно параметрів підпрограми LINUR. Як і в підпрограмі GRAFIK, її параметри NX і NY визначають кількість символів уздовж осей X і Y для побудови всього зображення функції. Природно, що вздовж осі X (горизонтальна вісь) $NX \leq 70$.

Для зміни кількості вузлів сітки NI вздовж осі X за допомогою терміналу в підпрограмі є параметр $NDIM$ рядкової розмірності двовимірного масиву V , що містить значення функції у вузлах сітки. Описавши цей масив у головній програмі з максимальною кількістю елементів за першим індексом, фактичну розмірність масиву за цим індексом можна вибрати меншою, тобто $NI \leq NDIM$.

Параметр EPS дає змогу регулювати ширину смуг рівня функції: при зменшенні його до нуля вони будуть «стоншуватися», а при збільшенні до максимального значення 0,5 на графіку одна смуга неперервно переходить в іншу. Можна рекомендувати вибирати значення EPS у межах 0,2—0,3.

Для відображення ліній (смуг) рівня функцій використовуються цифри від 0 до 9. Перед виведенням цих ліній (смуг) на пристрій з номером NW виводиться таблиця відповідності символів зображення та інтервалів значень функції, які виводяться. Проміжні значення функції відображаються точками.

Програма, що ілюструє побудову ліній рівня функції $f(x, y) = 1 + \sin^2 \pi x - \ln(1 + y^2)$ в квадраті $0 \leq x \leq 1$, $0 \leq y \leq 1$ при $NX = 60$, $NY = 35$, $N = 6$, $NI = 20$ і $NJ = 10$, має вигляд:

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  ГОЛОВНА ПРОГРАМА ДО ПІДПРОГРАМИ LINUR:                       C
C  ПОБУДОВА ЛІНІЙ РІВНЯ ФУНКЦІЇ Z=F(X,Y)                       C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DIMENSION X(101),Y(101),U(121,51)
C  ЗАДАВАННЯ ПАРАМЕТРІВ ПІДПРОГРАМИ LINUR
      NDIM=121
      TYPE 100
      ACCEPT *,NW
      NX=60
      NY=40
      NI=20
      NJ=15
      N=6
      EPS=.3
      HJ=1./FLOAT(NJ-1)
      HI=1./FLOAT(NI-1)
C  ОБЧИСЛЕННЯ ЗНАЧЕНЬ ФУНКЦІЇ
      Y(1)=0.
      DO 20 J=1,NJ
          X(1)=0.
          DO 10 I=1,NI
              U(I,J)=F(X(I),Y(J))
          10   X(I+1)=X(I)+HI
          20   Y(J+1)=Y(J)+HJ
C  ЗВЕРНЕННЯ ДО ПІДПРОГРАМИ ПОБУДОВИ ЛІНІЙ РІВНЯ
      CALL LINUR(NW,NX,NY,N,EPS,X,Y,NI,NJ,NDIM,U)
      STOP
      100  FORMAT(1H$'NW=')
      END
C
      FUNCTION F(X,Y)
      PI=8.*ATAN(1.)
      F=1.+SIN(PI*X)**2-ALOG(1.+Y*Y)
      RETURN
      END
```

Обчислення за цією програмою на комп'ютері ДБК-2М дають такий результат:

СИМВОЛ

ДІАПАЗОН ЗМІНИ ФУНКЦІЇ

0	0. 30685E+00	0. 40803E+00
1	0. 54294E+00	0. 74530E+00
2	0. 88020E+00	0. 10826E+01
3	0. 12175E+01	0. 14198E+01
4	0. 15547E+01	0. 17571E+01
5	0. 18920E+01	0. 19932E+01

```

-----
!0000000... 1111... 2222... 333333333333... 2222... 1111... 0000000!
!00000... 11111... 2222... 333333333333... 2222... 11111... 00000!
!0000... 1111... 222... 333333333333333... 222... 1111... 0000!
!00... 1111... 2222... 333333333333333... 2222... 1111... 00!
!..... 11111... 2222... 333333333333333333... 2222... 11111... !
!..... 11111... 2222... 333333... 333333... 2222... 11111... !
!..... 111111... 2222... 33333... 33333... 2222... 111111... !
!..... 111111... 2222... 33333... 33333... 2222... 111111... !
!.11111111... 2222... 33333... 33333... 2222... 11111111... !
!.11111111... 2222... 33333... 33333... 2222... 11111111... !
!.111111111... 2222... 33333... 4444... 33333... 2222... 111111111!
!.11111111... 2222... 3333... 44444444... 3333... 2222... 11111111!
!.111111... 2222... 3333... 444444444444... 3333... 2222... 111111!
!.11111... 2222... 3333... 44444444444444... 3333... 2222... 11111!
!.1111... 2222... 3333... 4444444444444444... 3333... 2222... 1111!
!.11... 2222... 3333... 444444444444444444... 3333... 2222... 11!
!.22222... 333... 444444444444444444... 333... 22222... !
!.2222... 3333... 444444... 444444... 3333... 2222... !
!.22222... 3333... 444444... 444444... 3333... 22222... !
!.222222... 3333... 44444... 44444... 3333... 222222... !
!.2222222... 3333... 4444... 4444... 3333... 22222222!
!.22222222... 3333... 44444... 5555... 44444... 3333... 222222222!
!.22222222... 3333... 4444... 5555... 4444... 3333... 22222222!
!.22222222... 3333... 4444... 555555... 4444... 3333... 22222222!
!.2222222... 3333... 4444... 5555555555... 4444... 3333... 2222222!
!.2222222... 3333... 4444... 5555555555... 4444... 3333... 2222222!
!.2222222... 3333... 4444... 5555555555... 4444... 3333... 2222222!
!.222222... 3333... 4444... 555555555555... 4444... 3333... 222222!
!.222222... 3333... 44444... 555555555555... 44444... 3333... 222222!
!.222222... 3333... 44444... 555555555555... 44444... 3333... 222222!
!.222222... 3333... 4444... 555555555555... 4444... 3333... 222222!
!.222222... 3333... 4444... 555555555555... 4444... 3333... 222222!
!222222... 3333... 4444... 555555555555... 4444... 3333... 222222!
-----

```

NX= 60 NY= 35 NI= 20 NJ= 10 EPS=0.30
 XINI= 0.00E-01 XEND= 1.00E+00
 YINI= 0.00E-01 YEND= 1.00E+00

Контрольні завдання

1. Графічно відокремити корені рівнянь:

а) $x^3 - 3x^2 + 5x - 2 = 0$; б) $x = (x^3 - 1)/3$; в) $x = \sin x + \ln x$.

2. Побудувати графік функції

$$\omega_n(x) = (x - x_0)(x - x_1) \dots (x - x_n)$$

для $x_i = -1 + 2i/n$ та $x_i = \cos((2i + 1)\pi/(2n + 1))$, $i = \overline{0, n}$ при $n = 5 \div 10$.

3. Побудувати графіки функцій, заданих у полярній системі координат:

а) $r = \sin 3\varphi$ (трипелюсткова троянда); б) $r^2 = \cos 2\varphi$ (лемніскаста Бернуллі);
в) $\varphi = 2 \sin r$.

4. Графічно визначити корені систем рівнянь:

а) $\begin{cases} (x-1)^2 + y^2 = 1; \\ y = 2x - 1; \end{cases}$ б) $\begin{cases} x = \cos y + 0,85; \\ y = 2x - 1; \end{cases}$

в) $\begin{cases} x = \sin y^2; \\ y = \sin x^2; \end{cases}$ г) $\begin{cases} x = \sin y; \\ y = 2x. \end{cases}$

5. Побудувати лінії рівня функцій двох змінних:

а) $f(x, y) = x^3 + y^3 - x^2 - 2xy - y^2$ ($-1 \leq x \leq 1$; $-1 \leq y \leq 1$);

б) $f(x, y) = \sin(xy)$ ($0 \leq x \leq a$; $0 \leq y \leq b$);

в) $f(x, y) = e^x \cos y$ ($0 \leq x \leq 2$; $-\pi/2 \leq y \leq \pi/2$).

6. Графічно визначити мінімум і максимум функцій:

а) $f(x, y) = x^2 - 2xy + 3y^2 - x + y - 1$ ($0 \leq x \leq 1$; $0 \leq y \leq 1$);

б) $f(x, y) = \sin x + \cos y + \cos(x - y)$ ($0 \leq x < \pi/2$; $0 \leq y < \pi/2$);

в) $f(x, y) = x - 2y + \ln(x^2 + y^2 + 1) + 3 \arctg(x + y)$
($-1 < x < 1$; $0 < y < 2$).

§ 2. Розв'язування нелінійного рівняння

Постановка задачі. Знайти з заданою точністю ε корені рівняння

$$f(x) = 0. \quad (1)$$

Вигляд функції $f(x)$ і додаткова інформація про шукані корені рівняння (1) визначаються контрольним завданням.

Методичні вказівки. Визначення коренів рівняння (1) зводиться до розв'язання таких задач:

- 1) дослідження кількості, характеру та розташування їх;
- 2) відокремлення коренів (визначення інтервалів, на яких знаходиться тільки один корінь);

3) знаходження коренів із заданою точністю ε [10, с. 138, 139].

У загальному випадку алгоритмів розв'язування перших двох задач не існує. Для відокремлення дійсних коренів рівняння корисно скласти таблицю значень функції $f(x)$. Зміна знаків у двох сусідніх рядках таблиці свідчить про наявність кореня (функція $f(x)$ вважається неперервною). Більш точний результат дає побудова графіка з використанням методів математичного аналізу*.

* Про способи дослідження нелінійного алгебраїчного рівняння див. А. Г. Курош «Курс высшей алгебры» (М.: Наука, 1971, — 432 с.).

▼ Найбільш простим і надійним методом розв'язування нелінійного рівняння є *метод ділення навпіл* (його називають *методом бісекції* або *дихотомії*) [1, с. 40, 41; 5, с. 194, 195; 10, с. 139—141]. Метод ґрунтується на теоремі Больцано — Коші, що формулюється так: якщо функція $f(x)$ є неперервною на відрізку $[a, b]$ і $f(a)f(b) < 0$, то існує така точка $c \in (a, b)$, що $f(c) = 0$.

Відповідно до цієї теореми, коли відомо, що на відрізку $[a, b]$ знаходиться один корінь рівняння (1), визначають середину відрізка — точку $x_0 = (a + b)/2$ й обчислюють значення функції $f(x_0)$. Якщо $f(x_0) \neq 0$, то з двох половин відрізка вибирають ту, на кінцях якої функція $f(x)$ набуває різних за знаком значень. Корінь рівняння (1) знаходиться на цьому відрізку.

Далі новий відрізок ділять навпіл і т. д. Якщо треба знайти корінь рівняння (1) з точністю ε , то ділення навпіл продовжують доти, поки довжина відрізка не стане меншою від 2ε . Тоді середина останнього відрізка і дасть значення кореня з потрібною точністю.

Недоліком методу ділення навпіл є невелика швидкість його збіжності. Для досягнення точності ε знаходження кореня нелінійного рівняння кількість ітерацій становить $\lceil \log_2 ((b - a)/\varepsilon) \rceil + 1$, де $\lceil X \rceil$ — ціла частина числа X . За одну ітерацію точність ε збільшується приблизно вдвічі.

▼ *Метод простої ітерації* [10, с. 141—143; 5, с. 176—183; 3, с. 407—411] (його називають також *методом послідовних наближень*) застосовують до нелінійного рівняння вигляду

$$x = \varphi(x). \quad (2)$$

Зазначимо, що (1) можна замінити еквівалентним йому рівнянням (2) багатьма способами, наприклад, поклавши $\varphi(x) = x + \psi(x)f(x)$, де $\psi(x)$ — довільна неперервна знакостала функція.

Задаючись деяким початковим наближенням x_0 , наступні наближення до кореня x_* знаходять за формулою

$$x_{n+1} = \varphi(x_n), \quad n = 0, 1, 2, \dots \quad (3)$$

Сформулюємо достатні умови збіжності методу простої ітерації.

ТЕОРЕМА. Нехай функція $\varphi(x)$ є визначеною та диференційовною на відрізку $[a, b]$, причому всі її значення $\varphi(x) \in [a, b]$. Тоді, якщо існує таке число q , що на цьому відрізку

$$|\varphi'(x)| \leq q < 1, \quad (4)$$

то послідовність (3) збігається до єдиного на відрізку $[a, b]$ розв'язку рівняння (2) для будь-якого початкового наближення $x_0 \in [a, b]$. При цьому, якщо на зазначеному відрізку похідна $\varphi'(x) > 0$,

$$|x_n - x_*| < \frac{q}{1 - q} |x_n - x_{n-1}|; \quad (5)$$

якщо ж похідна $\varphi'(x) < 0$, то

$$|x_n - x_*| < |x_n - x_{n-1}|. \quad (6)$$

Ці висновки узагальнюються на більш широкий клас функцій, які задовольняють умову Ліпшиця з константою, меншою від одиниці. Якщо ж умова (4) не виконується, то ітерації (3) можуть розбігатися.

На кожному кроці ітерацій обчислюють значення $y = \varphi(x_n)$ і, якщо $|y - x_n| \geq \varepsilon$, то, поклавши $x_n = y$, переходять до наступної ітерації. Якщо ж $|y - x_n| < \varepsilon$, то обчислення припиняють і як корінь приймають наближення x_n . Похибка знайденого результату залежить від знака похідної $\varphi'(x)$. При $\varphi'(x) > 0$ похибка визначення кореня становить $q\varepsilon/(1 - q)$; якщо ж $\varphi'(x) < 0$, то похибка не перевищує ε .

▼ *Метод Ньютона* [3, с. 411—416; 5, с. 189—193; 10, с. 143, 144] (його називають ще *методом дотичних* або *методом лінеаризації*) полягає в побудові ітераційної послідовності

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}, \quad n = 0, 1, 2, \dots \quad (7)$$

Цей метод можна розглядати як частковий випадок методу простої ітерації, якщо покласти $\varphi(x) = x - f(x)/f'(x)$. Оскільки $\varphi'(x) = f(x)f''(x)/(f'(x))^2$, з умови збіжності методу простої ітерації випливає, що метод Ньютона збігається, коли початкове наближення вибрано досить близьким до простого кореня x_* ($\varphi'(x_*) = 0$).

При довільному початковому наближенні ітерації збігаються, якщо скрізь у області, що розглядається, виконується нерівність

$$|f(x)f''(x)| < (f'(x))^2.$$

Наведемо ще одну достатню ознаку збіжності методу Ньютона.

ТЕОРЕМА. Нехай $f(x) \in C_2[a, b]$, $f(a)f(b) < 0$ і похідні $f'(x)$, $f''(x)$ зберігають знак на відрізку $[a, b]$. Тоді, якщо початкове наближення $x_0 \in [a, b]$ задовольняє нерівність

$$f(x_0)f''(x_0) > 0, \quad (8)$$

послідовність (7) збігається (причому монотонно) до єдиного на відрізку $[a, b]$ кореня x_* рівняння (1).

Щоб оцінити похибку x_n визначення цього кореня, можна скористатися нерівністю

$$|x_n - x_*| < \frac{M}{2m} |x_n - x_{n-1}|^2, \quad (9)$$

де

$$M = \max_{x \in [a, b]} |f''(x)|; \quad m = \min_{x \in [a, b]} |f'(x)|.$$

3 (9) впливає, що коли $|x_n - x_{n-1}| < \varepsilon$, маємо $|x_n - x_*| \leq \frac{M\varepsilon^2}{2m}$, тобто похибка наступного наближення приблизно дорівнює квадрату похибки попереднього наближення.

Таким чином, метод Ньютона досить ефективний, якщо відомо добре початкове наближення і в околі кореня рівняння (1) графік функції $f(x)$ має велику крутість. Інакше процес знаходження кореня може виявитись занадто довгим.

▼ Вивчимо вплив заокруглень у процесі обчислень на точність знайденого кореня рівняння [1, с. 471]. Для цього разом з рівнянням (1) розглянемо збурене рівняння

$$f(x) + \varepsilon q(x) = 0, \quad (10)$$

де $\varepsilon > 0$ — деякий малий параметр; $q(x)$ — деяка неперервна функція.

Якщо x_* — простий корінь рівняння (1), то з урахуванням теореми про існування функцій, обернених до аналітичної, корінь $x_*(\varepsilon)$ рівняння (10) можна розкласти в ряд за степенями ε :

$$x_*(\varepsilon) = x_* + \sum_{k=1}^{\infty} a_k \varepsilon^k. \quad (11)$$

Підставивши (11) у (10) і прирівнявши коефіцієнти при однакових степенях ε , дістанемо

$$a_1 = -q(x_*) [f'(x_*)]^{-1}. \quad (12)$$

Якщо x_* — корінь кратності $k > 1$, то відповідно до [1, с. 472] маємо

$$a_1 = \left[-\frac{k! q(x_*)}{f^{(k)}(x_*)} \right]^{1/k}. \quad (13)$$

Величина

$$v(x_*) = \left[\frac{k!}{f^{(k)}(x_*)} \right]^{1/k} \quad (14)$$

називається *числом обумовленості кореня x_** (кратності $k = 1, 2, 3, \dots$).

Очевидно, якщо число обумовленості кореня велике (погано обумовлене рівняння), то малі похибки вихідних даних призводять до великих похибок знайденого наближеного значення кореня.

Погано обумовленими будуть, наприклад, такі квадратні рівняння:

$$x^2 - 2,000001x + 1,000001 = 0$$

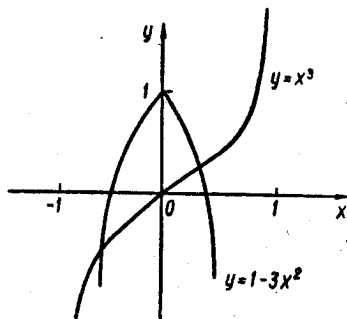


Рис. 1



(корені $x_1 = 1$; $x_2 = 1,000001$; $v(x_1) = -10^6$; $v(x_2) = 10^6$);

$$0,000001x^2 - 0,000003x + 0,000002 = 0$$

(корені $x_1 = 1$; $x_2 = 2$; $v(x_1) = -10^6$; $v(x_2) = 10^6$).

Приклади використання підпрограм. Чисельну реалізацію на ЕОМ алгоритмів наведених вище методів розглянемо на прикладі такої задачі: потрібно знайти додатний корінь рівняння $x^3 + 3x^2 - 1 = 0$ з точністю $\varepsilon = 0,0005$.

З побудови графіків функцій $y = x^3$ і $y = 1 - 3x^2$ (рис. 1) випливає, що шуканий корінь знаходиться на відрізку $[0, 1]$. Зазначаємо, що відокремити корінь можна і за допомогою підпрограми GRAPH (див. додаток).

▼ Головна програма, яка містить звернення до підпрограми BISECT, та підпрограма-функція, що реалізують для розглядуваного прикладу метод ділення навпіл, мають вигляд:

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ РІВНЯННЯ                               C
C МЕТОДОМ ДІЛЕННЯ НАВПІЛ                                             C
●CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
EXTERNAL F
CALL BISECT(0.,1.,.5E-03,F,X,K)
FX=F(X0)
TYPE 101
ACCEPT 102,NW
WRITE(NW,100)X,K,FX
STOP
```

```
100 FORMAT('КОРІНЬ=',E13.5/' КІЛ.ІТЕР.=',15/
*      'НЕВ'ЯЗКА=',E13.5)
101 FORMAT(1H$, 'NWRITE=')
102 FORMAT(11)
END
```

```
C ПІДПРОГРАМА-ФУНКЦІЯ
FUNCTION F(X)
F=X**3.+3.*X*X-1.
RETURN
END
```

Обчислення за цією програмою на комп'ютері ДБК-2М дають такий результат:

```
КОРІНЬ = 0,53174E + 00
КІЛ.ІТЕР. = 10
НЕВ'ЯЗКА = -0.10000E + 01
```

▼ Для застосування методу простої ітерації розглядуване рівняння запишемо у вигляді

$$x = \sqrt{\frac{1-x^3}{3}} = \varphi(x).$$

Уточнимо межі його додатного кореня: $x_* \in [0, 2/3]$.

Оскільки $\varphi'(x) = -\sqrt{3}x^2/2\sqrt{1-x^3}$, можна взяти

$$q = \max_{x \in [0, 2/3]} |\varphi'(x)| = |\varphi'(2/3)| \approx 0,265.$$

Отже, умова збіжності методу простої ітерації виконується. Початковим наближенням кореня вважатимемо $x_0 = 0$.

Головна програма, яка містить звернення до підпрограми SITER, та підпрограма-функція, що реалізують для розглядуваного прикладу метод простої ітерації, мають вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ НЕЛІНІЙНОГО C
C РІВНЯННЯ МЕТОДОМ ПРОСТОЇ ІТЕРАЦІЇ C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL F
X0=0.
Q=-0.265
CALL SITER(X0,0.5E-03,K,F,Q)
FX=X0-F(X0)
TYPE 101
ACCEPT 102,NW
WRITE(NW,100)X0,K,FX
STOP
100 * FORMAT('КОРІНЬ=' ,E13.5) KІЛ.ІТЕР.=',15/
      'НЕВ'ЯЗКА=' ,E13.5)
101 FORMAT(1H$, ' NWRITE='')
102 FORMAT(1I)
END

```

```

C ПІДПРОГРАМА-ФУНКЦІЯ
FUNCTION F(X)
F=((1.-X**3./3.)**0.5.
RETURN
END

```

Зазначимо, що допоміжна величина Q , яка є в підпрограмі SITER, визначається виразом $Q = q \operatorname{sign} \varphi'(x)$.

Обчислення за наведеною програмою на комп'ютері ДВК-2М дають такий результат:

```

КОРІНЬ= 0.53202E+00
КІЛ. ІТЕР.= 6
НЕВ'ЯЗКА= -0.82254E-04

```

▼ Для застосування методу Ньютона до розглядуваного прикладу знайдемо похідні $f'(x) = 3x^2 + 6x$ і $f''(x) = 6x + 6$, а додатний корінь рівняння шукатимемо в межах $x_* \in [0,5; 1]$. Похідні $f'(x)$, $f''(x)$ на цьому відрізку знака не міняють. За початкове наближення візьмемо значення $x_0 = 1$, яке задовольняє умову (8).

Зазначимо, що початкове наближення $x_0 = 0$ цю умову не задовольняє. Для розглядуваного прикладу

$$m = \min_{x \in [0,5; 1]} |3x^2 + 6x| = 3,75; \quad M = \max_{x \in [0,5; 1]} |6x + 6| = 12.$$

Головна програма, яка містить звернення до підпрограми NEWTON і підпрограма-функцій, що реалізують для розглядуваного прикладу метод Ньютона, мають вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ НЕЛІНІЙНОГО C
C РІВНЯННЯ МЕТОДОМ НЬЮТОНА C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL F,F1
X0=1.
EPS=SQRT(3.7E*0.5E-4/6.)
CALL NEWTON(X0,100,EPS,1H,I,F,F1)

```

```

FX=F(X0)
TYPE 101
ACCEPT 102,NW
IF(IH.EQ.0) WRITE(NW,100)X0,I,FX
STOP
100 FORMAT('КОРІНЬ X=',E15.7/' КІЛ.ІТЕР. K=',I4/
* 'НЕВ'ЯЗКА FX=',E15.7)
101 FORMAT(IH$, 'NWRITE=')
102 FORMAT(I1)
END

```

С ПІДПРОГРАМИ-ФУНКЦІЇ

```

FUNCTION F(X)
F=X**3.+3.*X*X-1.
RETURN
END

```

```

FUNCTION F1(X)
F1=3.*X*X+6.*X
RETURN
END

```

Обчислення за наведеною програмою на ДВК-2М дають такий результат:

```

КОРІНЬ X= 0.5320890E+00
КІЛ.ІТЕР. K= 4
НЕВ'ЯЗКА FX= 0.4768372E-06

```

▼ Одним з кращих машинних алгоритмів, які використовуються за нашого часу для визначення нуля функції, є підпрограма ZEROIN, що поєднує безвідмовність методу бісекції зі швидкістю методу січних [26, с. 117—183]. Звернення до цієї підпрограми має вигляд

```
CALL ZEROIN (AX, BX, F, TOL, Z),
```

де TOL — задана похибка визначення кореня.

Підпрограма забезпечує виконання ітераційного процесу, в якому на кожному кроці присутні абсциси A, B, C. У всіх випадках абсциси B і C обмежують нуль функції, причому $|F(B)| \leq |F(C)|$. Крім параметра TOL, у перевірці збіжності наближено застосовується параметр машинної точності EPS для того, щоб підстрахувати можливий випадок, коли задане значення TOL є досить малим.

На кожному кроці вибирається чергове наближення з двох можливих — одне дістаємо за методом бісекції, а інше — за алгоритмом інтерполяції. Якщо абсциси A, B і C різні, то виконується обернена квадратична інтерполяція, в іншому випадку — лінійна за методом січних.

Алгоритм ZEROIN завжди збігається, причому кількість обчислень значень функції не може перевищувати числа, що визначається виразом

$$\log_2^2 \left(\frac{B-A}{TOL1} \right),$$

де $TOL1 = 0,5 * TOL + 2,0 * EPS * ABS(B)$.

Головна програма, яка містить звернення до підпрограми ZEROIN, що реалізує розглядуваний приклад, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ РІВНЯННЯ С
С ЗА ДОПОМОГОЮ ПІДПРОГРАМИ ZEROIN С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL F
CALL ZEROIN(0.,1.,F,5E-05,Z)

```

TYPE 101
ACCEPT 102,NW
WRITE (NW,100)Z
STOP

100 FORMAT(2X,'КОРІНЬ=',E13.5)
101 FORMAT(1H\$, 'NWRITE=')
102 FORMAT(1I)
END

Підпрограму-функцію F наведено вище. Обчислення на комп'ютері ДВК-2М дають такий результат:

КОРІНЬ= 0.53209E+00

Запитання для самоперевірки

1. В чому полягає метод ділення навпіл? Які його переваги та недоліки? 2. Як обчислюється наближення за методом простої ітерації? Як записуються умови збіжності методу? 3. В чому полягає метод Ньютона? Які достатні ознаки його збіжності? 4. Чому метод Ньютона називають ще методом дотичних? 5. Який з методів розв'язування нелінійного рівняння збігається швидше і чому? 6. Який вигляд має алгоритм, що реалізується підпрограмою ZEROIN? 7. В чому полягає геометрична інтерпретація розглянутих вище методів розв'язування нелінійного рівняння?

Контрольні завдання

1. Виконати за вказівкою викладача одне з наведених у табл. 1 завдань за методами ділення навпіл, простої ітерації, Ньютона та підпрограмою ZEROIN і порівняти знайдені результати. В усіх варіантах завдань параметр $\varepsilon = 0,0001$.

Відповіді: $x = 1,0000$ (варіанти 1—5); $\tilde{x} = 2,0000$ (варіанти 6—10); $\tilde{x} = 3,0000$ (варіанти 11—15); $\tilde{x} = 4,0000$ (варіанти 16—20); $\tilde{x} = 5,0000$ (варіанти 21—25).

2. Оцінити, яке з рівнянь $x = \sqrt{1 - 3x^2}$ чи $x = \sqrt{\frac{1-x^3}{3}}$, еквівалентних рівнянню $x^3 + 3x^2 - 1 = 0$, краще підходить для розв'язування методом простої ітерації при $x_0 = 0; 0,25; 0,5; 0,75; 1$.

3. Резервуар для нафти завдовжки l (м) має форму лежачого циліндра радіусом r . На якій висоті h знаходиться рівень нафти в резервуарі, якщо він заповнений на q ($0 < q < 1$) своєї місткості?

Вказівка. На рис. 2 зображено поперечний розріз резервуара. Залежність об'єму нафти від кута $AOB = \alpha$ ($0 \leq \alpha \leq 2\pi$) визначається виразом $V = \frac{r^2 l}{2} (\alpha - \sin \alpha)$.

З умови задачі випливає, що α є розв'язком рівняння

$$\alpha - \sin \alpha - 2\pi q = 0. \quad (15)$$

Цей розв'язок можна знайти за одним із розглянутих вище методів. Тоді

$$h = r \left(1 - \cos \frac{\alpha}{2} \right).$$

Значення радіуса r циліндра (1—10 м) і параметра $q = 0,05n$ ($n = \overline{1, 20}$) задають викладачем. Довжина l резервуара на висоту рівня нафти не впливає.

4. Вивести ітераційну формулу для знаходження кореня нелінійного рівняння методом січних, замінивши у відповідній формулі методу Ньютона похідну $f'(x)$ роз-

Таблиця 1

Номер варіанта	Завдання	a	b	c
1	Знайти корінь рівняння $e^{ax} + bx + c = 0$	0,9100	-10,8975	8,4132
2		0,9300	-11,2798	8,7453
3		0,9500	-11,6058	9,0201
4		0,9700	-12,0382	9,4003
5		0,9900	-12,5370	9,8463
6	Знайти додатний корінь рівняння $ax^3 + bx^2 + c = 0$	0,9801	10,0804	-48,1624
7		1,0002	10,5432	-50,1744
8		1,1103	10,6431	-51,4548
9		1,2105	10,7503	-52,6852
10		1,3417	11,0002	-54,7544
11	Знайти найменший додатний корінь рівняння $a \sin \frac{\pi}{3} x + bx + c = 0$	0,2437	4,8203	-14,4609
12		0,3783	5,0027	-15,0081
13		0,4398	5,1132	-15,3396
14		0,5647	5,4117	-16,2351
15		0,6789	5,7203	-17,1609
16	Знайти найбільший корінь рівняння $\ln \frac{ax}{4} + bx + c = 0$	3,0200	1,0021	-5,1134
17		3,2600	1,0341	-5,3182
18		3,3700	1,2107	-6,0577
19		3,4800	1,3405	-6,6090
20		3,5600	1,5230	-7,3618
21	Знайти найменший додатний корінь рівняння $a \cos \frac{\pi x}{10} + bx + c = 0$	0,7487	1,0012	-5,0060
22		0,7823	1,3132	-6,5615
23		0,8154	1,4534	-7,2670
24		0,8911	1,5817	-7,9085
25		0,9427	1,6146	-8,0730

діленою різницею $\frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}$. Скласти підпрограму розв'язування нелінійного рівняння методом січних.

5. Визначити критичну силу для стержня, один кінець якого затиснено, а інший може переміщатися у вертикальному напрямку.

Вказівка. Критична сила P , при якій стержень втрачає стійкість, визначається рівнянням

$$\operatorname{tg} \sqrt{\frac{P}{EI}} L = \sqrt{\frac{P}{EI}} L,$$

де EI — жорсткість згину стержня; L — його довжина.

Якщо позначити $x = \sqrt{\frac{P}{EI}} L$, то задача зводиться до відшукування мінімального додатного кореня рівняння $\operatorname{tg} x - x = 0$.

6. При хімічній реакції $\text{CO} + \frac{1}{2} \text{O}_2 \rightleftharpoons \text{CO}_2$ процентний вміст x дисоційованого моля CO_2 визначається рівнянням

$$\left(\frac{p}{k^2} - 1\right)x^3 + 3x - 2 = 0,$$

де p — тиск, а k — константа рівноваги, що залежить від температури середовища.

Знайти x , якщо k і p мають такі значення:

k	1,365	1,412	1,533	1,647	1,718
p	2,0	1,8	1,6	1,4	1,2

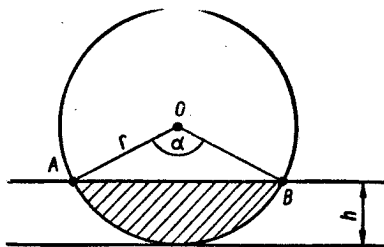


Рис. 2

§ 3. Прямі методи розв'язування систем лінійних алгебраїчних рівнянь

Постановка задачі. Розв'язати систему лінійних алгебраїчних рівнянь (СЛАР)

$$Ax = \vec{b},$$

де $A = (a_{ij})$ — квадратна матриця порядку n ; $\vec{b} = (b_i)$ — вектор-стовпець правої частини рівнянь системи.

Оцінити точність знайденого розв'язку. Матриця A , вектор \vec{b} та метод розв'язування визначаються контрольним завданням.

Методичні вказівки. Для розв'язування СЛАР (1) з довільною невідродженою матрицею A найчастіше застосовують метод послідовного виключення невідомих — *метод Гаусса*.

Зупинимося на деяких моментах цього методу, який ґрунтується на розкладі (факторизації) матриці A системи (1) [26, с. 43]. Власне цей варіант методу Гаусса реалізовано в підпрограмі DECOMP для розкладу матриці та в підпрограмі SOLVE для обчислення розв'язку СЛАР за допомогою знайденого розкладу.

Нехай $a_{11} \neq 0$. Визначимо $m_{i1} = -a_{i1}/a_{11}$, $i = 2, 3, \dots, n$. Додоживши перший рядок матриці A на m_{i1} і додавши його до i -го рядка, дістанемо матрицю $A^{(1)}$, усі елементи якої в першому стовпці, за винятком першого, дорівнюють нулю. Цей процес можна записати у вигляді

$$A^{(1)} = M_1 A,$$

де

$$M_1 = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{21} & 1 & 0 & \dots & 0 \\ m_{31} & 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ m_{n1} & 0 & 0 & \dots & 1 \end{bmatrix}.$$

Аналогічно робимо з другим, третім та іншими стовпцями матриці A , поки всі піддіагональні елементи її не стануть нульовими. В результаті матимемо матрицю

$$U = MA = A^{(n-1)} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22}^{(1)} & a_{22}^{(1)} & \dots & a_{2n}^{(1)} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & a_{nn}^{(n-1)} \end{pmatrix}, \quad (2)$$

яка є верхньою трикутною.

Очевидно, матриця перетворень

$$M = M_1 M_2 \dots M_{n-1} = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ m_{21} & 1 & 0 & \dots & 0 \\ m_{31} & m_{32} & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ m_{n1} & m_{n2} & m_{n3} & \dots & 1 \end{bmatrix}$$

є нижньою трикутною.

Елементи матриць M_k і $A^{(k)}$ ($k = 1, 2, \dots, n-1$) обчислюються за формулами

$$\begin{aligned} m_{ik} &= -a_{ik}^{(k-1)} / a_{kk}^{(k-1)}, \quad i = k+1, \dots, n; \\ a_{ij}^{(k)} &= a_{ij}^{(k-1)} + m_{ik} a_{kj}^{(k-1)}, \quad i = k+1, \dots, n, \end{aligned} \quad (3)$$

де $a_{ij}^{(0)} = a_{ij}$.

Елементи матриці U визначаються так:

$$u_{ij} = a_{ij}^{(j-1)}, \quad i = 1, 2, \dots, n; \quad j = i, i+1, \dots, n; \quad u_{ij} = 0, \quad i > j. \quad (4)$$

Щоб уникнути ділення на нульовий або малий ведучий елемент $a_{kk}^{(k-1)}$, в методі Гауса використовують схеми з вибором ведучого елемента. Для цього в матриці $A^{(k)}$ роблять перестановку рядків. На $(k-1)$ -му кроці серед усіх елементів k -го стовпця матриці $A^{(k-1)}$, що лежать нижче її головної діагоналі, вибирають максимальний за модулем. Нехай це буде елемент $a_{mn}^{(k-1)}$. Тоді переставляють k -й та m -й рядки матриці A і до знайденої матриці застосовують описаний вище процес виключення k -ї змінної.

У результаті дістають матрицю $U = MA$, де матриця перетворень M тепер має вигляд

$$M = M_{n-1}P_{n-1} \dots M_1P_1.$$

Тут P_k — матриці перестановок, кожна з яких є матрицею, знайденою з одиничної матриці тією самою перестановкою рядків, що й матриця A . Наприклад, після перестановки k -го та m -го рядків ($k < m$) матриці P_k матимемо

$$P_k = \begin{pmatrix} 1 & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ 0 & 1 & \dots & & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & \dots & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & \dots & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 0 & \dots & 0 & \dots & 1 \end{pmatrix} \begin{matrix} k\text{-й рядок} \\ \\ \\ m\text{-й рядок} \\ \\ \\ \end{matrix} \quad (5)$$

$k\text{-й} \quad m\text{-й}$
стовпець стовпець

Після того, як знайдено трикутний розклад матриці A , легко знайти розв'язок СЛАР (1). Дійсно, спочатку визначаємо вектор-стовпець $\vec{c} = M\vec{b}$, а потім розв'язуємо СЛАР

$$U\vec{x} = \vec{c}$$

з верхньою трикутною матрицею U . Формули, що реалізують цей розв'язок, мають вигляд

$$x_n = c_n/u_{nn}; \quad x_i = \left(c_i - \sum_{k=1+i}^n u_{ik}x_k \right) / u_{ii}, \quad i = n-1, \dots, 1. \quad (6)$$

Метод Гаусса застосовують для розв'язування СЛАР з щільно заповненою матрицею. Його реалізація потребує n^2 комірок пам'яті ЕОМ та $2/3n^3 + 0(n^2)$ арифметичних операцій. Зручність використання цього методу у формі трикутного розкладу полягає в тому, що при розв'язуванні кількох СЛАР з однаковою матрицею A та різними правими частинами \vec{b} економляться арифметичні операції на зведення системи до трикутного вигляду, оскільки воно проводиться тільки один раз для всіх СЛАР зразу.

Потреба в розв'язуванні кількох СЛАР з однаковою матрицею виникає, наприклад, у разі необхідності обчислення оберненої матриці A^{-1} . Її елементи можуть бути знайдені як компоненти векторів $\vec{\alpha}_j = = (\alpha_{ij})$, які є розв'язками СЛАР

$$A\vec{\alpha}_j = \vec{e}_j, \quad j = 1, 2, \dots, n,$$

де $\vec{e}_j = (\delta_{ij})$, а δ_{ij} — символ Кронекера.

Зауважимо, що за допомогою трикутного розкладу можна обчислити визначник матриці A , тобто

$$\det A = \det U = (-1)^l \prod_{i=1}^r u_{ii} = (-1)^r a_{11} a_{22}^{(1)} \dots a_{nn}^{(n-1)}, \quad (7)$$

де l — кількість перестановок рядків.

▼ При розв'язуванні СЛАР з симетричною матрицею доцільно використовувати метод квадратних коренів, реалізація якого потребує вдвічі меншої кількості арифметичних операцій $-1/3 n^3 + 0(n^2)$, а також меншої кількості комірок пам'яті для збереження цієї матриці $-n(n+1)/2$.

Як і метод Гаусса, метод квадратних коренів можна розглядати як процедуру розкладання матриці на трикутні множники [20, с. 69]:

$$A = S^T D S, \quad (8)$$

де S — верхня трикутна матриця; S^T — нижня трикутна транспонована матриця; D — діагональна матриця з елементами $d_{ii} = \pm 1$. Елементи матриць S і D визначаються виразами

$$d_{11} = \text{sign } a_{11}, \quad s_{11} = \sqrt{|a_{11}|}, \quad s_{1j} = a_{1j}/(s_{11}d_{11}), \quad j = 2, 3, \dots, n;$$

$$d_{ii} = \text{sign } p_i, \quad s_{ii} = \sqrt{|p_i|}, \quad p_i = a_{ii} - \sum_{k=1}^{i-1} |s_{ki}|^2 d_{kk};$$

$$s_{ij} = \left(a_{ij} - \sum_{k=1}^{i-1} s_{ki} s_{kj} d_{kk} \right) / (s_{ii} d_{ii}),$$

$$i = 2, 3, \dots, n, \quad j = i+1, i+2, \dots, n.$$

Після того як знайдено розклад (8), \vec{x} як розв'язок СЛАР з трикутними матрицями, тобто

$$S^T D \vec{y} = \vec{b}; \quad S \vec{x} = \vec{y}, \quad (9)$$

має вигляд

$$y_1 = b_1 / (s_{11} d_{11}), \quad y_i = \left(b_i - \sum_{k=1}^{i-1} s_{ki} d_{kk} y_k \right) / (s_{ii} d_{ii}), \quad i = 2, 3, \dots, n; \quad (10)$$

$$x_n = y_n / s_{nn}, \quad x_i = \left(y_i - \sum_{k=i+1}^n s_{ik} x_k \right) / s_{ii}, \quad i = n-1, \dots, 1.$$

За допомогою трикутного розкладу симетричної матриці можуть бути обчислені обернена матриця A^{-1} та визначник матриці A .

▼ Оскільки арифметичні операції на ЕОМ виконуються з округ-

леннями, замість точного розв'язку \vec{x} дістають наближений розв'язок \vec{x}_* , точність якого характеризується оцінкою

$$\|\vec{x} - \vec{x}_*\| \|\vec{x}_*\| < \nu(A) \beta^{1-t}, \quad (11)$$

де β — основа системи числення ЕОМ; t — кількість розрядів мантиси чисел з плаваючою комою; $\nu(A)$ — число обумовленості матриці A , яке можна визначити за формулою [26, с. 59]

$$\nu(A) = \|A\| \|A^{-1}\|,$$

де $\|A\|$ — одна з норм матриці (наприклад, норма $\|A\|_1 = \max_i \left(\sum_{j=1}^n |a_{ij}| \right)$), узгоджена з нормою вектора $\|\vec{x}\|_1 = \max_i |x_i|$.

З оцінки (11) випливає, що відносна похибка знайденого розв'язку, обумовлена похибками заокруглення, може досягати значного значення (більш як 100 %, що означає відсутність хоча б однієї вірної цифри в наближеному розв'язку), якщо $\nu(A) > \beta^{t-1}$. Для обчислювальних машин серій ЕС (наприклад, ЕС-1040), СМ, ДВК-2М це дає критичне значення числа обумовленості $\nu(A) = 10^7$.

Якщо точність завдання вихідних даних (коефіцієнтів матриці A та правих частин \vec{b}) менша від точності, обумовленої розрядною сіткою ЕОМ, то і при менших значеннях $\nu(A)$ СЛАР будуть погано обумовлені. Для розв'язування таких систем розроблено спеціальні алгоритми.

Використання підпрограм. Описаний вище метод Гауса реалізовано в підпрограмах DECOMP і SOLVE. Перша виконує розкладання матриці A на трикутні множники, а друга знаходить розв'язок рівняння (1) за знайденим розкладом. Підпрограма DECOMP дає також оцінку COND числа обумовленості $\nu(A)$, причому ця оцінка відповідає нижній границі числа обумовленості, тобто

$$\frac{\nu(A)}{n} \leq \text{COND} \leq \nu(A).$$

Параметри підпрограм DECOMP і SOLVE описано в коментарях до них. Зробимо зауваження щодо особливостей застосування цих підпрограм і їх параметрів.

Перший параметр NDIM підпрограми DECOMP дає змогу знайти розклад матриці різної розмірності. Для цього в головній програмі масив, де зберігається матриця A , повинен бути описаний з максимальною розмірністю, а параметр NDIM мати значення цієї розмірності.

Параметру COND після виконання підпрограми DECOMP надається значення оцінки числа обумовленості. Якщо якийсь ведучий елемент дорівнює нулю (тобто за програмою DECOMP встановлено, що

матриця є виродженою), то цьому параметру надається нульове значення, завдяки чому застосовувати підпрограму SOLVE в цьому випадку немає рації.

Масив IPVТ містить номери ведучих елементів на відповідному кроці. Останній елемент цього масиву дорівнює $(-1)^l$, де l — кількість перестановок рядків, яка може бути використана для обчислення визначника матриці A за формулою (7).

Масив WORK призначений для обчислення оцінки числа обумовленості. Його треба описати в головній програмі з розмірністю N .

Слід мати на увазі, що після звернення до підпрограми DECOMP значення елементів матриці A не зберігаються. Тому, якщо вони ще потрібні для інших обчислень, їх треба сховати в якийсь масив. Це саме стосується й значень масиву B , в якому при зверненні до підпрограми SOLVE повинні знаходитися значення правих частин \vec{b} .

▼ Метод квадратних коренів реалізовано в підпрограмах FACTSQ та RESQ. Зробимо зауваження щодо особливостей застосування їх.

Матриця A , яка є симетричною, для економії пам'яті EOM зберігається в одновимірному масиві розмірністю $n(n+1)/2$ по стовпцях, тобто

$$A(1) = a_{11}, \quad A(2) = a_{12}, \quad A(3) = a_{22}, \quad \dots, \quad A(i+j(j-1)/2) = a_{ij}, \quad \dots, \quad A(n(n+1)/2) = a_{nn}.$$

Параметр COND і масив WORK мають такі самі призначення, що й у підпрограмі DECOMP. Дійсним залишається також зауваження щодо вхідних значень масивів A та B .

▼ Приклад розв'язування СЛАР з контрольного завдання 1. Головна програма для визначення розв'язку цієї системи при $n = 5$, $\alpha = 10^{-5}$, $p = 1,5$, $q = 2,5$ за допомогою методу Гаусса має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ДО ТЕМИ C
C «ПРЯМІ МЕТОДИ РОЗВ'ЯЗУВАННЯ СЛАР». C
C ВИПАДОК ДОВІЛЬНОЇ МАТРИЦІ C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DIMENSION A(10,10),B(10),IPVT(10),WORK(10)
      TYPE 100
C ЗАДАЄМО НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ РОЗВ'ЯЗКУ
      АССЕРТ *,NW
      TYPE 101
C ЗАДАЄМО РОЗМІРНІСТЬ СИСТЕМИ N<=10
      АССЕРТ *,N
C РЯДКОВА РОЗМІРНІСТЬ NDIM=10
      NDIM=10
C ВВЕДЕННЯ З ТЕРМІНАЛА ПАРАМЕТРІВ ЗАДАЧІ
      TYPE 102
      АССЕРТ *,AL,P,Q
C ВИВЕДЕННЯ ПАРАМЕТРІВ ЗАДАЧІ
      WRITE(NW,103) N,AL,P,Q

```

```

C ЗВЕРНЕННЯ ДО ПІДПРОГРАМИ ОБЧИСЛЕННЯ МАТРИЦІ А
C ТА ВЕКТОРА В
  CALL DAN(NDIM,N,AL,P,Q,A,B)
C ВИВЕДЕННЯ МАТРИЦІ А ТА ВЕКТОРА В
  WRITE(NW,104)
  DO 10 I=1,N
    WRITE(NW,105) (A(I,J),J=1,N)
  10 CONTINUE
  WRITE(NW,106) (B(I),I=1,N)
C ТРИКУТНЕ РОЗКЛАДАННЯ МАТРИЦІ А
  CALL DECOMP(NDIM,N,A,COND1,IPVT,WORK)
C ПЕРЕВІРКА МАТРИЦІ А НА ВИРОДЖЕНІСТЬ
  IF(COND1-1.) 20,30,30
  20 WRITE(NW,107)
  STOP
  30 CONTINUE
C РОЗВ'ЯЗУВАННЯ СЛАП
  CALL SOLVE(NDIM,N,A,B,IPVT)
C ВИВЕДЕННЯ РОЗВ'ЯЗКУ ТА ОЦІНКИ ЧИСЛА ОБУМОВЛЕНОСТІ
  WRITE(NW,108) (B(I),I=1,N)
  WRITE(NW,109) COND1
  STOP
100 FORMAT(1H$, ' NW=' )
101 FORMAT(1H$, ' N=' )
102 FORMAT(1H$, ' AL,P,Q=' )
103 FORMAT(/2X, ' N=' ,I1, ' AL=' ,E12.5, ' P=' ,E12.5,
*      ' Q=' ,E12.5)
104 FORMAT(/6X, 'МАТРИЦЯ А')
105 FORMAT(2X,4E12.4)
106 FORMAT(/6X, 'ВЕКТОР В'/2X,4E12.4))
107 FORMAT(6X, 'МАТРИЦЯ ВИРОДЖЕНА')
108 FORMAT(/6X, 'РОЗВ'ЯЗОК X'/(2X,3E15.7))
109 FORMAT(/6X, 'ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ=' ,
*      E15.7)
  END

```

Підпрограма обчислення матриці А та вектор-стовпця \vec{b} варіанта завдання має вигляд:

```

SUBROUTINE DAN(NDIM,N,AL,P,Q,A,B)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ПІДПРОГРАМА ОБЧИСЛЕННЯ ВАРІАНТА ЗАВДАННЯ.
C ВХІДНІ ПАРАМЕТРИ:
C .....
C NDIM — РЯДКОВА РОЗМІРНІСТЬ МАТРИЦІ,
C ОГОЛОШЕНА В ГОЛОВНІЙ ПРОГРАМІ;
C N — ПОРЯДОК МАТРИЦІ;
C AL,P,Q — ПАРАМЕТРИ ВАРІАНТА ЗАВДАННЯ.
C ВИХІДНІ ПАРАМЕТРИ:
C .....
C A — МАТРИЦЯ СИСТЕМИ;
C B — ВЕКТОР ПРАВИХ ЧАСТИН.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DIMENSION A(NDIM,N),B(N)
C=1

```

```

R=1.
NM1=N-1
DO 10 K=1,NM1
  R=R*P*Q
  C=C+R
  Z=(AL-I.)/C
  RP=1.
  DO 30 I=1,N
    RQ=1.
    DO 20 J=1,N
      A(I,J)=Z*RP*RQ
      IF(I.EQ.J) A(I,J)=A(I,J)+1.
      RQ=RQ*Q
    RP=RP*P
  DO 50 I=1,N
    S=0.
  DO 40 K=1,N
    S=S+A(I,K)*(K-1.)
  B(I)=S
RETURN
END

```

Після виконання програми, якщо параметри $NW = 6$, $N = 5$, $AL = 1E - 5$, $P = 1,5$, $Q = 2,5$, дістаємо такий результат:
 $N=5$ $AL= 0.10000E-04$ $P= 0.15000E+01$ $Q= 0.25000E+01$

МАТРИЦЯ А

0.9963E+00	-0.9283E-02	-0.2321E-01	-0.5802E-01
-0.1451E+00			
-0.5570E-02	0.9861E+00	-0.3481E-01	-0.8703E-01
-0.2176E+00			
-0.8355E-02	-0.2089E-01	0.9478E+00	-0.1305E+00
-0.3264E+00			
-0.1253E-01	-0.3133E-01	-0.7833E-01	0.8042E+00
-0.4895E+00			
-0.1880E-01	-0.4700E-01	-0.1175E+00	-0.2937E+00
0.2657+00			

ВЕКТОР В

-0.8100E+00	-0.2149E+00	0.1776E+00	0.2664E+00
-0.1004E+00			

РОЗВ'ЯЗОК X

0.6548305E-03	0.1000982E+01	0.2001473E+01
0.3002210E+01	0.4003315E+01	

ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ= 0.2013034E+06

Оцінимо відносну похибку визначення розв'язку СЛАР. Вважаючи, що вона обумовлена похибкою заокруглень при обчисленні правих частин \vec{b} , маємо

$$\delta(\vec{x}^*) < 2 \cdot 10^5 \cdot 10^{-7} = 0,02.$$

Це значення відповідає реальній відносній похибці

$$\delta(\vec{x}^*) = \frac{\|\vec{x} - \vec{x}^*\|_1}{\|\vec{x}^*\|_2} = \frac{6 \cdot 548 \cdot 10^{-4}}{4,003} = 1,657 \cdot 10^{-4}.$$

▼ Приклад розв'язування симетричної СЛАР за допомогою методу квадратних коренів. Нехай маємо таку СЛАР четвертого порядку з матрицею Гільберта:

$$A = (a_{ij}), a_{ij} = 1/(i + j - 1); \quad \vec{b} = (b_i), b_i = 1/(4 + i), \quad i, j = 1, 2, 3, 4.$$

Компонентами розв'язку цієї системи є такі коефіцієнти многочлена третього степеня $P_3(t) = x_1 + x_2t + x_3t^2 + x_4t^3$ — найкращого середнього квадратичного наближення до функції $f(t) = t^4$ на відрізку $[0, 1]$: $x_1 = -1/70 = -0,0142857$; $x_2 = 2/7 = 0,285714$; $x_3 = -9/7 = -1,285714$; $x_4 = 2$.

Головна програма для визначення розв'язку розглядуваної СЛАР за допомогою підпрограм FACTSQ та RESQ має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ДО ТЕМИ C
C «ПРЯМІ МЕТОДИ РОЗВ'ЯЗУВАННЯ СЛАР». C
C ВИПАДОК СИМЕТРИЧНОЇ МАТРИЦІ C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DIMENSION A(10),B(4),D(4),WORK(4)
TYPE 100
C ЗАДАЄМО НОМЕР ПРИБОРУ ВИВЕДЕННЯ РОЗВ'ЯЗКУ
ACCEPT *,NW
C ФОРМУВАННЯ МАСИВІВ ДЛЯ ЗБЕРЕЖЕННЯ ВЕРХНЬОГО
C ТРИКУТНИКА МАТРИЦІ А ТА ВЕКТОРА ПРАВИХ ЧАСТИН В
N=4
DO 10 I=1,N
B(I)=1./FLOAT(4+I)
DO 10 J=I,N
K=I+J*(J-1)/2
A(K)=1./FLOAT(I+J-1)
10 CONTINUE
C ТРИКУТНЕ РОЗКЛАДАННЯ МАТРИЦІ А
CALL FACTSQ(N,A,D,COND,WORK)
C ПЕРЕВІРКА МАТРИЦІ А НА ВИРОДЖЕНІСТЬ
IF(COND-1.) 20,30,30
20 WRITE(NW,101)
STOP
30 CONTINUE
C РОЗВ'ЯЗУВАННЯ СЛАР
CALL RESQ(N,A,D,B)
C ВИВЕДЕННЯ РОЗВ'ЯЗКУ ТА ОЦІНКИ ЧИСЛА ОБУМОВЛЕНОСТІ
WRITE(NW,102) (B(I),I=1,N)
WRITE(NW,103) COND
STOP
100 FORMAT(1H$, ' NW=')
101 FORMAT(/6X, 'МАТРИЦЯ ВИРОДЖЕНА')
102 FORMAT(/6X, 'РОЗВ'ЯЗОК X'/(2X,4E12.4))
103 FORMAT(/6X, 'ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ=',
* E15.7)
END

```

Обчислення за цією програмою дають такий результат:
РОЗВ'ЯЗОК X

-0.1428389E-01 0.2856948E+00 -0.1285668E+01
0.1999970E+01

ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ= 0.3185403E+05

Відносна похибка визначення розв'язку

$$\delta(\vec{x}^*) \leq 0,32 \cdot 10^{-2}$$

Запитання для самоперевірки

1. В чому полягає основна ідея методу Гаусса (методу квадратних коренів)?
2. Для чого робляться перестановки рядків у методі Гаусса?
3. Коли вигідніше застосовувати метод квадратних коренів замість методу Гаусса й чому?
4. Які причини появи похибок при розв'язуванні СЛАР прямими методами?
5. Що характеризує число обумовленості матриці?
6. В якому випадку не можна довіряти знайденому розв'язку СЛАР?
7. Як програмним шляхом можна збільшити точність розв'язку СЛАР, похибка якого обумовлена заокругленням обчислень на ЕОМ?

Контрольні завдання

1. Розв'язати СЛАР $\vec{A}\vec{x} = \vec{b}$, де $A = (a_{ij})$, $\vec{b} = (b_i)$; $a_{ij} = \delta_{ij} + zp^{i-1}q^{j-1}$, $z = (\alpha - 1)/c$, $c = \sum_{k=1}^n p^{k-1}q^{k-1}$, δ_{ij} — символ Кронекера, $b_i = \sum_{k=1}^n (k-1) a_{ik}$.

Матриця $A = A(\alpha)$ залежить від параметрів α , p та q , маючи такі властивості: число обумовленості $\nu(A)$ мінімальне при $\alpha = 1$ і дорівнює 1; при $\alpha \rightarrow 0$ обумовленість матриці A погіршується (якщо $\alpha = 0$, то матриця A стає виродженою); при $\alpha \rightarrow \infty$ число обумовленості $\nu(A)$ також зростає (якщо $q = p$, то матриця A стає симетричною).

Розв'язок такої СЛАР має вигляд $\vec{x} = (0, 1, \dots, n-1)$. Можна рекомендувати такі значення її параметрів: $\alpha = 10^{-2} \div 10^{-5}$ або $\alpha = 10^2 \div 10^6$; $p = 1 \div 10$; $q = 1 \div 10$ (для методу квадратних коренів $q = p$).

Більш складне завдання зводиться, наприклад, до обчислення не тільки розв'язку, а й визначника матриці A та оберненої матриці A^{-1} .

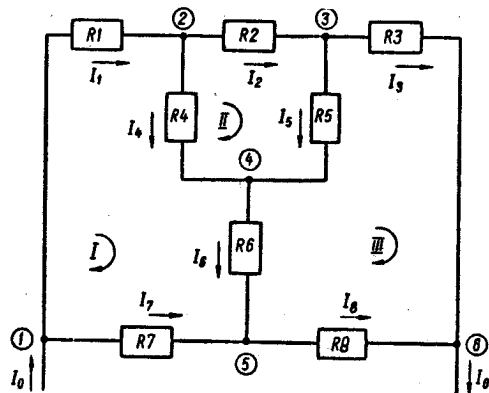


Рис. 3

2. Вважаючи, що коефіцієнти многочлена найкращого середньоквадратичного наближення $Q_{n-1}(t) = c_0 + c_1 t + \dots + c_{n-1} t^{n-1}$ для функції на відрізку $[0, 1]$ є розв'язком СЛАР (1), де

$$\vec{b} = (c_0, c_1, \dots, c_{n-1}); \quad a_{ij} = a_{ji} = \int_0^1 t^{i+j-2} dt = 1/(i+j-1);$$

$$b_i = \int_0^1 f(t) t^{i-1} dt; \quad i, j = 1, 2, \dots, n,$$

вадаються різними функціями та степенями многочлена $Q_{n-1}(t)$, дістаємо кілька варіантів завдання. Наприклад, можна вибрати функцію $f(t) = d_0 + d_1 t + \dots + d_n t^n$ з різними коефіцієнтами d_i , де $i = 0, 1, \dots, n$, а $d_n \neq 0$.

Число обумовленості матриці A (матриці Гільберта) зі збільшенням n швидко зростає. Тому треба обмежитися значеннями $n = 3 \div 6$.

3. В електричному колі, що складається з резисторів опором R_i , струми I_i в його вітках обчислюються за допомогою двох законів Кірхгофа [11, с. 85]:

1) правила вузла, згідно з яким у кожному вузлі кола виконується співвідношення $\sum_k I_k = 0$, де k відповідає множині індексів усіх струмів, які проходять крізь

вузол (струми, що входять, вважаються додатними, а ті, що виходять, — від'ємними);

2) правила петлі, згідно з яким у кожному замкненому контурі (петлі) виконуються співвідношення $\sum_k U_k = \sum_k I_k R_k = 0$, де k відповідає множині індексів оп-

орів резисторів, які входять у петлю. Якщо обхід контура здійснюється в деякому фіксованому напрямі, то струми в цьому напрямі вважаються додатними, а в зворотному — від'ємними.

Для містка Томсона (рис. 3) значення невідомих струмів I_1, I_2, \dots, I_8 , які визначаються струмом I_0 та опором резисторів R_1, R_2, \dots, R_8 , задовольняють СЛАР

$$I_1 + I_7 = I_0 \quad (\text{вузол } I);$$

$$I_1 + I_2 - I_4 = 0 \quad (\text{вузол } 2);$$

$$I_2 - I_3 - I_5 = 0 \quad (\text{вузол } 3);$$

$$I_4 + I_5 - I_8 = 0 \quad (\text{вузол } 4);$$

$$I_6 + I_7 - I_8 = 0 \quad (\text{вузол } 5);$$

$$I_1 R_1 + I_4 R_4 - I_6 R_6 - I_7 R_7 = 0 \quad (\text{петля } I);$$

$$I_2 R_2 + I_5 R_5 - I_4 R_4 = 0 \quad (\text{петля } II);$$

$$I_3 R_3 - I_5 R_5 - I_6 R_6 - I_8 R_8 = 0 \quad (\text{петля } III).$$

Рівняння $I_3 + I_8 = I_0$ для вузла 6 є лінійно залежним від рівнянь для вузлів 1—5.

Знайти значення струмів у колі при значеннях струмів I_i (в амперах) та опорів R_i (в омах), $i = 1, 2, \dots, 8$, наведених у табл. 2.

Таблиця 2

Номер варіанта	I_0	R_1	R_2	R_3	R_4	R_5	R_6	R_7	R_8
1	6	100	20	100	10	10			
2	6	100	20	50	10	10	0,1	5	5
3	6	10	20	100	5	20	2	30	30
4	6	10	10	10	10	10	30	5	30
5	6	5	2,5	4,5	0,5	0,05	10	10	10
6	10	1,5	2,5	3,5	4,5	5,5	50	4	2,4
7	10	10	9	8	7	6	6,5	7,5	8,5
8	10	3	2	3	2	6	5	3	4
9	10	4	2,1	3	2	3	2	3	2
10	10	7	4	5,4	6,7	8,4	3,2	4,1	5,8
				8	4	5	2	3	1

§ 4. Розв'язування систем нелінійних рівнянь

Постановка задачі. Розв'язати з заданою точністю ϵ систему n нелінійних рівнянь з n невідомими

$$\begin{cases} f_1(x_1, x_2, \dots, x_n) = 0; \\ f_2(x_1, x_2, \dots, x_n) = 0; \\ \dots \dots \dots \dots \dots \dots \dots \\ f_n(x_1, x_2, \dots, x_n) = 0. \end{cases} \quad (1)$$

Систему (1) у векторній формі можна записати так:

$$\vec{f}(\vec{x}) = \vec{0}. \quad (2)$$

Вектор-функція \vec{f} і деякі додаткові дані визначаються контрольним завданням.

Методичні вказівки. Системи нелінійних рівнянь розв'язують практично лише ітераційними методами. Розглянути в § 2 методи простої ітерації та Ньютона розв'язування нелінійного рівняння узагальнюються й на системи таких рівнянь.

Для застосування методу простої ітерації [2, с. 319—328; 10, с. 150—152] систему рівнянь (2) зведемо до еквівалентного вигляду

$$\vec{x} = \vec{\varphi}(\vec{x}). \quad (3)$$

Виберемо деяке початкове наближення розв'язку $\vec{x}^{(0)} = (x_1^{(0)}, x_2^{(0)}, \dots, x_n^{(0)})$. У випадку двох змінних його можна знайти графічно або за допомогою підпрограми GRAFIK. Для більшої кількості змінних у загальному випадку задовільних способів вибору початкового наближення розв'язку системи рівнянь (3) не існує.

Вибравши початкове наближення, наступні наближення до розв'язку системи рівнянь у векторній формі знайдемо за формулами

$$\vec{x}^{(k+1)} = \vec{\varphi}(\vec{x}^{(k)}), \quad k = 0, 1, 2, \dots, \quad (4)$$

або в скалярній формі

$$x_s^{(k+1)} = \varphi_s(x_1^{(k)}, x_2^{(k)}, \dots, x_n^{(k)}), \quad s = \overline{1, n}. \quad (5)$$

Достатню умову збіжності методу простої ітерації можна записати у вигляді

$$\|M\| \leq 1, \quad (6)$$

де M — матриця з елементами $M_{ij} = \max \left| \frac{\partial \varphi_i}{\partial x_j} \right|$.

Функцію $\vec{\varphi}(\vec{x})$ можна вибрати, наприклад, у вигляді $\vec{\varphi}(\vec{x}) = \vec{x} - t\vec{f}(\vec{x})$, а додатний параметр t підібрати так, щоб виконувалась умова (6).

▼ Метод Ньютона ґрунтується на лінеаризації системи рівнянь (2) [3, с. 410—412]. Нехай вектор $\vec{x}^* = (x_1^*, x_2^*, \dots, x_n^*)$ є її точним розв'язком і відомо досить добре його початкове наближення $\vec{x}^{(0)}$. Вважаючи, що задана функція двічі неперервно диференційовна, за формулою Тейлора маємо

$$0 \equiv f_i(\vec{x}^*) = f_i(\vec{x}^{(0)}) + \sum_{k=1}^n \frac{\partial f_i}{\partial x_k}(\vec{x}^{(0)}) (x_k - x_k^{(0)}) + r_i. \quad (7)$$

Опускаючи в (7) члени r_i вищого порядку мализни, дістаємо таку обчислювальну процедуру методу Ньютона:

$$\vec{x}^{(k+1)} = \vec{x}^{(k)} + \vec{\varepsilon}^{(k+1)}; \quad (8)$$

$$\left[\frac{\partial f}{\partial x} \right]_{x=\vec{x}^{(k)}} \vec{\varepsilon}^{(k+1)} = -\vec{f}(\vec{x}^{(k)}), \quad (9)$$

де $\left[\frac{\partial f}{\partial x} \right]$ — матриця Якобі, що визначається виразом

$$\left[\frac{\partial f}{\partial x} \right] = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_n} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_n} \\ \dots & \dots & \dots & \dots \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix}. \quad (10)$$

Критерієм закінчення ітераційного процесу (8) може бути умова $\|\varepsilon^{(k+1)}\| < \varepsilon$, де ε — задане значення похибки. Якщо ітераційний процес Ньютона збігається, то швидкість його збіжності буде квадратичною, про що свідчить така теорема.

ТЕОРЕМА. Нехай $\Omega_a = \{\vec{x} : \|\vec{x} - \vec{x}^*\| < a\}$ і при деяких додатних числах a, a_1, a_2 виконуються умови:

$$a) \|\vec{f}'(\vec{x})^{-1}\| \leq a_1, \quad \forall \vec{x} \in \Omega_a;$$

$$б) \|\vec{f}'(\vec{x}_1) - \vec{f}'(\vec{x}_2) - \vec{f}'(\vec{x}_2)(\vec{x}_2 - \vec{x}_1)\| \leq a_2 \|\vec{x}_2 - \vec{x}_1\|^2, \quad \forall \vec{x}_1, \vec{x}_2 \in \Omega_a.$$

Тоді при довільному векторі $\vec{x}_0 \in \Omega_b$ ітераційний процес Ньютона збігається і дійсною є така оцінка похибки:

$$\|\vec{x}^{(n)} - \vec{x}^*\| \leq c^{-1} (c \|\vec{x}^{(0)} - \vec{x}^*\|)^{2^n},$$

де $c = a_1 a_2$; $b = \min(a, c^{-1})$.

У випадку, коли ітераційний процес розбігається, треба більш ретельно підібрати вектор початкового наближення розв'язку $\vec{x}^{(0)}$.

Оскільки при використанні методу Ньютона необхідно розв'язувати СЛАР (9), добра чи погана обумовленість вихідної системи нелінійних рівнянь залежить від обумовленості (9) (див. про обумовленість СЛАР у відповідному параграфі цього посібника). Розв'язування системи нелінійних рівнянь (1) стає некоректною задачею, якщо матриця Якобі (10) погано обумовлена.

Приклади використання підпрограм. Приклад 1. Чисельну реалізацію на ЕОМ методу простої ітерації розглянемо на прикладі такої задачі: з точністю $\varepsilon = 10^{-4}$ потрібно знайти розв'язок системи рівнянь

$$\begin{cases} x = \cos y + 0,85; \\ y = \sin x - 1,32. \end{cases}$$

Виходячи з геометричних міркувань, вибираємо початкове наближення розв'язку $x^{(0)} = 1,85$; $y^{(0)} = -0,32$.

Головна програма, яка містить звернення до підпрограми SITERS, та підпрограма-функція, що обчислює значення правої частини рівнянь розглядуваної системи, мають вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ СИСТЕМИ НЕЛІНІЙНИХ          С
С РІВНЯНЬ МЕТОДОМ ПРОСТОЇ ІТЕРАЦІЇ                          С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL F
REAL X(2),Y(2)
K0=100
TYPE 100
ACCEPT *,NW
TYPE 101
TYPE 102
ACCEPT *,X(1)
TYPE 103
ACCEPT *,X(2)
EPS=.1E-03
N=2
CALL SITERS(N,X,Y,F,EPS,K0,K,IER,S)
IF(IER) 20,10,10
10 WRITE(NW,104) K,S,(X(I),I=1,N)
STOP
20 WRITE(NW,105)
STOP
100 FORMAT(1H$, 'NWRITE=' )
101 FORMAT(' ЗАДАЄМО ПОЧАТКОВЕ НАБЛИЖЕННЯ:')
102 FORMAT(1H$, 'X(1)=' )
103 FORMAT(1H$, 'X(2)=' )
104 FORMAT(/2X, ' КІЛЬКІСТЬ ІТЕРАЦІЙ:',I3/
*           ' НОРМА НЕВ'ЯЗКИ:',E12.5/
*           ' РОЗВ'ЯЗОК СИСТЕМИ:/'
*           *3X, 'X(1)=' ,E12.5,3X, 'X(2)=' ,E12.5)
105 FORMAT(/2X, 'ПОГАНЕ ПОЧАТКОВЕ НАБЛИЖЕННЯ')

```

END

```

SUBROUTINE F(N,X,Y)
С ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯНЬ СИСТЕМИ
REAL X(N),Y(N)
Y(1)=COS(X(2))+0.85
Y(2)=SIN(X(1))-1.32
RETURN
END

```

Обчислення на комп'ютері ДВК-2М дають такий результат:

КІЛЬКІСТЬ ІТЕРАЦІЙ: 6
 НОРМА НЕВ'ЯЗКИ: 0.99678E-04
 РОЗВ'ЯЗОК СИСТЕМИ:
 X(1)= 0.17913E+01 X(2)=-0.34423E+00

Приклад 2. Методом Ньютона розв'яжемо систему нелінійних рівнянь

$$\begin{cases} x_1^2 + x_2^2 + x_3^2 - 1 = 0; \\ 2x_1^2 + x_2^2 - 4x_3 = 0; \\ 3x_1^2 - 4x_2 - x_3^2 = 0. \end{cases}$$

Виходячи з геометричних міркувань, вибираємо початкове наближення розв'язку $x_1^{(0)} = x_2^{(0)} = x_3^{(0)} = 0,5$.

Головна програма, яка містить звернення до підпрограми NEWTS, та підпрограми-функції F і G, що обчислюють праву частину рівнянь розглядуваної системи та елементи матриці Якобі, мають вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ СИСТЕМИ С
С НЕЛІНІЙНИХ РІВНЯНЬ МЕТОДОМ НЬЮТОНА С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL F,G
REAL X(3),A(3,3),Y(3)
NF=4
K0=100
X(1)=.5
X(2)=.5
X(3)=.5
EPS=.5E-05
CALL NEWTS(3,X,A,F,G,EPS,K,K0)
CALL F(3,X,Y)
WRITE(NF,100) X,Y,K
STOP
100 FORMAT (//2X,'РОЗВ'ЯЗОК СИСТЕМИ: '//
* 13X,'X(1)=' ,E12.5/13X,'X(2)=' ,E12.5/
* 13X,'X(3)=' ,E12.5//
* 2X,'НЕВ'ЯЗКИ ДЛЯ НАБЛ. РОЗВ'ЯЗКУ: '//
* 13X,'Y(1)=' ,E12.5/13X,'Y(2)=' ,E12.5/
* 13X,'Y(3)=' ,E12.5//
* 'КІЛЬКІСТЬ ІТЕРАЦІЙ:',I3)
END

```

SUBROUTINE G(N,X,A)
 C ФОРМУВАННЯ МАТРИЦІ ЯКОБІ СИСТЕМИ РІВНЯНЬ
 REAL X(1),A(N,N)
 A(1,1)=2.*X(1)
 A(2,1)=4.*X(1)
 A(3,1)=6.*X(1)
 A(1,2)=2.*X(2)
 A(2,2)=2.*X(2)
 A(3,2)=-4.
 A(3,3)=2.*X(3)
 RETURN
 END

SUBROUTINE F(N,X,Y)
 B ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯНЬ СИСТЕМИ
 REAL X(1),Y(1)
 Y(1)=X(1)*X(1)+X(2)*X(2)+X(3)*X(3)-1.
 Y(2)=2.*X(1)*X(1)+X(2)*X(2)-4.*X(3)
 Y(3)=3.*X(1)*X(1)-4.*X(2)-X(3)*X(3)
 RETURN
 END

Обчислення на комп'ютері ДВК-2М дають такий результат:
 РОЗВ'ЯЗОК СИСТЕМИ:

$$\begin{aligned}
 X(1) &= 0.80747E+00 \\
 X(2) &= 0.45339E+00 \\
 X(3) &= 0.37740E+00
 \end{aligned}$$

НЕВ'ЯЗКИ ДЛЯ НАБЛ. РОЗВ'ЯЗКУ:

$$\begin{aligned}
 Y(1) &= 0.00000E+00 \\
 Y(2) &= -0.31352E-04 \\
 Y(3) &= 0.56326E-05
 \end{aligned}$$

КІЛЬКІСТЬ ІТЕРАЦІЙ: 33

Запитання для самоперіврки

1. Як з методів розв'язування нелінійного рівняння можна узагальнити на випадок розв'язування систем таких рівнянь? 2. Як записується достатня умова збіжності методу простої ітерації для системи нелінійних рівнянь? 3. Що є підставою до застосування методу Ньютона при розв'язуванні систем нелінійних рівнянь? 4. Яка швидкість збіжності методу Ньютона?

Контрольні завдання

1. Розв'язати систему нелінійних рівнянь

$$\begin{cases}
 y = x^2 (a_1x + b_1); \\
 x = y^2 (a_2y + b_2).
 \end{cases}$$

Значення її параметрів і похибки є наведено в табл. 3.

Таблиця 3

Номер варіанта	a_1	a_2	b_1	b_2	ε
1	10	20	19	36	0,005
2	100	100	10	20	0,002
3	150	150	5	10	0,001
4	100	50	10	30	0,001
5	50	100	15	20	0,001

Таблиця 4

Номер варіанта	R	a	b	c	d
1	5	4	4	1	1
2	5	5	5	1	2
3	5	7	7	2	1
4	5	5	5	2	-1
5	5	8	4	2	1

2. З точністю $\varepsilon = 10^{-4}$ розв'язати систему нелінійних рівнянь

$$\begin{cases} x^2 + y^2 = R^2; \\ y = \frac{ax + b}{cx + d}. \end{cases}$$

Значення її параметрів наведено в табл. 4.

Відповідь: $x = 3$, $y = 4$ (точний розв'язок).

3. З точністю $\varepsilon = 10^{-5}$ розв'язати систему нелінійних рівнянь

$$\begin{cases} 3x^2 - 4y + z^2 = 0; \\ 2x^2 + y^2 - 4z = -1; \\ \frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = d. \end{cases}$$

Значення її параметрів наведено в табл. 5.

Відповідь: $x = y = z = 1$ (точний розв'язок).

Таблиця 5

Номер варіанта	a	b	c	d
1	1	1	1	3
2	1	2	3	49/36
3	2	1	3	49/36
4	2	3	3	17/36
5	2	3	2	11/18

Таблиця 6

Номер варіанта	a	b	c
1	5	4	3
2	2	3	4
3	2,5	4	5
4	1	2	3,5
5	2	2	1

4. З точністю $\varepsilon = 10^{-4}$ розв'язати систему нелінійних рівнянь

$$\begin{cases} x + y + z = 0; \\ cx + ay + bz = 0; \\ (x + b)^2 + (y + c)^2 + (z + a)^2 = a^2 + b^2 + c^2. \end{cases}$$

Значення її параметрів залежно від номера варіанта наведено в табл. 6.

Відповідь: $x = a - 6$, $y = 6 - c$, $z = c - a$ (точний розв'язок).

5. Розв'язати системи нелінійних рівнянь з заданими початковими наближеннями та точністю розв'язків їх:

$$a) \begin{cases} x = \lg \frac{y}{z} + 1; & \text{при } x^{(0)} = 1,2; \quad y^{(0)} = 2,4; \\ y = 0,4 + z^2 - 2x^2; & z^{(0)} = 2,1; \quad \varepsilon = 10^{-4}; \\ z = 2 + xy/20 \end{cases}$$

$$б) \begin{cases} x + x^2 - 2yz = 0,1; & \text{при } x^{(0)} = y^{(0)} = z^{(0)} = 0; \\ y - y^2 + 3xz = 0,2; & \varepsilon = 10^{-5}; \\ z + z^2 + 2xy = 0 \end{cases}$$

$$в) \begin{cases} 3x^2 + 1,5y^2 + z^2 - 5 = 0; & \text{при } x^{(0)} = 1,25; \quad y^{(0)} = 0; \\ 6xyz - x + 5y + 3z = 0; & z^{(0)} = 0,25; \quad \varepsilon = 10^{-4}. \\ 5xz - yz - 1 = 0 \end{cases}$$

6. Розв'язати наближено таку нелінійну крайову задачу:

$$y'' = y^2 - 1; \quad y(0) = 0; \quad y(1) = 1.$$

Вказівка. Заданий інтервал значень функції y розбити на n однакових частин точками

$$0 = x_0 < x_1 < x_2 < \dots < x_{n-1} < x_n = 1.$$

Задане диференціальне рівняння замінити різницевою рівнянням

$$y_{i+1} - 2y_i + y_{i-1} = h^2 (y_i^2 - 1), \quad i = \overline{1, n-1};$$

$$y_0 = 0, \quad y_n = 1, \quad h = 1/n, \quad y_i = y(x_i).$$

Знайдену нелінійну систему рівнянь розв'язати методом Ньютона при $n = 20$. Початковим наближенням розв'язку є $y_i^{(0)} = x_i = ih$.

§ 5. Задачі на власні значення

Постановка задачі. Розв'язати повну або часткову задачу на власні значення для заданої матриці A , що визначається контрольним завданням.

Методичні вказівки. Ненульовий вектор \vec{x} називається *власним вектором* матриці A , якщо існує таке число λ , що

$$A\vec{x} = \lambda\vec{x}. \quad (1)$$

У цьому разі число λ називається *власним значенням* (власним числом), яке відповідає власному вектору \vec{x} .

Нехай $A = (a_{ij})$ є невідродженою квадратною матрицею порядку n . Система рівнянь (1) матиме нетривіальний (ненульовий) розв'язок тоді й тільки тоді, коли

$$\det(A - \lambda E) = 0. \quad (2)$$

Розкривши визначник у лівій частині (2), дістанемо відносно λ многочлен n -го степеня $R_n(\lambda)$ (характеристичний многочлен).

Отже, для заданої матриці A існує n власних значень (чисел, у загальному випадку — комплексних) λ_k , $k = \overline{1, n}$, серед яких можуть бути й однакові (кратні). Задача визначення цих чисел і відповідних власних векторів називається ще *повною проблемою власних значень* [1, с. 400—404; 10, с. 175—177; 14, с. 191—202].

▼ Для розв'язання повної проблеми власних значень використовують в основному два способи:

1) розкладання визначника в поліном n -го степеня з наступним знаходженням його коренів;

2) наближене визначення коренів характеристичного многочлена методом ітерації без попереднього розкладання характеристичного визначника.

Процес розв'язування повної проблеми власних значень може бути нестійким, особливо тоді, коли матриці мають велику розмірність (тобто корені та коефіцієнти полінома можуть бути чутливими до малих похибок заокруглення). Тому для розв'язування задачі, що розглядається, застосовують перетворення подібності [3, с. 62—69; 10, с. 170—189; 25, с. 174—178]. Найчастіше використовують перетворення відбиття й обертання. Метод відбиття є одним з найшвидших. Прямий метод обертання дещо поступається йому за швидкістю, проте розрахункові формули в ньому простіші.

Не дивлячись на свою швидкість, алгоритми прямих методів складаються з різнорідних частин і не спрощуються для спеціальних форм матриць. Тому застосовують ітераційні методи обертання, що мають деякі часткові переваги.

Одним з найбільш витончених методів розв'язування повної проблеми власних значень для дійсної симетричної матриці є метод обертання Якобі, який дає змогу звести матрицю до діагонального вигляду за допомогою послідовності елементарних перетворень. Існує велика кількість чисельних схем, пов'язаних з реалізацією цього методу. Відповідно до однієї з них [25, с. 182] $(k + 1)$ -й крок алгоритму здійснюється за таким правилом:

$$A_{k+1} = u_k^T A_k u_k,$$

де $u_k = u_k(p, q, \varphi)$ — ортогональна матриця, що відрізняється від одиничної тільки елементами $u_{pp} = u_{qq} = \cos \varphi$, $u_{pq} = u_{qp} = \sin \varphi$.

У цілому згідно з методом Якобі вхідна матриця $A = (a_{ij})$ перетворюється до вигляду $D = V^T A V$, де D — діагональна, а V — ортогональна матриці.

Елементи діагональної матриці D є шуканими власними значеннями заданої матриці A . Будується також матриця власних векторів, причому як початкове наближення береться одинична матриця.

Ітераційний процес закінчується, якщо всі позадіагональні елементи за модулем не перевищують числа

$$v = \left\{ 2 \sum_{i < k} a_{ik} \right\}^{1/2} \cdot 10^{-6/n}.$$

Перетворення виконуються за формулами [23, с. 300, 301]

$$\lambda = -a_{lm}; \quad \mu = 0,5(a_{ll} - a_{mm}); \quad \omega = \operatorname{sign} \mu \lambda / \sqrt{\lambda^2 + \mu^2};$$

$$\sin \theta = \omega / \sqrt{2(1 + \sqrt{1 - \omega^2})}; \quad \cos \theta = \sqrt{1 - \sin^2 \theta};$$

$$B = a_{ll} \cos \theta - a_{lm} \sin \theta; \quad C = a_{ll} \sin \theta + a_{lm} \cos \theta;$$

$$B = R_{ll} \cos \theta - R_{lm} \sin \theta; \quad R_{lm} = R_{ll} \sin \theta + R_{lm} \cos \theta;$$

$$R_{ll} = B; \quad a_{ll} = a_{ll} \sin^2 \theta + a_{mm} \sin^2 \theta - 2a_{lm} \sin \theta \cos \theta;$$

$$a_{mm} = a_{ll} \sin^2 \theta + a_{mm} \cos^2 \theta + 2a_{lm} \sin \theta \cos \theta;$$

$$a_{lm} = (a_{ll} - a_{mm}) \sin \theta \cos \theta + a_{lm} (\cos^2 \theta - \sin^2 \theta).$$

▼ При розв'язуванні деяких задач з ядерної фізики потрібно знати максимальне за модулем власне число, а при дослідженні коливальних процесів — два таких числа. Інколи виникає задача знаходження мінімального власного числа. Такі задачі, де не потрібно знаходити всі власні числа, називаються *частковою проблемою власних значень* [5, с. 169—172].

Формально методи розв'язування повної проблеми власних значень дають змогу розв'язувати і часткову проблему власних значень. Однак це потребує невиправдано великого обсягу обчислень. Тому для розв'язання часткової проблеми власних значень розроблено свої методи. Розглянемо один із них — *метод ітерації єдиного власного вектора*.

Припустимо, що виконується нерівність

$$|\lambda_1| > |\lambda_2| \geq \dots \geq |\lambda_n|$$

і матриця A має повну систему нормованих векторів $\vec{e}_1, \dots, \vec{e}_n$, що, наприклад, буде тоді, коли ця матриця симетрична.

Нехай $\vec{x}_0 = c_1 \vec{e}_1 + \dots + c_n \vec{e}_n$. Якщо $c_1 \neq 0$, то можна послідовно знаходити вектори

$$\vec{x}^{(k)} = A \vec{x}^{(k-1)}, \quad k = 1, 2, 3, \dots \quad (3)$$

Неважко показати (див., наприклад, [5, с. 170]), що при цьому дійсними будуть такі рівності:

$$(\vec{x}^{(k)}, \vec{x}^{(k-1)}) = \lambda_1^{2k-1} \left(c_1 + 0 \left(\left| \frac{\lambda_2}{\lambda_1} \right|^{k-1} \right) \right);$$

$$(\vec{x}^{(m)}, \vec{x}^{(m)}) = \lambda_1^{2m} \left(c_1^2 + 0 \left(\left| \frac{\lambda_2}{\lambda_1} \right|^m \right) \right).$$

Отже,

$$\lambda_{1k} = \frac{(\vec{x}^{(k)}, \vec{x}^{(k-1)})}{(\vec{x}^{(k-1)}, \vec{x}^{(k-1)})} = \lambda_1 + 0 \left(\left| \frac{\lambda_2}{\lambda_1} \right|^{k-1} \right);$$

$$\|\vec{x}^{(k)}\| = (\vec{x}^{(k)}, \vec{x}^{(k)})^{1/2} = |\lambda_1| \left(|c_1| + 0 \left(\left| \frac{\lambda_2}{\lambda_1} \right|^k \right) \right);$$

$$\vec{e}_1^{(k)} = \frac{\vec{x}^{(k)}}{\|\vec{x}^{(k)}\|} = (\text{sign } \lambda_1)^k \vec{e}_1 + \vec{r}^{(k)},$$

де $\vec{e}_1 = (\text{sign } c_1) \vec{e}_1$; $\|\vec{r}^{(k)}\| = 0 \left(\left| \frac{\lambda_2}{\lambda_1} \right|^k \right)$.

Таким чином, ітераційний процес (3) дає змогу знайти максимальне за модулем власне значення λ_1 і відповідний власний вектор \vec{e}_1 . Якщо $|\lambda_1| > 1$, то $\|\vec{x}^{(k)}\| \rightarrow \infty$, а якщо $|\lambda_1| < 1$, то $\|\vec{x}^{(k)}\| \rightarrow 0$ при $k \rightarrow \infty$. У першому випадку в ЕОМ може виникнути переповнення, в другому $\|\vec{x}^{(k)}\|$ може стати машинним нулем і інформація губиться. Тому ітерації доцільно вести за формулами

$$\vec{e}_1^{(0)} = \frac{\vec{x}^{(0)}}{\|\vec{x}^{(0)}\|}; \quad \vec{x}^{(k)} = A \vec{e}_1^{(k-1)};$$

$$\lambda_{1k} = (\vec{x}^{(k)}, \vec{e}_1^{(k-1)}); \quad \vec{e}_1^{(k)} = \frac{\vec{x}^{(k)}}{\|\vec{x}^{(k)}\|}.$$

Ітераційний процес може здійснитися, якщо власне значення змінюється менше, ніж деяке задане число (наприклад, 0,01 %), або якщо кількість ітерацій перевищила задане значення ITER.

Відзначимо, що коли ітераційний процес при різних початкових векторах $\vec{x}^{(0)}$ збігається до одного й того самого власного вектора (з точністю до протилежного вектора), то це свідчить про те, що λ_1 не є кратним власним значенням і не існує власного значення $-\lambda_1$.

Для знаходження λ_2, \vec{e}_2 можна скористатися таким ітераційним процесом:

$$\vec{x}^{(0)} - \text{довільний вектор}; \quad \vec{y}^{(0)} = \vec{x}^{(0)} - (\vec{x}^{(0)}, \vec{e}_1) \vec{e}_1;$$

$$\vec{e}_1 = \frac{\vec{y}^{(0)}}{\|\vec{y}^{(0)}\|}; \quad \vec{x}^{(k)} = A \vec{e}_2^{(k-1)}; \quad \lambda_{2k} = (\vec{x}^{(k)}, \vec{e}_2^{(k-1)});$$

$$\vec{y}^{(k)} = \vec{x}^{(k)} - (\vec{x}^{(k)}, \vec{e}_1^{(k)}) \vec{e}_1; \quad \vec{e}_2^{(k)} = \frac{\vec{y}^{(k)}}{\|\vec{y}^{(k)}\|}, \quad k = 1, 2, 3, \dots$$

Тут

$$\lambda_{2k} = \lambda_2 + 0 \left(\left| \frac{\lambda_3}{\lambda_2} \right|^{2k-1} \right); \quad \vec{e}_2^{(k)} = (\text{sign } \lambda_2)^k \vec{e}_2^1 + \vec{\rho}^{(k)};$$

$$\|\vec{\rho}^{(k)}\| = 0 \left(\left| \frac{\lambda_3}{\lambda_2} \right|^{2k} \right).$$

▼ Одним з основних питань алгебраїчної проблеми власних значень є з'ясування того, як сильно змінюються власні значення матриці при варіації її елементів. При цьому дійсною є така теорема [1, с. 417].

ТЕОРЕМА. Нехай дві n -вимірні матриці $A = (a_{ij})$, $B = (b_{ij})$ задовольняють умови $|a_{ij}| < 1$, $|b_{ij}| < 1$. Тоді власні значення λ і $\lambda(\epsilon)$ матриць A та $A + \epsilon B$ при деякому значенні $\epsilon > 0$ можна звести до взаємоднозначної відповідності так, що

$$|\lambda(\epsilon) - \lambda| < 2(n+1)^2 (n^2 \epsilon)^{1/n}. \quad (4)$$

Тому, хоча теоретично задача на власні значення є коректною, практично оцінка (4) накладає на ϵ надмірну умову мализни, і задача знаходження власних значень практично може виявитись некоректною. Зазначимо, що для симетричних і ермітових матриць проблема власних значень добре обумовлена.

Приклади використання програм. Приклад 1. Чисельну реалізацію методу обертання Якобі розглянемо на прикладі розв'язання повної проблеми власних значень для матриці

$$A = \begin{pmatrix} 2 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 2 \end{pmatrix}. \quad (5)$$

У цьому випадку головна програма, яка містить звернення до підпрограми EIGEN і реалізує метод обертання Якобі, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ЗНАХОДЖЕННЯ ВЛАСНИХ                               C
C ЗНАЧЕНЬ І ВЛАСНИХ ВЕКТОРІВ ДІЙСНОЇ                               C
C СИМЕТРИЧНОЇ МАТРИЦІ А                                           C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DIMENSION B(10,10),A(55), R(100)
C  ВВЕДЕННЯ ПОЧАТКОВИХ ДАНИХ
      CALL VID(N,B,A)
      MV=1
C  ЗНАХОДЖЕННЯ ВЛАСНИХ ЗНАЧЕНЬ І
C      ВЛАСНИХ ВЕКТОРІВ
      CALL EIGEN(A,R,N,MV)
C  ДРУКУВАННЯ РЕЗУЛЬТАТУ
      CALL PR(1,A,R,N)
      STOP
      END

```

```

SUBROUTINE VID(N,B,A)
С ПІДПРОГРАМА ВВЕДЕННЯ ПОЧАТКОВИХ ДАНИХ
DIMENSION B(N,N),A(1)
N=3
B(1,1)=2.
B(1,2)=-1.
B(2,2)=2.
B(1,3)=0.
B(2,3)=-1.
B(3,3)=2.
A(1)=B(1,1)
A(2)=B(1,2)
A(3)=B(2,2)
A(4)=B(1,3)
A(5)=B(2,3)
A(6)=B(3,3)
RETURN
END

```

```

SUBROUTINE PR(MV,A,R,N)
С ПІДПРОГРАМА ДРУКУВАННЯ РЕЗУЛЬТАТУ
DIMENSION A(1),R(1)
IF(MV-1)10,30,10
10 DO 20 I=1,N
20 WRITE(2,101)(R(K),K=1,N)
30 DO 40 I=1,N
K=I*(I+1)/2
40 WRITE(2,100)A(K)
RETURN
100 FORMAT(2X,'ВЛАСНЕ ЗНАЧЕННЯ=',E12.5)
101 FORMAT(2X,'ВЛАСНИЙ ВЕКТОР=',E12.5)
END

```

Обчислення на комп'ютері ДВК-2М дають такий результат:

```

ВЛАСНЕ ЗНАЧЕННЯ= 0.58579E+00
ВЛАСНЕ ЗНАЧЕННЯ= 0.20000E+00
ВЛАСНЕ ЗНАЧЕННЯ= 0.34142E+01

```

Для порівняння наведемо точні власні значення: $\lambda_1 = 0,585786$; $\lambda_2 = 2$; $\lambda_3 = 3,4142135$.

Приклад 2. Знайдемо максимальне власне значення та відповідний власний вектор матриці (5), використовуючи підпрограму SVISZ.

Головна програма в цьому випадку має вигляд:

```

С ГОЛОВНА ПРОГРАМА ОБЧИСЛЕННЯ МАКСИМАЛЬНОГО
С ВЛАСНОГО ЗНАЧЕННЯ ТА ВІДПОВІДНОГО
С ВЛАСНОГО ВЕКТОРА
С ВЛАСНОГО ЗНАЧЕННЯ
С ДРУКУВАННЯ РЕЗУЛЬТАТУ
DIMENSION X(20),Y(20),A(20,20)
CALL DAT(X,N,ITER,A,Y,RL)
CALL SVISZ(X,N,ITER,A,Y,RL)

```

```

CALL PCR(N,ITER,RL,Y)
STOP
END

```

```

SUBROUTINE DAT(X,N,ITER,A,RL)
C ПІДПРОГРАМА ВВЕДЕННЯ ПОЧАТКОВИХ ДАНИХ
  DIMENSION X(N),A(N,N)
C ITER—МАКСИМАЛЬНА КІЛЬКІСТЬ ІТЕРАЦІЙ
  ITER=100
  TYPE 100
100 FORMAT(1H$, ' N')
  ACCEPT *,N
C ЗАДАВАННЯ ПОЧАТКОВОГО НАБЛИЖЕННЯ
  DO 1 I=1,N
    TYPE 101,I
101 FORMAT(1H$, ' X(',12,')=')
    ACCEPT *,X(I)
  1 CONTINUE
C ВХІДНА МАТРИЦЯ А
  DO 2 I=1,N
    DO 2 J=1,N
      TYPE 102,I,J
102 FORMAT(1H$, ' A(', 12,',', 12,')=')
      ACCEPT *,A(I,J)
  2 CONTINUE
  RL=0.
  RETURN
END

```

```

SUBROUTINE PCR(N,ITER,RL,Y)
C ПІДПРОГРАМА ДРУКУВАННЯ РЕЗУЛЬТАТУ
  DIMENSION Y(N)
  WRITE(3,100)
  WRITE(3,101)
  WRITE(3,102)
  WRITE(3,101)
  WRITE(3,103)RL,Y(1),ITER
  WRITE(3,104)(Y(I),I=2,N)
  WRITE(3,105)
  RETURN
100 FORMAT(2X,50('—'))
101 FORMAT(1X,'I',50X,'I')
102 FORMAT(1X,'I',3X,'ВЛАСНЕ.ЗНАЧ.',5X,
  * 'НОРМ.ВЛАСН.ВЕКТ.',2X,
  * 'КІЛ.ІТЕР.',3X,'I')
103 FORMAT(1X,'I',3X,E12.5,6X,E12.5,5X,I3,7X,'I')
104 FORMAT(1X,'I',21X,E12.5,15X,'I')
105 FORMAT(2X,50('—'))
END

```

Обчислення на комп'ютері ДВК-2М дають такий результат:

ВЛАСНЕ ЗНАЧ. 0.34142E+00

НОРМ. ВЛАСН. ВЕКТ.
 0.50007E+00
 -0.70700E+00
 0.50007E+00

КІЛ. ІТЕР.

6

Запитання для самоперевірки

1. Чому для матриць великої розмірності не бажано застосовувати метод розкриття характеристичного многочлена? 2. Які недоліки та переваги прямих методів знаходження власних значень? 3. У чому полягає часткова проблема власних значень? 4. Чи можна розв'язувати часткову проблему власних чисел за допомогою повної проблеми і чому? 5. Нехай ітераційний процес збігається до одного й того самого власного вектора за різних початкових умов. Про що це свідчить?

Контрольні завдання

1. Знайти всі власні вектори та значення матриці

$$A = \begin{pmatrix} 0,25(a+c) + 0,5b & 0,25\sqrt{2}(a-c) & 0,25 - 0,5b + 0,25\sqrt{2}c \\ 0,25\sqrt{2}(a-c) & 0,5(a+c) & 0,25\sqrt{2}a - 0,5c \\ 0,25a - 0,5b + 0,25\sqrt{2}c & 0,25\sqrt{2}a - 0,5c & 0,25a + 0,5(b+c) \end{pmatrix}.$$

де елементи $a = \frac{k+3}{12}$, $b = \frac{k+7}{14}$, $c = \frac{k+10}{15}$, $k \in \mathbb{N}$ задаються викладачем.

Відповідь: $\lambda_1 = a$; $\lambda_2 = b$; $\lambda_3 = c$.

2. Знайти максимальне власне значення та відповідний власний вектор для матриці, наведеної в попередньому завданні, використовуючи підпрограму SVISZ.

3. Баржа призначена для перевезення через озеро зчепу з шести залізничних вагонів. Буксир тягне її за носову частину. Вагони мають масу m_i , $i = \overline{1, 6}$; коефіцієнти жорсткості елементів становлять k_i , $i = \overline{1, 6}$. Існує побоювання, що при хвилюванні на озері можуть виникнути резонансні поздовжні коливання. Визначити шість власних значень зазначеної механічної системи і порівняти їх за частотою хвилі, що дорівнює 1 рад/с. Власні частоти пов'язані з власними значеннями динамічної матриці D співвідношенням

$$\omega_i = \sqrt{|\lambda_i|}.$$

Динамічна матриця утворюється з матриць жорсткості K і мас M , причому $D = K^{-1}M$, а $k_1 = 5 \cdot 10^{-5}$ Н/м, $k_i = 10^{-5}$ Н/м ($i = \overline{2, 6}$), $m_1 = 8 \cdot 10^4$ кг, $m_2 = m_4 = 3 \cdot 10^4$ кг, $m_3 = m_5 = 4 \cdot 10^4$ кг, $m_6 = 2 \cdot 10^4$ кг.

4. Консольний брус завдовжки 10 м, що має жорсткість при згині $EI = 10$ Н/м і погонну масу 10 кг/м, апроксимується двома точковими масами по 50 кг кожна, розміщеними в центрі та на вільному кінці бруса. Знайти дві основні частоти коливання бруса, якщо відомо власні значення динамічної матриці $D = \Gamma M$, де Γ — матриця згину, елементами i -го рядка якої є відхилення точки j під дією одиничної сили, прикладеної в точці i ; M — діагональна матриця, на діагоналі якої знаходяться маси точок. Осьова сила відсутня. Деформацією зсуву знехтувати.

5. Три одиничні маси підвішено на пружинах одну до одної з коефіцієнтами пружності c_1, c_2, c_3 відповідно. Визначити основні частоти та форми коливань системи для випадку, коли $c_1 = c_2 = c_3 = 1$.

Вказівка. Якщо сили, які діють на маси, позначити через F_i , а відхилення від положення рівноваги мас — через y_i ($i = \overline{1, 2, 3}$), то рівняння системи матимуть вигляд

$$\begin{aligned} -\frac{d^2 y_1}{dt^2} &= F_1 - F_2 = c_1 y_1 - c_2 (y_2 - y_1); \\ -\frac{d^2 y_2}{dt^2} &= F_2 - F_3 = c_2 (y_2 - y_1) - c_3 (y_3 - y_2); \\ -\frac{d^2 y_3}{dt^2} &= F_3 = c_3 (y_3 - y_2). \end{aligned}$$

Після введення заміни $\vec{y} = \vec{x} \sin \omega t$ контрольне завдання зводиться до задачі відшукування власних значень $\vec{A}\vec{x} = \lambda\vec{x}$, де

$$A = \begin{pmatrix} c_1 + c_2 & c_2 & 0 \\ -c_2 & c_2 + c_3 & -c_3 \\ 0 & -c_3 & c_3 \end{pmatrix},$$

а $\lambda = \omega^2$.

6. Протяжне ізотропне середовище зазнає дії деформації, внаслідок чого одиничний куб з ребрами

$$\vec{c}_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}; \quad \vec{c}_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}; \quad \vec{c}_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

переходить у паралелограм з ребрами

$$\vec{e}_1 = \begin{pmatrix} 2 \\ -1 \\ 0 \end{pmatrix}; \quad \vec{e}_2 = \begin{pmatrix} -1 \\ 2 \\ -1 \end{pmatrix}; \quad \vec{e}_3 = \begin{pmatrix} 0 \\ -1 \\ 2 \end{pmatrix}.$$

Знайти головні осі деформації, тобто напрямки, які під час її дії не змінюються.

Вказівка. Деформація описується симетричною матрицею A , для якої $\vec{A}\vec{e}_i = \vec{e}_i$.

Шукані напрямки є власними векторами \vec{x} матриці A .

7. Визначити мінімальне власне значення $\lambda_1(A)$ і відповідний власний вектор матриці A , наведеної в контрольному завданні 1.

Вказівка. Мінімальне власне значення $\lambda_1(A)$ можна знайти, знаючи максимальне власне значення $\lambda_n(A)$ та максимальні власні значення матриць B_1 і B_2 , тобто

$$B_1 = \|A\|E - A; \quad B_2 = \lambda_n(A)E - A.$$

§ 6. Інтерполювання многочленами

Постановка задачі. За даними наборами значень аргументу x_1, x_2, \dots, x_n відповідних значень функції та її похідних $f(x_1), f'(x_1), \dots, f^{(m_1-1)}(x_1), f(x_2), f'(x_2), f''(x_2), \dots, f^{(m_2-1)}(x_2), \dots, f(x_n), \dots, f^{(m_n-1)}(x_n)$, $m_i \geq 1$, $\sum_{i=1}^n m_i = s$ знайти алгебраїчний многочлен $P_{s-1}(x)$ степеня $s-1$ такий, що $P_{s-1}^{(j)}(x_i) = f^{(j)}(x_i)$, $i = \overline{1, n}$, $j = \overline{1, m_i - 1}$.

▼ За даними значеннями $x_i, f^{(j)}(x_i)$, $i = \overline{1, n}$, $j = \overline{1, m_i - 1}$, $\sum_{i=1}^n m_i = s$, $m_i \geq 1$ та x ($x \neq x_i$, $i = \overline{1, n}$) знайти $P_{s-1}(x)$.

Методичні вказівки. *Задача інтерполювання многочленами.* Нехай $f(x) \in C[a, b]$ і $\{\varphi_i(x)\}_{i=0}^n \subset C[a, b]$ — система лінійно незалежних функцій. Лінійна комбінація

$$\varphi(x) = \sum_{i=0}^n a_i \varphi_i(x)$$

з дійсними коефіцієнтами a_i називається *узагальненим многочленом* степеня n по системі функцій $\{\varphi_i(x)\}_{i=0}^n$. Узагальнений многочлен $\varphi^*(x)$ такий, що $\varphi^*(x_i) = f(x_i)$, $i = 0, n$, називається *інтерполяційним* для функції $f(x)$. Умови того, щоб задача побудови узагальненого інтерполяційного многочлена мала єдиний розв'язок, задаються такою теоремою [10].

ТЕОРЕМА. Для того щоб існував єдиний узагальнений інтерполяційний многочлен, необхідно й достатньо, щоб система функцій $\{\varphi_i(x)\}_{i=0}^n$ була системою Чебишова.

Однією з найбільш зручних для практичного використання є система функцій $\varphi_i(x) = x^i$, $i = 0, n$, яка є системою Чебишова на будь-якому скінченному відрізку $[a, b] \in (-\infty, \infty)$. Відповідний інтерполяційний многочлен, який є алгебраїчним многочленом не вище n -го степеня, позначається так:

$$P_n(x) \equiv P_n(x; f) = \sum_{i=0}^n a_i x^i. \quad (1)$$

Існують різні форми запису інтерполяційного многочлена. Наприклад, неважко бачити, що многочлен

$$P_n(x) = \sum_{i=0}^n f(x_i) l_{n,i}(x), \quad (2)$$

де $l_{n,i}(x) = \prod_{\substack{j=0 \\ j \neq i}}^n (x - x_j) / \prod_{\substack{j=0 \\ j \neq i}}^n (x_i - x_j)$ — фундаментальні інтерполяційні

многочлени, причому $l_{n,i}(x_j) = \delta_{ij}$ (δ_{ij} — символ Кронекера), є інтерполяційним. Він називається *інтерполяційним многочленом у формі Лагранжа*.

▼ *Інтерполяційний многочлен у формі Ньютона.* Ціле число m_i (див. постановку задачі) називається *кратністю вузла x_i* . Розглянемо спочатку випадок, коли $m_i = 1$, $i = 1, n$. Таблиця

$$\begin{array}{llll} f(x_1) & & & \\ f(x_2) & f(x_1; x_2) & & \\ f(x_3) & f(x_2; x_3) & f(x_1; x_2; x_3) & \\ \dots & \dots & \dots & \\ f(x_n) & f(x_{n-1}; x_n) & f(x_{n-2}; x_{n-1}; x_n) & \dots f(x_1; \dots; x_n), \end{array}$$

де

$$f(x_i; x_j) = \frac{f(x_j) - f(x_i)}{x_j - x_i}; \quad f(x_i; x_j; x_k) = \frac{f(x_j; x_k) - f(x_i; x_j)}{x_k - x_i},$$

$$\dots, f(x_i; x_j; \dots; x_k; x_l) = \frac{f(x_j; \dots; x_k; x_l) - f(x_i; x_j; \dots; x_k)}{x_l - x_i}$$

є розділеними різницями відповідно першого, другого і т. д. порядків, називається *таблицею розділених різниць*. Якщо через c_{ik} позначити елемент цієї таблиці, розміщений в i -му рядку та k -му стовпцю, то

$$c_{ik} = (c_{i,k-1} - c_{i-1,k-1}) / (x_i - x_{i-k}), \quad k = \overline{1, n}; \quad i = n, n-1, \dots, k.$$

Ця формула лежить в основі алгоритму, реалізованого в підпрограмі DD.

Інтерполяційний многочлен у формі Ньютона [3] має вигляд

$$\begin{aligned} P(x; f) \equiv P_{n-1}(x; f) &= f(x_1) + (x - x_1)f(x_1; x_2) + \\ &+ (x - x_1)(x - x_2)f(x_1; x_2; x_3) + \dots + (x - x_1)(x - x_2) \dots \\ &\dots (x - x_{n-1})f(x_1; x_2; \dots; x_n). \end{aligned} \quad (3)$$

▼ *Інтерполяційний многочлен з кратними вузлами.* Якщо хоча б одне з чисел m_i більше від одиниці, то многочлен $P_{s-1}(x)$, визначений в постановці задачі, називається *інтерполяційним многочленом з кратними вузлами*, або *інтерполяційним многочленом Ерміта*.

Многочлен Ерміта існує і він єдиний [2, с. 47]. Його можна записати у вигляді

$$\begin{aligned} P_{s-1}(x) \equiv P_{s-1}(x; f) &= f(x_1) + (x - x_1)f(x_1; x_1) + \\ &+ (x - x_1)^2 f(x_1; x_1; x_1) + \dots + (x - x_1)^{m_1-1} f(x_1; \dots; x_1) + \\ &+ (x - x_1)^{m_1} f(x_1; \dots; x_1; x_2) + (x - x_1)^{m_1} (x - x_2) \times \\ &\times f(x_1; \dots; x_1; x_2; x_2) + \dots + (x - x_1)^{m_1} (x - x_2)^{m_2} \dots \\ &\dots (x - x_{n-1})^{m_{n-1}} f(x_1; \dots; x_1; x_2; \dots; x_2; \dots \\ &\dots; x_{n-1}; \dots; x_{n-1}; x_n) + (x - x_1)^{m_1} (x - x_2)^{m_2} \dots (x - x_{n-1})^{m_{n-1}} \times \\ &\times (x - x_n) f(x_1; \dots; x_1; x_2; \dots; x_2; \dots; x_{n-1}; \dots; x_{n-1}; x_n; x_n) + \\ &+ \dots + (x - x_1)^{m_1} (x - x_2)^{m_2} \dots (x - x_{n-1})^{m_{n-1}} (x - x_n)^{m_{n-1}} \times \\ &\times f(x_1; \dots; x_1; \dots; x_n; \dots; x_n), \end{aligned} \quad (4)$$

де розділені різниці з кратними вузлами $f(x_1), f(x_1, x_1), \dots, f(x_1; \dots; x_1; \dots; x_n; \dots; x_n)$ беруться з такої таблиці:

$$\begin{array}{l}
 m_1 \left\{ \begin{array}{l} f(x_1) \\ f(x_1) f(x_1; x_1) \\ f(x_1) f(x_1, x_1) f(x_1; x_1; x_1) \\ \dots \\ f(x_1) f(x_1; x_1) f(x_1; x_1; x_1) \dots \underbrace{f(x_1; \dots; x_1)}_{m_1} \end{array} \right. \\
 \left\{ \begin{array}{l} f(x_2) f(x_1; x_2) f(x_1; x_1; x_2) \dots \underbrace{f(x_1; \dots; x_1; x_2)}_{m_1-1} \underbrace{f(x_1; \dots; x_1; x_2)}_{m_1} \\ f(x_2) f(x_2; x_2) f(x_1; x_2; x_2) \dots \\ \dots \underbrace{f(x_1; \dots; x_1; x_2; x_2)}_{m_1-2} \underbrace{f(x_1; \dots; x_1; x_2; x_2)}_{m_1-1} \\ \dots \\ f(x_2) f(x_1; x_2) f(x_2; x_2; x_2) \dots \end{array} \right. \\
 m_2 \left\{ \begin{array}{l} f(x_n) f(x_{n-1}; x_n) f(x_{n-1}; x_{n-1}; x_n) \\ \dots \\ f(x_n) f(x_n; x_n) f(x_n; x_n; x_n) \dots \underbrace{f(x_1; \dots; x_1; x_n; \dots; x_n)}_{m_1} \underbrace{f(x_1; \dots; x_1; x_n; \dots; x_n)}_{m_n} \end{array} \right. \\
 m_n
 \end{array}$$

Ця таблиця заповнюється з використанням формули

$$\underbrace{f(x_i; \dots; x_i)}_{p+1} = f^{(p)}(x_i)/p!$$

і рекурентного співвідношення при $i \neq j$

$$\underbrace{f(x_i; \dots; x_i; \dots; x_j; \dots; x_j)}_{m_i-l+1} = \underbrace{(f(x_i; \dots; x_i; \dots; x_j; \dots; x_j))}_{p} - \underbrace{f(x_i; \dots; x_i; x_j; \dots; x_j)}_{m_i-l+1} \underbrace{(x_j - x_i)}_{p-1}$$

Якщо позначити через c_{ij} елемент таблиці розділених різниць з кратними вузлами в i -му рядку та j -му стовпцю, то $c_{i1} = f(x'_i)$, $i = 1, s$;

$$c_{ij} = \begin{cases} (c_{ij-1} - c_{i-1, j-1}) / (x'_i - x'_{i-j+1}), & x'_i \neq x'_{i-j+1}; \\ f^{(j-1)}(x_i) / (j-1)!, & x'_i = x'_{i-j+1}; \end{cases} \quad j = 2, n; \quad i = n, n-1, \dots, 1$$

де $s = \sum_{i=1}^n m_i$; $x' = \{x'_i\}_{i=1}^s = \{\underbrace{x_1; \dots; x_1}_{m_1}; \dots; \underbrace{x_n; \dots; x_n}_{m_n}\}$.

При $m_i = 1, i = \overline{1, n}$ розділені різниці з кратними вузлами збігаються зі звичайними розділеними різницями [2, с. 47], а інтерполяційний многочлен з кратними вузлами — зі звичайним інтерполяційним многочленом у формі Ньютона.

Очевидно, многочлен $P_{s-1}(x)$ можна подати у вигляді

$$P_{s-1}(x) \equiv P_{s-1}(x; f) = \sum_{j=1}^n \sum_{k=0}^{m_j-1} f^{(k)}(x_j) l_{j,k}(x), \quad (5)$$

де $l_{j,k}(x)$ — фундаментальні інтерполяційні многочлени з кратними вузлами, що задовольняють умови $l'_{j,k}(x_i) = \delta_{ji} \delta_{kr}$, $k = \overline{0, m_j - 1}$, $r = \overline{0, m_i - 1}$, $j, i = \overline{1, n}$ (δ_{ij} — символ Кронекера).

▼ *Схема обчислення інтерполяційного многочлена за Горнером.* Розглянемо алгоритм обчислення в точці x многочлена, заданого у вигляді

$$P_{s-1}(x) = d_1 + d_2(x - x'_1) + d_3(x - x'_1)(x - x'_2) + \dots + d_s(x - x'_1) \dots (x - x'_{s-1}). \quad (6)$$

Саме такий вигляд мають інтерполяційні многочлени (3) та (4), якщо покласти $d_1 = f(x_1)$, $d_2 = f(x_1; x_1)$, ...,

$$d_s = f(\underbrace{x_1; \dots; x_1}_{m_1}; \dots; \underbrace{x_n; \dots; x_n}_{m_n}).$$

Запис многочлена (6) у вигляді

$$P_{s-1}(x) = (\dots ((0(x - x'_s) + d_s(x - x'_{s-1}) + d_{s-1})(x - x'_{s-2}) + \dots + d_2)(x - x'_1) + d_1$$

підказує такий спосіб його обчислення, що називається *схемою Горнера*:

$$b_{j+1} = b_j(x - x'_{s-j}) + d_{s-j}, \quad j = \overline{0, s-1}, \quad b_0 = 0; \quad (7)$$

$$P_{s-1}(x) = b_s.$$

Аналіз похибок заокруглення показує, що похибка обчислення многочлена $P_{s-1}(x)$ за формулою (7) може бути значною, якщо абсолютні значення d_j , $j = \overline{0, s}$ великі [1, с. 479].

На практиці може виникнути необхідність запису інтерполяційного многочлена не в формі Ньютона (3) або (4), а, наприклад, у вигляді (1), тобто

$$P_{s-1}(x) = d'_1 + d'_2 x + \dots + d'_s x^{s-1}, \quad (8)$$

або

$$P_{s-1}(x) = d''_1 + d''_2 T_1(x) + \dots + d''_s T_{s-1}(x), \quad (9)$$

де $T_k(x)$ — многочлен Чебишова зі старшим коефіцієнтом 1, тобто $T_k(x) = 2^{1-k}t_k(x)$, $k = 1, 2, \dots$, а $t_k(x)$ визначається рекурентними співвідношеннями

$$t_k(x) = 2xt_{k-1}(x) - t_{k-2}(x), \quad k = 2, 3, \dots; \\ t_0(x) = 1, \quad t_1(x) = x$$

або явним виразом $t_k(x) = \cos(k \arccos x)$ [3, с. 56].

Якщо деякий многочлен $P_{s-1}(x)$ потрібно обчислювати багато разів у різних точках, то його подання у вигляді (9) має перевагу перед записами (3), (4) чи (8). Це пов'язано з тим, що для многочлена $P_{s-1}(x)$ вигляду (8), який на відрізку $[0, 1]$ за модулем не перевищує одиниці, типова оцінка суми модулів коефіцієнтів (а саме вона визначає похибку заокруглення) визначається виразом $\sum_i |d_i| \sim c^{s-1}$, де $c = \text{const} > 1$ [1, с. 480]. У той же час сума модулів коефіцієнтів у (9) росте не швидше, ніж $c\sqrt{s-1}$ [1, с. 480].

Зауважимо, що сума (9) є окремим випадком суми вигляду

$$S_{s-1}(x) = \sum_{k=1}^{s-1} d_{k+1}q_k(x),$$

де функції $q_k(x)$ задовольняють лінійне рекурентне співвідношення другого порядку

$$q_{k+1}(x) + \alpha_k(x)q_k(x) + \beta_k(x)q_{k-1}(x) = 0, \quad k = \overline{0, s-2}, \quad q_{-1}(x) \equiv 0,$$

причому функція $q_0(x)$ є заданою.

Для обчислення суми $S_{s-1}(x)$ у заданій точці x застосовується таке узагальнення схеми Горнера [1]:

$$A_j = d_{j+1} - \alpha_j A_{j+1} - \beta_{j+1} A_{j+2}, \quad j = s-1, s-2, \dots, 0;$$

$$A_s = A_{s+1} = 0; \quad S_{s-1}(x) = A_0 q_0.$$

При $\alpha_j = -x$, $\beta_j = 0$, $j = 0, 1, \dots$ дістаємо схему Горнера (7).

▼ *Похибка інтерполювання.* Залишковий член інтерполяційного многочлена з кратними вузлами має вигляд [10, с. 35]

$$R(x) \equiv R_{s-1}(x) \equiv R_{s-1}(x; f) = f(x) - P_{s-1}(x; f) = f^{(s)}(\xi) \omega_s(x)/s!,$$

$$\text{де } \xi \in (a, b); \quad a = \min x_i; \quad b = \max x_i; \quad \omega_s = \prod_{j=1}^n (x - x_j)^{m_j}.$$

Для звичайного інтерполяційного многочлена $P_{n-1}(x; f)$ маємо

$$R_{n-1}(x; f) = f(x) - P_{n-1}(x; f) = \frac{f^{(n)}(\xi)}{n!} \prod_{j=1}^n (x - x_j). \quad (10)$$

Якщо $f(x) \in C[a, b]$, то має місце нерівність Лебега [1, с. 189]

$$\|R_{n-1}(x)\|_{C[a,b]} \equiv \max_{x \in [a,b]} |R_{n-1}(x; f)| \leq (1 + \Lambda_n) E_{n-1}(f), \quad (11)$$

де $\Lambda_n = \max_{x \in [a, b]} \sum_{k=1}^n |l_{n,k}(x)|$ — стала Лебега; $E_{n-1}(f) = \inf_{p \in \pi_{n-1}} \max_{x \in [a, b]} |f(x) - p(x)|$ — значення найкращого рівномірного наближення до функції $f(x)$ деякого многочлена з множини π_{n-1} многочленів не вище $(n-1)$ -го степеня. Стала Лебега не залежить від довжини інтерполювання $[a, b]$, а залежить від відносного розміщення вузлів на ньому.

Якщо для розв'язування задачі, тобто для наближеного обчислення значення функції $f(x)$, застосовується інтерполяційний многочлен $P_{n-1}(x)$, то, крім похибки методу, виникає ще похибка від того, що значення функції $f(x)$ у вузлах x_i обчислено не точно. Нехай похибка при обчисленні функції $f(x_h)$ становить ϵ_h . Похибки ϵ_h вносять при обчисленні $P_{n-1}(x; f)$ сумарну похибку $\delta = \max_{x \in [a, b]} \left| \sum_{k=1}^n \epsilon_k l_{n,k}(x) \right|$. При $\epsilon_h \approx \epsilon$ і несприятливому збігу знаків величин $\delta \approx \epsilon \Lambda_n$.

Таким чином, стала Лебега можна назвати сталою обумовленості задачі обчислення інтерполяційного многочлена в точці: якщо ця стала велика, то слід чекати великого впливу похибок заокруглення і в цьому випадку задача є погано обумовленою.

Для інтерполювання алгебраїчними многочленами оптимальні вузли, для яких стала Лебега мінімальна, є невідомими, хоча доведено, що вони існують [1, с. 190]. Доведено також, що стала Лебега інтерполяційного многочлена з вузлами в нулях многочлена Чебишова першого роду

$$x_i = \cos \frac{2i-1}{2n} \pi, \quad i = \overline{1, n}, \quad x_i \in (-1, 1)$$

визначається рівністю

$$\Lambda_n = \frac{2}{\pi} \ln n + 1 - \theta_n, \quad 0 \leq \theta_n < \frac{1}{n},$$

причому, якщо Λ_* — оптимальна стала Лебега, то

$$\frac{2}{\pi} \ln n + 1 - \theta_n - 0,201 < \Lambda_* < \frac{2}{\pi} \ln n + 1 - \theta_n.$$

Для порівняння наведемо таку оцінку сталої Лебега щодо рівновіддалених вузлів у разі інтерполювання алгебраїчними многочленами:

$$2^{n-3} (n-1)^{-1/2} (n-3/2)^{-1} < \Lambda_n < 2^{n-1}, \quad n \geq 4.$$

Звідси випливають важливі висновки: 1) не слід використовувати на практиці алгебраїчні інтерполяційні многочлени з рівновіддаленими вузлами при великій кількості їх; 2) стала Лебега дуже чутлива до вибору вузлів інтерполювання, тому їх краще вибирати як нулі многочленів Чебишова першого роду $t_n(x)$ або як нулі многочлена

$(1 - x^2) U_{n-2}(x)$, де $U_{n-2}(x)$ — многочлен Чебишова другого роду [1, с. 192].

Порядок спадання наближення $E_{n-1}(f)$ при $n \rightarrow \infty$ залежить від гладкості функції f . Наприклад, якщо остання має $r \geq 1$ похідних, інтегрованих з p -м степенем, то [1, с. 159] $E_{n-1}(f) = O(n^{-r})$.

Теорема Джексона стверджує: якщо $f \in C^{k,\lambda}[a, b]$, то $E_{n-1}(f) \leq \leq Cn^{-k-\lambda}$, де $C^{k,\lambda}[a, b]$ — клас функцій, які мають похідні до k -го порядку, причому похідна k -го порядку задовольняє умову Ліпшица з показником λ .

Таким чином, знаючи вузли інтерполювання, від яких залежить асимптотика по n сталогі Лебега, а також гладкість функції f , з нерівності (11) можна вивести теореми про рівномірну збіжність многочлена $P_{n-1}(x; f)$ до функції $f(x)$ при $n \rightarrow \infty$. Проте, якщо $f(x) \in C[a, b]$, то цього для збіжності може виявитись мало.

Відомо [10, с. 43], що при будь-якому наборі вузлів $x_i, i = \overline{1, n}$ існує функція $f(x) \in C[a, b]$, для якої дійсною є нерівність

$$\lim_{n \rightarrow \infty} P_{n-1}(x; f) \neq f(x).$$

Для доброго наближення многочлена $P_{n-1}(x; f)$ навіть до досить гладких функцій $f(x)$ не обов'язково збільшувати n . Можна також, залишаючи n фіксованим, згущувати вузли інтерполювання навколо точки x . У цьому разі, як впливає з (10), завжди можна досягти того, що $|R_{n-1}(x; f)| < \varepsilon$ для будь-якого заздалегідь заданого значення $\varepsilon > 0$.

На практиці здебільшого вузли інтерполювання вибирають рівновіддаленими (при невеликому n) з кроком h і, користуючись (10), а також тим, що $|\omega_n(x)| = |(x - x_1) \dots (x - x_n)| \leq ((n - 1)h)^n$, вибирають крок h , виходячи з умови

$$M_n ((n - 1)h)^n \leq \varepsilon n!, \quad (12)$$

де

$$M_n = \max_{x \in [a, b]} |f^{(n)}(x)|.$$

Очевидно, коли вузли рівновіддалені з кроком h , а точка лежить між крайніми вузлами інтерполювання, то $|\omega_s(x)| = O(h^s)$ (при фіксованому s), тобто порядок точності по h як ермітового, так і звичайного інтерполювання на одиницю більший від степеня інтерполяційного многочлена.

▼ *Застосування інтерполяційного многочлена.* Інтерполяційний многочлен $P_{n-1}(x; f)$ для функції $f(x)$ застосовується при побудові формул чисельного диференціювання ($f^{(k)}(x) \approx P_{n-1}^{(k)}(x; f)$), чисельного

інтегрування $\left(\int_a^b \rho(x) f(x) dx \approx \int_a^b \rho(x) P_{n-1}(x; f) dx \right)$, розв'язування рівнянь тощо.

Нехай, наприклад, потрібно розв'язати рівняння $f(x) = \bar{y}$. Якщо функція $f(x)$ на відрізку $[a, b]$, що містить шуканий корінь, є монотонною, то спочатку за таблицею значень $x_i, y_i \equiv f(x_i), i = \bar{1}, n, x_i \in [a, b]$, вважаючи y_i значеннями незалежної, а x_i — залежної змінних, будують інтерполяційний многочлен $x = Q_{n-1}(y)$, після чого наближене значення кореня обчислюють за формулою $x = Q_{n-1}(y)$. Цей метод розв'язування рівняння називається оберненим інтерполюванням.

Якщо ж функція $f(x)$ не монотонна, то її замінюють інтерполяційним многочленом $P_{n-1}(x; f)$ і замість рівняння $f(x) = y$ розв'язують рівняння $P_{n-1}(x; f) = y$.

Використання підпрограм. Зауважимо, що при формуванні другого стовпця таблиці розділених різниць (перший задано!) функція $f(x_n)$ використовується лише при обчисленні функції $f(x_{n-1}; x_n)$, а далі в процесі формування таблиці розділених різниць вона не застосовується. Аналогічно за допомогою функції $f(x_{n-1})$ обчислюється лише функція $f(x_{n-2}; x_{n-1})$ і т. д. Тому після обчислення функції $f(x_{n-1}; x_n)$ її значення можна розмістити на місці функції $f(x_n)$, значення функції $f(x_{n-2}; x_{n-1})$ — на місці функції $f(x_{n-1})$ і т. д. Це саме стосується й інших стовпців таблиці розділених різниць, завдяки чому всі проміжні та кінцевий результати — поли-

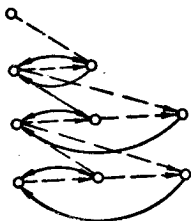


Рис. 4

лу діагональ цієї таблиці можна розмістити в одному масиві, що здійснено в підпрограмі DD.

Схему руху інформації в масиві Y, який спочатку містить функції $f(x_1), \dots, f(x_n)$, зображено на рис. 4.

▼ Підпрограма GOR призначена для обчислення значення інтерполяційного многочлена вигляду (5) у точці x за заданими масивами $\{d_i\}_{i=1,s}$ та $\{x_i\}_{i=1,s}$. В ній реалізовано рекурентне співвідношення (6). Використання підпрограм DD і GOR проілюструємо на прикладі розв'язування такої задачі.

За допомогою інтерполяційних многочленів по рівновіддалених вузлах $x_i = -1 + 2(i-1)/(n-1), i = \bar{1}, n$ і вузлах Чебишова $x_i^{(1)} = \cos(2i-1)\pi/(2n), i = \bar{1}, n$ обчислимо значення функції

$$f_m(x) = \begin{cases} 0, & x \in [-1, 0]; \\ x^m, & x \in [0, 1] \end{cases}$$

(m — натуральне число) в точці $x = 0,5$.

Головна програма та підпрограма-функція, що описує функцію $f_m(x)$, мають вигляд:

```
С ГОЛОВНА ПРОГРАМА, ЩО ІЛЮСТРУЄ ІНТЕРПОЛЮВАННЯ С
С ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ І ВУЗЛАХ ЧЕБИШОВА С
С ГОЛОВНА ПРОГРАМА, ЩО ІЛЮСТРУЄ ІНТЕРПОЛЮВАННЯ С
С ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ І ВУЗЛАХ ЧЕБИШОВА С
DIMENSION X(50),X1(50),Y(50),Y1(50)
LOGICAL*1 IND
10 TYPE 101
ACCEPT 102,N
TYPE 103
ACCEPT 102,M
С ОБЧИСЛЕННЯ МАСИВУ ВУЗЛІВ РІВНОМІРНОЇ СІТКИ
H=2./(N-1)
DO 20 I=1,N
20 X(I)=-1+(I-1)*H
С ОБЧИСЛЕННЯ МАСИВУ ВУЗЛІВ ЧЕБИШОВА
PI=4.*ATAN(1.)
DO 30 I=1,N
30 X1(N-I+1)=COS((2.*I-1) *PI/(2.*N))
С ОБЧИСЛЕННЯ МАСИВІВ ЗНАЧЕНЬ ФУНКЦІЇ
DO 40 I=1,N
Y(I)=F(M,X(I))
Y1(I) = F(M,X1(I))
40 CONTINUE
С ОБЧИСЛЕННЯ ІНТЕРПОЛЯЦІЙНОГО МНОГОЧЛЕНА В ТОЧЦІ Z
TYPE 104
ACCEPT 105,Z
CALL DD(X,Y,N)
CALL GOR(X,Y,Z,S,N)
CALL DD(X1,Y1,N)
CALL GOR(X1,Y1,Z,S1,N)
S2=F(M,Z)
WRITE(3,106)M,Z,S2,N,S1,S
TYPE 107
ACCEPT 108,IND
IF (IND.EQ.'Y') GO TO 10
STOP
101 FORMAT (IH$,' ВВЕДІТЬ КІЛЬКІСТЬ ВУЗЛІВ РІВНО',
* 'МІРНОЇ СІТКИ ->')
102 FORMAT (12)
103 FORMAT (IH$,' ВВЕДІТЬ СТЕПІНЬ ФУНКЦІЇ X**M:')
104 FORMAT (IH$,' В ЯКІЙ ТОЧЦІ ОБЧИСЛИТИ ІНТ.',
* 'МН-Н?')
105 FORMAT (E12.5)
106 FORMAT (' СТЕПІНЬ ФУНКЦІЇ X**M',I2//
* ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ',
* E12.5/ ДОРІВНЮЄ ',E12.5/
* 'КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ:',
* I4//
* 'ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ',
* 'ВУЗЛАХ: '/5X,E12.5/
* 'ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА: '/
* 5X,E12.5//)
107 FORMAT (IH$,' ПРОДОВЖИТИ (Y/N) ?')
```

```

108  FORMAT(A1)
      END

      FUNCTION F(M,X)
      F=0.
      IF(X.GT.0.) F=10.*X**M
      RETURN
      END

```

Головна програма в режимі діалога запитує про кількість вузлів сітки, степінь m і значення аргументу, при якому необхідно обчислити значення інтерполяційного полінома. Обчислення на ЕОМ дають такий результат:

СТЕПІНЬ ФУНКЦІЇ X**M2

```

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ .50000E+00
ДОРІВНЮЄ .25000E+00
КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ: 10
ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ:
.25185E+00
ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА:
.25097E+00

```

СТЕПІНЬ ФУНКЦІЇ X**M2

```

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ .50000E+00
ДОРІВНЮЄ .25000E+00
КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ: 20
ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ:
.24926E+00
ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА:
.24986E+00

```

СТЕПІНЬ ФУНКЦІЇ X**M2

```

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ .50000E+00
ДОРІВНЮЄ .25000E+00
КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ: 30
ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ:
.24923E+00
ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА:
.24991E+00

```

СТЕПІНЬ ФУНКЦІЇ X**M2

```

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ .50000E+00
ДОРІВНЮЄ .25000E+00
КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ: 40
ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ:
.27277E+00
ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА:
.24994E+00

```

СТЕПІНЬ ФУНКЦІЇ X**M2

```

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ .50000E+00
ДОРІВНЮЄ .25000E+00
КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ: 50

```

ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ:
.58217E+01

ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА:
.26919E+00

СТЕПІНЬ ФУНКЦІЇ $X^{**}M2$

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ В ТОЧЦІ .50000E+00

ДОРІВНЮЄ .25000E+00

КІЛЬКІСТЬ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ: 60

ЗНАЧ. ІНТ. ПОЛІНОМА ПО РІВНОВІДДАЛЕНИХ ВУЗЛАХ:
.10229E+03

ЗНАЧ. ІНТ. ПОЛІНОМА ПО ВУЗЛАХ ЧЕБИШОВА:
.45437E+00

Як бачимо, вже при 40 вузлах значення інтерполяційного многочлена по рівновіддалених вузлах значно відрізняються від точного значення функції. Виходячи з оцінки сталої Лебега, цього і слід було чекати. Більш того, інтерполяційний процес по рівновіддалених вузлах для цієї функції розбігається в усіх точках відрізка $[-1, 1]$, за винятком точок $-1, 0, 1$ (див. статтю *Рабкіна Е. Л., Шапиро Е. П.* Об одном расходящемся интерполяционном процессе // Изв. вузов. Сер. мат., 1971.— № 8.— С. 103—110).

Неважко бачити, що $f_m(x) \in C^{m-1}[-1, 1]$. За теоремою Джексона $E_{n-1}(f_2) \leq cn^{-1}$, $c = \text{const}$ і, оскільки для вузлів Чебишова $\Lambda_n \sim c \ln n$, інтерполяційний процес по вузлах Чебишова повинен збігатися при $n \rightarrow \infty$. Проте з розрахунків випливає, що вже при 50 вузлах значення функції, знайдене за допомогою інтерполяційного многочлена по вузлах Чебишова, гірше наближає значення функції, знайдене при меншій кількості вузлів. Тут починає даватися взнаки обумовленість задачі інтерполювання: через те що обчислення проводяться не точно і неусувна похибка пропорційна $\Lambda_n \sim c \ln n$, то зі збільшенням n погіршуються результати і для вузлів Чебишова.

▼ Підпрограма GDD призначена для моделювання процесу побудови таблиці розділених різниць з кратними вузлами — стовпець за стовпцем аналогічно підпрограмі DD.

Якщо вузол x_i є кратним, то в масиві Y елемент з номером i містить значення функції $f(x_i)$, елемент з номером $(i + 1)$ — значення похідної $f'(x_i)$ і т. д. При формуванні k -го стовпця ($k = \overline{1, s}$, $s =$

$= \sum_{i=1}^n m_i$) таблиці розділених різниць з кратними вузлами на i -те місце

($i = s, s - 1, \dots, k$) треба записати різницю i -го та $(i - k)$ -го елементів попереднього стовпця, розділену на $x_i - x_{i-k}$, якщо $x_i - x_{i-k} \neq 0$, або значення $f^{(k)}(x_i)/k!$, якщо $x_i - x_{i-k} = 0$. Тому головною трудністю є відшукання та розміщення відповідної інформації в масиві Y .

Для подолання цієї трудності вводиться робочий масив $M1$ розмірністю s , який називається *масивом вказівок*. Спочатку в ньому записують номери індексів (з масиву Y) елементів першого стовпця таблиці розділених різниць з кратними вузлами, а всі похідні в масиві Y ділять на відповідні факторіали. Після закінчення чергового циклу по k ($k + 1$

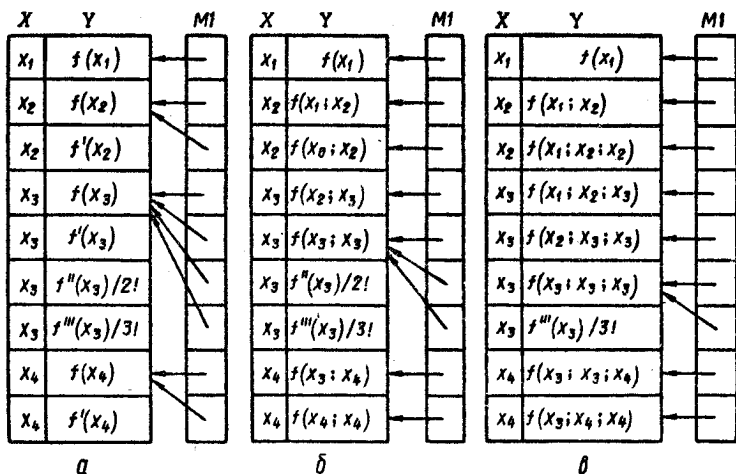


Рис. 5

вказує номер стовпця таблиці розділених різниць, який в цей момент обчислюється) масив $M1$ міститиме вказівки на елементи $(k + 1)$ -го стовпця цієї таблиці з кратними вузлами в масиві Y . Графічно цей процес можна відобразити так, як показано на рис. 5, а—в.

Таким чином, процес формування таблиці (5) йде так само, як і процес формування звичайної таблиці розділених різниць, але замість індексів масиву Y застосовуються значення цілочисельного масиву $M1$ з відповідними індексами. Зауважимо, що параметри масиву M

Таблиця 7

Момент часу	Координата	Швидкість	Прискорення
0	3	1	—
1	0	—	—
2	1	2	1
3	5	—	—

(кратностей вузлів) і масиву $M1$ (кількості вузлів інтерполювання без урахування кратності) використовуються в підпрограмі GDD лише для початкового встановлення масивів Y , $M1$ і тому без них можна легко обійтись, замінивши перші вісім її операторів.

▼ Підпрограма PER призначена для перетворення масиву d_i , $i = \overline{1, s}$ в ньютонівому запису многочлена (6) у масив d'_i , $i = \overline{1, s}$ при запису (8) того самого многочлена. Вона

реалізує рекурентні формули, що пов'язують між собою коефіцієнти многочленів з послідовності

$$Q_i(x) = Q_{i-1}(x)(x - x'_{s-i}) + d_{s-i}, \quad i = \overline{1, s-1};$$

$$Q_0(x) = d_s; \quad Q_{s-1}(x) = P_{s-1}(x).$$

Використання підпрограм GDD, GOR і PER проілюструємо на прикладі розв'язування такої задачі.

Маніпулятор, який має один ступінь вільності (тобто рухається по прямій), повинен рухатись за законом відповідно до табл. 7.

Апроксимуючи траєкторію руху маніпулятора многочленом, треба обчислити координату його положення в момент часу $x = 1,5$.

Щоб розв'язати задачу, потрібно побудувати інтерполяційний многочлен Ерміта й обчислити його значення в точці 1,5. Вузол $x_1 = 0$ має кратність 2, вузол $x_3 = 2$ — кратність 3, інші вузли є однократними. Головна програма, що ілюструє інтерполювання по кратних вузлах, має вигляд:

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА, ЩО ІЛЮСТРУЄ ІНТЕРПОЛЮВАННЯ                               С
С ПО КРАТНИХ ВУЗЛАХ                                                         С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DIMENSION X(7),Y(7),M(4),M1(7)
      DATA X/2*0.,1.,3*2.,3./,
      *      Y/3.,1.,0.,2.,1.,5./,
      *      M/2,1,3,1/
      N=7
      N1=4
      T=1.5
С ВИКЛИК ПІДПРОГРАМИ ОБЧИСЛЕННЯ ТАБЛИЦІ УЗАГАЛЬНЕНИХ
С РОЗДІЛЕНИХ РІЗНИЦЬ
      CALL GDD(X,Y,M,M1,N,N1)
С ВИКЛИК ПІДПРОГРАМИ ОБЧИСЛЕННЯ ЗНАЧЕННЯ
С ІНТЕРПОЛЯЦІЙНОГО МНОГОЧЛЕНА В ТОЧЦІ
      CALL GOR(X,Y,T,S,N)
      WRITE(4,101)T,S
С ВИКЛИК ПІДПРОГРАМИ ПЕРЕТВОРЕННЯ МАСИВУ РОЗДІЛЕНИХ
С РІЗНИЦЬ У МАСИВ КОЕФІЦІЄНТІВ ЗА СТЕПЕНЯМИ X
      CALL PER(X,Y,N)
      WRITE(4,102)Y
      STOP
101  FORMAT(' ЗНАЧЕННЯ ІНТЕРП. ПОЛІНОМА В',
      *      ' ТОЧЦІ T=',E12.5/ ' ДОРІВНЮЄ',
      *      E12.5//)
102  FORMAT(' МАСИВ КОЕФІЦІЄНТІВ ІНТЕРП. ПОЛІНОМА'
      *      '/ЗА ЗРОСТАЮЧИМИ СТЕПЕНЯМИ X'//
      *      (4E12.5))
      END
```

Обчислення на ЕОМ дають такий результат:
ЗНАЧЕННЯ ІНТЕРП. ПОЛІНОМА В ТОЧЦІ T= 0.15000E+01
ДОРІВНЮЄ 0.13086E+00

МАСИВ КОЕФІЦІЄНТІВ ІНТЕРП. ПОЛІНОМА
ЗА ЗРОСТАЮЧИМИ СТЕПЕНЯМИ X

0.30000E+01 0.10000E+01 -0.15444E+02 0.18861E+02
0.93750E+01 0.21389E+01 -0.18056E+00

▼ На практиці використовують інтерполяційні многочлени невисокого степеня (причини зазначено вище), а для досягнення потрібної точності інтерполювання відповідно вибираються відстань між вузлами (крок сітки h) і розміщення їх навколо точки x , в якій проводиться інтерполювання. В разі віддаленої від вузлів точки x точність інтер-

полювання буде більшою для середніх відрізків $[x_i, x_{i+1}]$ і меншою для крайніх [12, с. 37, 38].

За допомогою інтерполяційного многочлена обчислимо значення функції $f(x) = \frac{1}{x}$ у точці $x = 1,75$ з точністю $\varepsilon = 10^{-4}$ і знайдемо корінь рівняння $f(x) = y$, $y = 1,982$.

Вибравши інтерполяційний многочлен четвертого степеня, на підставі (10) знайдемо крок h рівномірної сітки для досягнення заданої точності. Маємо: $n = 5$, $f^{(5)}(x) = -4!/x^6$. Відрізком $[a, b]$, що містить точки інтерполювання, вважатимемо відрізок $[1,5; 4]$. Тоді $M_5 = \max_{x \in [a,b]} |f^{(5)}(x)| = \frac{256}{81}$. Тому нерівність (12) набуває вигляду $256(4h)^5/81 \leq 5! \varepsilon$, звідки $h \leq 10^{-1}$.

Головна програма до підпрограм DD та GOR має вигляд:

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
G ГОЛОВНА ПРОГРАМА ДО ПІДПРОГРАМ C
G DD, GOR C
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
DIMENSION X(5),Y(5),Y1(5)
H=0.1
XX=1.75
DO 10 I=1,5
    X(I)=1.5+I*H
    Y(I)=ALOG(X(I))
10 Y1(I)=Y(I)
WRITE(2,101)((X(I),Y(I)),I=1,5)
CALL DD(X,Y,5)
CALL GOR(X,Y,XX,S,5)
S1=ALOG(1.75)
WRITE(2,102)X,X,S,S1
CALL DD(Y1,X,5)
CALL GOR(Y1,X,1.982,S,5)
WRITE(2,103)S
S1=EXP(1.982)
WRITE(2,104)S1
STOP
101 FORMAT('МАСИВИ X I Y:/(2E12.5))
102 FORMAT('У ТОЧЦІ XX=' ,E10.3,'НАБЛ.',
* 'ЗНАЧ.Ф—П=' ,E10.3/21X,
* 'ТОЧ. ЗНАЧ.Ф—П=' ,E10.3)
103 FORMAT('НАБЛИЖЕНЕ ЗНАЧ. КОРЕНЯ ДОРІВНЮЄ',
* E10.3)
104 FORMAT('ТОЧНЕ ЗНАЧЕННЯ КОРЕНЯ ДОРІВНЮЄ',
* E10.3)
END

```

Обчислення на ЕОМ дають такий результат:

```

МАСИВИ X I Y:
0.16000E+01 0.47000E+00
0.17000E+01 0.53063E+00
0.18000E+01 0.58779E+00
0.19000E+01 0.64185E+00
0.20000E+01 0.69315E+00

```

У ТОЧЦІ $XX = 0.175E+01$ НАБЛ. ЗНАЧ.Ф— $\Pi = 0.560E+00$
 ТОЧ. ЗНАЧ.Ф— $\Pi = 0.560E+00$
 НАБЛИЖЕНЕ ЗНАЧ. КОРЕНЯ ДОРІВНЮЄ $0.715E+01$
 ТОЧНЕ ЗНАЧЕННЯ КОРЕНЯ ДОРІВНЮЄ $0.726E+01$

Запитання для самоперевірки

1. Які умови існування та єдиності інтерполяційного многочлена за системою функцій $\{f_i(x)\}_{i=0}^n$? 2. Що таке система Чебишова? Чи є системи функцій $\{x^i\}_{i=0}^n$, $\{e^{\alpha_i x}\}_{i=0}^n$, де α_i — різні числа, системами Чебишова на довільному відрізку $[a, b] \in (-\infty, \infty)$? 3. Що таке розділені різниці та які їх основні властивості? 4. Що таке розділені різниці з кратними вузлами? Як заповнюється таблиця таких різниць? 5. Які форми запису алгебраїчних многочленів ви знаєте? Які основні етапи обчислення коефіцієнтів інтерполяційного многочлена у формі Ньютона? 6. Який алгоритм обчислення інтерполяційного многочлена, відмінний від схеми Горнера, ви знаєте? Як відрізняється він від цієї схеми за кількістю арифметичних операцій? 7. Які оцінки залишкового члена інтерполяційного многочлена ви знаєте? Що таке стала Лебега і від чого залежить її значення? 8. Як залежить значення залишкового члена інтерполяційного многочлена $R_n(x; f) = f(x) - P_n(x; f)$ від положення точки x ? 9. Як впливає стала Лебега на швидкість рівномірної збіжності інтерполяційного многочлена при збільшенні його степеня? 10. Як залежить значення неусувної похибки (похибки через неточність вхідних даних) від сталої Лебега? 11. Які практичні рекомендації можна дати щодо застосування інтерполяційних многочленів? 12. Які можна дати апостеріорні оцінки точності інтерполювання?

Контрольні завдання

1. Для функції $f(x) = e^{-kx} - mx$ обчислити значення інтерполяційного многочлена по п'яти вузлах в заданій точці x з точністю $\varepsilon = 10^{-4}$. За допомогою оберненого інтерполювання розв'язати рівняння $f(x) = y$. Значення k, m, y задаються викладачем.

2. Методом оберненого інтерполювання з точністю до $\varepsilon = 10^{-4}$ знайти найменший додатний корінь рівняння

$$\sin kx - bx = 0$$

при

$$b = 1 + ih, \quad k = \begin{cases} 2, & i \in [1, 5]; \\ 3, & i \in [6, 10], \end{cases}$$

де $i = \overline{1, 10}$ — номер варіанта, що задається викладачем.

3. Дослідити на ЕОМ вплив вибору вузлів (у разі рівномірного розподілу їх і розподілу в нулях многочлена Чебишова) при інтерполюванні аналітичної функції

$$f(x) = (1 + 25x^2)^{-1}, \quad x \in [-1, 1].$$

4. Для функції $f(x) = (kx)^2 \ln(jx)$ ($k, j \geq 1$), $x \in [1, 2]$ обчислити в заданій точці $x \in \{1, 2\}$, $x \neq x_i$ значення інтерполяційного многочлена Ерміта, побудованого за рівномірною сіткою вузлів кратністю 2 з кроком $h = 1/3$, якщо $x_i = 1 + (i - 1)h$, $i = \overline{1, 3}$. Інтерполяційний многочлен Ерміта записати по степенях x .

§ 7. Інтерполяційні та згладжуючі сплайни

Постановка задачі. Для функції $f(x)$, $x \in [a, b]$, значення якої відомо у вузлах $x_i \in [a, b]$, $i = \overline{1, n}$, побудувати інтерполяційний або згладжуючий кубічний сплайн. Знайти значення сплайна чи його похідних в одній або кількох точках.

Методичні вказівки. При розв'язуванні задачі наближення функції найчастіше використовують інтерполяційний многочлен. Однак збільшення числа вузлів інтерполювання не гарантує доброго наближення інтерполяційного многочлена до відповідної функції. Щоб досить добре наблизити функцію, замість побудови інтерполяційного многочлена високого степеня застосовується інтерполяція кусковими многочленами на кожній з частин відрізка $[a, b]$.

Вимога гладкого спряження многочлена на сусідніх відрізках приводить до побудови кусково-многочленних функцій з однорідною структурою, які називаються *сплайнами*.

▼ Головні значення для чисельного аналізу мають сплайни невисокого степеня, тому обмежимося розглядом кубічних сплайнів.

Нехай на відрізку $[a, b]$ задано сітку

$$\Delta: a = x_1 < x_2 < \dots < x_n = b, \quad (1)$$

у вузлах якої відомо значення $y_i = f(x_i)$ функції $f(x)$. Задача кусково-кубічного інтерполювання формулюється так: на відрізку $[a, b]$ потрібно знайти функцію $S(f; x)$, що задовольняє умови:

- 1) $S(f; x) \in C^2[a, b]$;
- 2) $S(f; x) \in \pi_3$, $x \in [x_i, x_{i+1}]$, $i = \overline{1, n-1}$;
- 3) $S(f; x_i) = y_i$, $i = \overline{1, n}$, $n \geq 2$.

Неважко бачити, що кубічний сплайн визначається $4n - 4$ параметрами. Зазначені умови дають $4n - 6$ рівнянь для їх визначення. Таким чином, два параметри залишаються вільними і необхідно ввести два додаткових рівняння, щоб знайти їх.

Якщо функція $f(x) \in b - a$ періодичною, то сплайн $S(f; x)$ також повинен бути періодичним, мати неперервну першу похідну на відрізку $(-\infty, \infty)$, а точка $x = a$ не повинна бути вузлом сплайна.

Отже, періодичний сплайн $S(f; x)$ задовольняє умову

$$S^{(i)}(f; x_1 + 0) = S^{(i)}(f; x_n - 0), \quad i = 1, 2. \quad (*)$$

Іншими умовами, що найчастіше використовуються для побудови сплайна, є такі:

- а) $S'(f; a) = a_n$; $S'(f; b) = b_n$;
- б) $S''(f; a) = A_n$; $S''(f; b) = B_n$,

де a_n, b_n, A_n, B_n — задані дійсні числа, конкретний вибір яких залежить від розглядуваної задачі. Наприклад, якщо функція $f(x)$ має відповідні похідні, то можна покласти

$$a_n = f'(a), \quad b_n = f'(b), \quad A_n = f''(a), \quad B_n = f''(b)$$

або замінити ці числа наближеними значеннями відповідних похідних.

Якщо вибір крайових умов нелегкий, то в точках x_2, x_{n-1} сплайн повинен мати неперервну третю похідну, що еквівалентно умові

$$S'''(f; z-0) = S'''(f; z+0), \quad z = x_i, \quad i = 2, n-1. \quad (**)$$

Визначена вище функція $S(f; x)$ називається інтерполяційним кубічним сплайном. Якщо на кінцях відрізка $[a, b]$ задано однорідні крайові умови типу б), то такий кубічний сплайн називається *природним*.

Співно збіжності інтерполяційних сплайнів до неперервних функцій відзначимо, що в загальному випадку її немає. Тому накладаються додаткові обмеження на сітку.

Отже, нехай задано послідовність сіток

$$\Delta_n : a = x_1^{(n)} < x_2^{(n)} < \dots < x_n^{(n)} = b. \quad (2)$$

Покладемо

$$R(\Delta_n) = \|\Delta_n\| [\min_{1 \leq i \leq k_n-1} (x_{i+1}^{(n)} - x_i^{(n)})]^{-1}, \quad (3)$$

де

$$\|\Delta_n\| = \max_{1 \leq i \leq k_n-1} (x_{i+1}^{(n)} - x_i^{(n)}),$$

і припустимо, що існує таке число β ($1 \leq \beta < \infty$), при якому для будь-якого натурального числа n

$$R(\Delta_n) \leq \beta, \quad (4)$$

де β — абсолютна стала, а $\|\Delta_n\| \rightarrow 0$ при $n \rightarrow \infty$.

Нехай

$$L_n = L_n(f) = \max_{1 \leq k \leq k_n-1} |f(x_k, x_{k+1})|.$$

Сформулюємо таку теорему.

ТЕОРЕМА 1. Якщо функція $f(x) \in C^3[a, b]$, сплайн $S(f; x)$ інтерполює цю функцію у вузлах послідовності (2), виконується нерівність (4) і задовольняються крайові умови а), б), то

$$|f^{(s)}(x) - S^{(s)}(f; x)| \leq (1 + 3\beta) K_s \|\Delta_n\|^{3-s} \omega(f'', \|\Delta_n\|),$$

де ω — модуль неперервності. При цьому $k_4 = k_3 = 2k_2 = 4k_1 = 1$ і додатково припускається, що для крайових умов типу а)

$$a_n = f'(x_1); \quad b_n = f'(x_n),$$

а для крайових умов типу б)

$$A_n = f''(x_1); \quad B = f''(x_n).$$

Якщо виконуються крайові умови типу (**), то дійсною є така теорема.

ТЕОРЕМА 2. Якщо функція $f(x) \in W_\infty^4$ [1, с. 222], сітка рівномір-
на і

$$f'(x_0) = S'(f; x_1 + 0), \quad f'(x_n) = S'(f; x_n - 0),$$

то

$$|S^{(l)}(f; x) - f^{(l)}(x)| \leq \frac{32}{2^l(3-l)!} Mh^{4-l}, \quad l = \overline{0, 3},$$

де h — крок сітки.

Екстремальні властивості сплайна відображує така теорема [6, с. 147].

ТЕОРЕМА 3. Серед усіх функцій $f(x) \in W_2^2[a, b]$, що інтерполюють значення $y_i = f(x_i)$, кубічний сплайн $S(f; x)$ з крайовими умовами $S''(a) = S''(b) = 0$ мінімізує функціонал

$$I(f) = \int_a^b |f''(x)|^2 dx.$$

Якщо функція $f(x)$ належить до одного з класів функцій $\overline{W}_2^2[a, b]$ (періодичних з періодом $b - a$) чи $\overline{W}_2^2[a, b]$ (що задовольняють умови $f'(a) = a_n$, $f'(b) = b_n$), то мінімум функціонала забезпечує кубічний сплайн з тієї самої множини. В усіх випадках мінімізуюча функція є єдиною.

Нехай S — поліномний сплайн із класу W_∞^r (r — ціле), що інтерполює неперервну функцію. Розглядаючи сплайн $S(f; x)$ як оператор $S: C[1] \rightarrow C[1]$, можна показати, що [1, с. 227]

$$\|S\| = \frac{2}{\pi} \ln(r+1) + 0(1).$$

Як бачимо, норма оператора S на відміну від оператора $P: C \rightarrow C$, який ставить у відповідність функції $f(x) \in C[a, b]$ її інтерполяційний многочлен $P_n(f; x)$, не залежить від кількості вузлів n (див. § 6). Це означає, що «коефіцієнт підсилення» неусувної похибки в значеннях функції $f(x_i)$, $i = \overline{1, n}$, яким і є $\|S\|$, при обчисленні інтерполяційного сплайна в разі збільшення кількості вузлів залишається майже сталим і з цієї точки зору інтерполювання сплайнами має перевагу порівняно з інтерполюванням многочленами. Правда, точність такого наближення збільшується лише при зростанні гладкості функції до певної межі, а потім залишається обмеженою.

Метод наближення за допомогою нелокальних сплайнів — це метод з насиченням точності (відносно гладкості функції наближення). Класом насичення для зазначених сплайнів є клас функцій $W_2^3 [a, b]$.

Тому наближення сплайнами особливо ефективно для функцій невисокої гладкості.

▼ З алгоритмом побудови сплайна можна ознайомитись у [26, с. 86], де розглянуто сплайн, який реалізує крайові умови типу б) і **) [6, с. 97], де розглянуто крайові умови типу а).

Сплайн $S(f; x) = S(x)$ на відрізьку $[x_i, x_{i+1}]$ можна подати у вигляді

$$S(x) = \omega y_{i+1} + \bar{\omega} y_i + h_i^2 [(\omega^3 - \omega) \sigma_{i+1} + (\omega^3 - \bar{\omega}) \sigma_i], \quad (5)$$

де $a < x_1 < x_2 < \dots < x_n = b$; $y_i = f(x_i)$; $h_i = x_{i+1} - x_i$; $\omega = (x - x_i)/h_i$; $\bar{\omega} = 1 - \omega$; $\sigma_i = S''(x_i)/6$.

З урахуванням умов неперервності першої похідної у вузлах приходимо до системи $n - 2$ лінійних рівнянь відносно невідомих коефіцієнтів σ_i , $i = \overline{1, n}$. Використовуючи розділені різниці та крайові умови типу б), дістаємо систему n лінійних рівнянь з n невідомими:

$$\begin{cases} -h_1 \sigma_1 + h_1 \sigma_2 = h_1^2 \Delta_1^{(3)}; \\ a_i \sigma_{i-1} + b_i \sigma_i + c_i \sigma_{i+1} = f_i; \\ h_{n-1} \sigma_{n-1} - h_{n-1} \sigma_n = -h_{n-1} \Delta_{n-3}^{(3)}, \quad i = \overline{2, n-1}, \end{cases} \quad (6)$$

де

$$\Delta_i = (y_{i+1} - y_i)/h_i; \quad \Delta_i^{(n)} = \frac{\Delta_{i+1}^{(n-1)} - \Delta_i^{(n-1)}}{x_{i+n} - x_i};$$

$$a_i = h_{i-1}; \quad b_i = 2(h_{i-1} - h_i); \quad c_i = h_i; \quad f_i = \Delta_i - \Delta_{i-1}.$$

Якщо задано крайові умови типу а), то перше й останнє рівняння системи (6) мають вигляд

$$2h_1 \sigma_1 + h_1 \sigma_2 = \Delta_1 - f'(a);$$

$$h_{n-1} \sigma_{n-1} + 2h_{n-1} \sigma_n = f'(b) - \Delta_{n-1}.$$

Матриця знайденої системи рівнянь (позначимо її A) — тридіагональна, симетрична, невинроджена при виборі $x_1 < x_2 < x_3 < \dots < x_n$ і має діагональну перевагу. Отже, єдиний розв'язок системи завжди існує.

Оскільки число обумовленості матриці системи рівнянь (6) $\text{cond}(A) \leq 3$ [1, с. 222], така низька межа цього числа приводить до того, що при розв'язуванні системи рівнянь (6) вплив похибки заокруглення буде дуже незначним.

Для розв'язування системи рівнянь (6) застосуємо метод прогонки. Розрахункові формули для визначення коефіцієнтів σ_i , $i = \overline{1, n}$ мають вигляд

$$\alpha_i \sigma_i + h_i \sigma_{i+1} = \beta_i, \quad i = \overline{1, n-1},$$

де

$$\alpha_1 = -h_1, \alpha_i = 2(h_{i-1} + h_i) - h_{i-1}^2/\alpha_{i-1}, \quad i = \overline{2, n-1};$$

$$\alpha_n = -h_{n-1} - \frac{h_{n-1}^2}{\alpha_{n-1}};$$

$$\beta_1 = h_1^2 \Delta_1^{(3)}, \beta_i = (\Delta_i - \Delta_{i-1}) - \frac{h_{i-1} \beta_{i-1}}{\alpha_{i-1}}, \quad i = \overline{2, n-1};$$

$$\beta_n = -h_{n-1}^2 \Delta_{n-3}^{(3)} - h_{n-1} \beta_{n-1} / \alpha_{n-1};$$

$$\sigma_n = \beta_n / \alpha_n.$$

У випадку крайових вимог типу (*) дістаємо

$$\alpha_1 = 2h_1, \alpha_n = 2h_{n-1} - h_{n-1}^2/\alpha_{n-1};$$

$$\beta_1 = \Delta_1 - f'(a), \beta_n = f'(b) - \Delta_{n-1} - h_{n-1} \beta_{n-1} / \alpha_{n-1}.$$

Якщо сплайн треба обчислювати багато разів, то слід перегрупувати його члени. Значно зручніше визначати коефіцієнти b_i, c_i, d_i кубічного сплайна

$$S(x) = y_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3;$$

$$x \in [x_i, x_{i+1}], \quad i = \overline{1, n-1}. \quad (7)$$

Відповідні розрахункові формули мають вигляд

$$b_i = (y_{i+1} - y_i)/h_i - h_i(\sigma_{i+1} + 2\sigma_i);$$

$$c_i = 3\sigma_i;$$

$$d_i = (\sigma_{i+1} - \sigma_i)/h_i, \quad i = \overline{1, n-1}.$$

▼ Часто буває, що значення функції наближення у вузлах сітки $a = x_1 < x_2 < \dots < x_n = b$ відомо не точно, а з деякою похибкою. Тоді, якщо похибки $\{\varepsilon_k\}_{k=1}^n$ мають випадковий характер і в деяких точках можуть бути великими, застосування звичайної інтерполяції призводить до великої похибки, тобто інтерполююча функція значно відрізняється від дійсної функції $f(x)$. У цьому разі корисною буде інтерполяція сплайнами зі згладжуванням.

Щоб показати це, розглянемо функціонал

$$I_1(f) = \int_a^b |f''(x)|^2 dx + \sum_{k=1}^n p_k [f(x_k) - f_k]^2, \quad p_k > 0, \quad (8)$$

в якому скомбіновано інтерполяційні умови проходження кривої поблизу заданих значень функції та умови мінімальності згину кривої. Будемо вимагати, щоб шукана згладжуюча функція $g(x)$ мінімізувала в класі функцій $W_2^2[a, b]$ цей функціонал. Значення вагових коефі-

цієнтів дають змогу керувати процесом згладжування залежно від похибок ε_h чи їх розподілу, якщо останній відомо.

Відзначимо, що чим більші вагові коефіцієнти p_h , тим ближче до заданих значень проходить згладжуюча функція. При практичних розрахунках, якщо вихідну інформацію задано з похибкою ε_h , покладають $p_h = 1/\varepsilon_h$.

Можна довести, що коли $p_h \rightarrow \infty$, то згладжуючий сплайн рівномірно прямує до інтерполяційного.

Сформулюємо таку теорему.

ТЕОРЕМА 4. Серед функцій $f(x) \in W_2^2[a, b]$ кубічний сплайн, який задовольняє умови $S''(a) = S''(b) = 0$, мінімізує функціонал $I_1(f)$. Якщо функція $f(x)$ належить до одного з класів $W_2^2[a, b]$ або $\overline{W}_2^2[a, b]$, то мінімум функціонала забезпечує кубічний сплайн з того ж класу. В кожному випадку мінімізуюча функція є єдиною.

▼ Розв'язком варіаційної задачі (8) є кубічний сплайн [10, с. 92]. Оскільки він повністю визначається своїми значеннями $\{\mu_h\}$ у вузлах сітки, а (8) є функцією $\{\mu_h\}$, мінімізація функціонала (8) еквівалентна знаходженню мінімуму цієї функції, умова якого у векторній формі має вигляд

$$H^*m + P\mu = Pf, \quad (9)$$

де m — вектор значень другої похідної від сплайна; P — діагональна матриця розмірністю n з елементами $\{p_h\}$; H — прямокутна матриця розмірністю $(n-2) \times (n-1)$ з елементами

$$H_{i,i-1} = 1/h_i, \quad H_{ii} = -(1/h_i + 1/h_{i+1}),$$

$$H_{i,i+1} = 1/h_{i+1}, \quad i = \overline{1, n-1}; \quad h_i = x_{i+1} - x_i, \quad i = \overline{1, n-2}.$$

Домножаючи (9) зліва на HP^{-1} , дістаємо

$$HP^{-1}H^*m + Hm = Hf, \quad (10)$$

або з урахуванням того, що $H\mu = Am$, маємо

$$(A + HP^{-1}H^*)m = Hf, \quad (11)$$

де A — тридіагональна матриця з елементами

$$a_{ii} = (h_i + h_{i+1})/3, \quad a_{i,i+1} = h_{i+1}/6, \quad a_{i-1,i} = h_i/6, \quad i = \overline{1, n-2}.$$

Матриця $Q = A + HP^{-1}H^*$ системи рівнянь — п'ятидіагональна симетрична й додатно визначена.

▼ Для розв'язування системи рівнянь (11) застосуємо метод квадратного кореня. З цією метою запишемо матрицю Q у вигляді $Q = S^*S$, де S — верхня трикутна тридіагональна матриця.

Подавши систему рівнянь (11) у вигляді

$$S^*Sm = b,$$

позначимо

$$Sm = y. \quad (12)$$

Тоді дістанемо

$$S^*y = b. \quad (13)$$

Матриця S^* — нижня трикутна. Зі системи рівнянь (13) визначаємо вектор y , потім із (12) знаходимо $n - 2$ компоненти вектора m ; решта їх дорівнює нулю, бо розглядаємо природний сплайн.

Вектор згладжуючих значень μ обчислюємо так:

$$\mu = f - P^{-1}H^*m;$$

$$\mu_1 = f_1 - m_1/h_1p_1;$$

$$\mu_2 = f_2 + m_1(1/h_1 + 1/h_2)/p_2 - m_2/p_2h_2;$$

.....

$$\mu_i = f_j - m_{j-2}/p_jh_{j-1} + m_{j-1}/(1/h_{j-1} + 1/h_j)/p_j - m_j/p_jh_j,$$

$$3 \leq j \leq n - 2;$$

$$\mu_{n-1} = f_{n-1} - m_{n-3}/p_{n-1}h_{n-2} + m_{n-2}(1/h_{n-2} + 1/h_{n-1})/p_{n-1};$$

$$\mu_n = f_n - m_n/p_nh_{n-1}.$$

Згладжуючий сплайн має вигляд

$$g(x) = m_{i-1}(x_i - x)^3/6h_i + m_i(x - x_{i-1})^3/6h_i + (\mu_i + m_{i-1}h_i/6)^2(x - x_i)/h_i + (\mu_i - m_ih_i/6)(x - x_i)/h_i; \quad x \in [x_i, x_{i+1}],$$

$$i = 1, n - 1.$$

Перегрупувавши його члени за аналогією з (7), дістанемо

$$g(x) = \mu_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3, \quad x \in [x_i, x_{i+1}],$$

де

$$b_i = (\mu_{i+1} - \mu_i)/h_i - h_i(m_{i+1} - 2m_i)/6;$$

$$c_i = m_i/2; \quad d_i = (m_{i+1} - m_i)/6h_i, \quad i = 1, n - 1.$$

▼ Одним з поширених способів задавання кривої є параметричний, тобто у вигляді двох функцій $x = x(u)$ та $y = y(u)$ деякого додатного параметра (наприклад, довжини дуги). При інтерполюванні кривої, заданої параметрично, природно ввести розбиття інтервалу зміни параметра $u_1 < u_2 < \dots < u_n$, потім визначити відповідні значення координат точок на кривій $x = x(u_i)$, $y = y(u_i)$ та побудувати інтерполяційні сплайни.

Сукупність таких двох сплайнів називається *параметричним інтерполяційним сплайном*.

▼ Дамо деякі рекомендації щодо використання крайових умов при побудові сплайнів. Якщо крива замкнена, то природно викорис-

товувати періодичні умови типу (*), а якщо не замкнена, то зручно застосовувати крайові умови типу (**). Якщо відомо значення перших і других похідних функції $f(x)$ на кінцях інтервалу зміни параметра x , то можна використовувати крайові умови типу а) чи б). При цьому, якщо існує можливість вибору їх, то кращою є умова а).

Зупинимось на особливостях вибору вузлів інтерполяції. Перш за все слід звернути увагу на точки розриву похідних функції $f(x)$. Для знаходження кращих наближень функцій з кусково-неперервними третіми похідними точки розриву останніх мають бути включені до числа точок сітки.

Розглянемо тепер випадок, коли розрив має друга похідна функції $f(x)$. Якщо при інтерполюванні її сплайном не вжити допоміжних заходів, то в сплайні з'являються осциляції. Нехай точка ξ , в якій є розрив другої похідної функції $f(x)$, лежить на відрізку $[x_k, x_{k+1}]$. Тоді слід потурбуватися про те, щоб виконувалась умова $h_k = \gamma_{\min} \{h_{k-1}, h_{k+1}\}$, де γ — досить мала константа (наприклад, $\gamma = 0,01 \div 0,1$; на практиці можна покласти $\gamma = 0,01$), а $h_k = x_{k+1} - x_k$.

Нехай тепер розрив має перша похідна функції $f(x)$. По-перше, точку її розриву η включимо до числа вузлів (наприклад, $\eta = x_k$). По-друге, вузли x_{k-1}, x_{k+1} виберемо так, щоб $h_{k-1} = \gamma_1 h_{k-2}$, $h_k = \gamma_2 h_{k+1}$, де γ_1, γ_2 — малі числа. По-третє, в системі рівнянь для визначення моментів сплайна праву частину k -го рівняння прирівняємо до нуля.

І, нарешті, остання рекомендація стосується вибору вузлів сітки на прямолінійних ділянках кривої. На межі цих ділянок необхідно брати два близьких вузла, щоб позбутися осциляції.

Такі самі рекомендації щодо вибору вузлів інтерполювання можна дати і для побудови параметричних сплайнів, вважаючи однаковими крайові умови для них.

▼ Згладжуючий інтерполяційний сплайн застосовується при чисельному диференціюванні функції. На практиці для зменшення похибки диференціювання при цьому часто роблять так. За заданою таблицею значень функції $f(x)$ будують кубічний сплайн $S(f; x)$, а потім його диференціюють. Або попередньо «згладжують» вихідні дані за допомогою згладжуючого сплайна, після чого за згладженими значеннями обчислюють похідну.

Приклади використання підпрограм. **Приклад 1.** Користуючись підпрограмами SPLINE та SEVAL, побудуємо інтерполяційний сплайн для функції

$$f(x) = \begin{cases} 0, & x \in [0, 1]; \\ x^r - 1, & x \in (1, 2], \end{cases}$$

після чого обчислимо його значення в точках $x_i = ih, i = 2/m, m = 17$ при кількості вузлів інтерполювання $n = 10; 20; 30; 40$.

Головна програма, в якій для розв'язування цієї задачі застосовуються підпрограма SPLINE та підпрограма-функція SEVAL, має вигляд:

С ГОЛОВНА ПРОГРАМА ОБЧИСЛЕННЯ КОЕФІЦІЄНТІВ ІНТЕРПО- С
С ЛЯЦІЙНОГО СПЛАЙНА ТА ЙОГО ЗНАЧЕНЬ У М ТОЧКАХ С
С

```
DIMENSION X(50),Y(50),Z(50),U(50),V(50),
*          B(50),C(50),D(50),Y1(50)
TYPE 101
ACCEPT *,N
TYPE 102
ACCEPT *,M
TYPE 103
ACCEPT *,NW
TYPE 104
ACCEPT *,R
WRITE(NW,105)N,M,R
H=2./N
H1=2./M
NP1=N+1
DO 10 I=1,NP1
    X(I)=H*FLOAT(I-1)
    Y(I)=F(X(I),R)
10    CONTINUE
CALL SPLINE(NP1,X,Y,3,0.,0.,B,C,D)
MP1=M+1
DO 20 J=1,MP1
    Z(J)=H1*FLOAT(J-1)
    V(J)=SEVAL(NP1,Z(J),X,Y,B,C,D)
    Y1(J)=F(Z(J),R)
20    CONTINUE
WRITE(NW,106)(V(J),J=1,MP1)
DO 30 J=1,MP1
    Y1(J)=ABS(Y1(J)-V(J))
30    CONTINUE
WRITE(NW,107)(Y1(J),J=1,MP1)
STOP
101  FORMAT(1H$,' N=')
102  FORMAT(1H$,' M=')
103  LORMAT(1H$,' NW=')
104  FORMAT(1H$,' R=')
105  FORMAT(7X,' КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=',
*      13,/7X' КІЛЬКІСТЬ ТОЧОК, У ЯКИХ',
*      ' ОБЧИСЛЮЄТ. СПЛАЙН, M=', 13,/7X,
*      ' ПОКАЗНИК СТЕПЕНЯ ФУНКЦІЇ R=',E13.5)
106  FORMAT(7X,' МАСИВ ЗНАЧЕНЬ ІНТЕРПОЛ.СПЛАЙНА',
*      /9(4X,4E13.5))
107  FORMAT(7X,' АБСОЛЮТНА ПОХИБКА',/9(4X,4E13.5))
END

FUNCTION F(X,R)
F=0.
IF (X.GE.1.)F=X**R-1.
RETURN
END
```

Обчислення на комп'ютері ДВК-2М дають такий результат:

КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=10
КІЛЬКІСТЬ ТОЧОК, У ЯКИХ ОБЧИСЛЮЄТ. СПЛАЙН, M=17
ПОКАЗНИК СТЕПЕНЯ ФУНКЦІЇ R=0.20000E+01
МАСИВ ЗНАЧЕНЬ ІНТЕРПОЛ. СПЛАЙНА

0.00000E+00	-0.33490E-03	0.27186E-03	0.62928E-03
-0.18389E-02	-0.80601E-03	0.92601E-02	-0.77759E-02
-0.34464E-01	0.90406E-01	0.37716E+00	0.68299E+00
0.99236E+00	0.13375E+01	0.17134E+01	0.21144E+01
0.25430E+01	0.30000E+01		

АБСОЛЮТНА ПОХИБКА

0.00000E+00	0.33490E-03	0.27186E-03	0.62928E-03
0.18389E-02	0.80601E-03	0.92601E-02	0.77759E-02
0.34464E-01	0.30701E-01	0.69270E-02	0.82490E-02
0.71800E-03	0.16381E-02	0.56052E-03	0.24152E-03
0.29922E-03	0.00000E+00		

КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=20
КІЛЬКІСТЬ ТОЧОК, У ЯКИХ ОБЧИСЛЮЄТ. СПЛАЙН, M=17
ПОКАЗНИК СТЕПЕНЯ ФУНКЦІЇ R=0.20000E+01
МАСИВ ЗНАЧЕНЬ ІНТЕРПОЛ. СПЛАЙНА

0.00000E+00	-0.18263E-06	0.12353E-05	-0.62206E-05
0.23152E-04	-0.50313E-04	-0.12719E-03	0.21836E-02
-0.14189E-01	0.10771E+00	0.38614E+00	0.67462E+00
0.99303E+00	0.13391E+01	0.17128E+01	0.21142E+01
0.25433E+01	0.30000E+01		

АБСОЛЮТНА ПОХИБКА

0.00000E+00	0.18263E-06	0.12353E-05	0.62206E-05
0.23152E-04	0.50313E-04	0.12719E-03	0.21836E-02
0.14189E-01	0.13393E-01	0.20610E-02	0.12040E-03
0.47445E-04	0.21815E-04	0.58413E-05	0.23842E-06
0.23842E-06	0.00000E+00		

КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=30
КІЛЬКІСТЬ ТОЧОК, У ЯКИХ ОБЧИСЛЮЄТ. СПЛАЙН, M=17
ПОКАЗНИК СТЕПЕНЯ ФУНКЦІЇ R=0.20000E+01
МАСИВ ЗНАЧЕНЬ ІНТЕРПОЛ. СПЛАЙНА

0.00000E+00	0.38561E-09	0.56745E-08	0.48539E-07
0.11602E-06	-0.31826E-05	-0.59266E-04	-0.58639E-03
-0.24975E-02	0.11870E+00	0.38352E+00	0.67468E+00
0.99308E+00	0.13391E+01	0.17128E+01	0.21142E+01
0.25433E+01	0.30000E+01		

АБСОЛЮТНА ПОХИБКА

0.00000E+00	0.38561E-09	0.56745E-08	0.48539E-07
0.11602E-06	0.31826E-05	0.59266E-04	0.58639E-03
0.24975E-02	0.24031E-02	0.56428E-03	0.57042E-04
0.31590E-05	0.23842E-06	0.23842E-06	0.47684E-06
0.23842E-06	0.00000E+00		

КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=40
КІЛЬКІСТЬ ТОЧОК, У ЯКИХ ОБЧИСЛЮЄТ. СПЛАЙН, M=17
ПОКАЗНИК СТЕПЕНЯ ФУНКЦІЇ R=0.20000E+01
МАСИВ ЗНАЧЕНЬ ІНТЕРПОЛ. СПЛАЙНА

0.00000E+00	0.11618E-11	0.21774E-10	-0.11963E-09
-0.13345E-07	-0.20073E-06	0.25750E-05	0.14889E-03
0.17200E-02	0.12278E+00	0.38423E+00	0.67474E+00

0.99308E+00 0.13391E+01 0.17128E+01 -0.21142E+01
 0.25433E+01 0.30000E+01

АБСОЛЮТНА ПОХИБКА

0.00000E+00 0.11618E-11 0.21774E-10 0.11963E-09
 0.13345E-07 0.20073E-06 0.25750E-05 0.14889E-03
 0.17200E-02 0.16711E-02 0.14466E-03 0.21458E-05
 0.17881E-06 0.35763E-06 0.35763E-06 0.71526E-06
 0.23842E-06 0.00000E+00

Як бачимо, на відміну від інтерполювання многочленами точність наближень сплайнами зі збільшенням кількості вузлів інтерполяції поліпшується, оскільки стала Лебега при інтерполюванні сплайнами залишається обмеженою.

Приклад 2. За допомогою підпрограм SGL і SEVAL побудуємо згладжуючий сплайн для такої таблично заданої функції $f(x)$:

x	0	0,1	0,2	0,3
$f(x)$	0,5	0,3	0,7	0,6

Обчислимо також значення сплайна та його першої похідної в точці $u = 0,15$. Вагові коефіцієнти вважатимемо однаковими, а саме: $p_h = 100$, $k = 1,4$.

Головна програма, в якій для розв'язування розглядуваної задачі застосовуються підпрограма SGL і підпрограма-функція SEVAL, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ОБЧИСЛЕННЯ ПАРАМЕТРІВ                               C
C ЗГЛАДЖУЮЧОГО СПЛАЙНА, ЙОГО ЗНАЧЕННЯ ТА                               C
C ПЕРШОЇ ПОХІДНОЇ В ТОЧЦІ U                                           C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  REAL X(4),P(4),F(4),MU(4),B(4),C(4),D(4)
  DATA X/0.,0.1,0.2,0.3/,
  *      F/0.5,0.3,0.7,0.6/,P/4*100./
  WRITE(4,101)
  WRITE(4,102)X,F
C  ОБЧИСЛЕННЯ КОЕФІЦІЕНТІВ ЗГЛАДЖУЮЧОГО СПЛАЙНА
  CALL SGL(3,P,X,F,MU,B,C,D)
  WRITE(4,103)
  WRITE(4,104)MU
  WRITE(4,105)
  WRITE(4,106)B,C,D
  U=0.15
C  ОБЧИСЛЕННЯ ЗНАЧЕННЯ СПЛАЙНА В ЗАДАНИЙ ТОЧЦІ U
  S=SEVAL(4,U,X,MU,B,C,D)
  WRITE(4,107)
  WRITE(4,108)U,S
C  ОБЧИСЛЕННЯ КОЕФІЦІЕНТІВ СПЛАЙНА ДЛЯ ВИЗНАЧЕННЯ
C  ПОХІДНОЇ
  DO 10 I=1,4
    MU(I)=B(I)
    B(I)=2*C(I)
    C(I)=3*D(I)
    D(I)=0.
  10  CONTINUE
C  ОБЧИСЛЕННЯ ЗНАЧЕННЯ ПОХІДНОЇ ВІД СПЛАЙНА В ЗАДАНИЙ
C  ТОЧЦІ U
  S1=SEVAL(4,U,X,MU,B,C,D)
  WRITE(4,109)
  WRITE(4,110)U,S1

```

```

STOP
101 FORMAT(/6X,'ЗНАЧЕННЯ ФУНКЦІ F У ТОЧКАХ X')
102 FORMAT(3X,'X=',4E12.5/3X,'F=',4E12.5)
103 FORMAT(/6X,'ЗГЛАДЖЕНІ ЗНАЧЕННЯ ФУНКЦІ F У',
*       'ТОЧКАХ X')
104 FORMAT(3X,'MU=',4E12.5)
105 FORMAT(/6X,'КОЕФІЦІЕНТИ ЗГЛАДЖУЮЧОГО СПЛАЙНА')
106 FORMAT(3X,'B=',4E12.5/3X,'C=',4E12.5/
*       3X,'D=',4E12.5)
107 FORMAT(/6X,'ЗНАЧЕННЯ ЗГЛАДЖУЮЧОГО СПЛАЙНА В-',
*       'ТОЧЦІ U')
108 FORMAT(6X,'U=',E12.5,4X,'S=',E12.5)
109 FORMAT(/6X,'ЗНАЧЕННЯ ПОХІДНОЇ ВІД СПЛАЙНА В',
*       'ТОЧЦІ U')
110 FORMAT(6X,'U=',E12.5,4X,'S1=',E12.5)
END

```

Обчислення на комп'ютері ДВК-2М дають такий результат:

ЗНАЧЕННЯ ФУНКЦІ F У ТОЧКАХ X

```

X=0.00000E+00  0.10000E+00  0.20000E+00  0.30000E+00
F=0.50000E+00  0.30000E+00  0.70000E+00  0.60000E+00

```

ЗГЛАДЖЕНІ ЗНАЧЕННЯ ФУНКЦІ F У ТОЧКАХ X

```

MU=0.42127E+00  0.48818E+00  0.55982E+00  0.63073E+00

```

КОЕФІЦІЕНТИ ЗГЛАДЖУЮЧОГО СПЛАЙНА

```

B=0.65593E+00  0.69530E+00  0.71930E+00  0.71930E+00
C=0.00000E+00  0.39363E+00 -0.15363E+00 -0.15363E+00
D=0.13121E+01 -0.18242E+01  0.51211E+00  0.51211E+00

```

ЗНАЧЕННЯ ЗГЛАДЖУЮЧОГО СПЛАЙНА В ТОЧЦІ U

```

U=0.15000E+00  S=0.52370E+00

```

ЗНАЧЕННЯ ПОХІДНОЇ ВІД СПЛАЙНА В ТОЧЦІ U

```

U=0.15000E+00  S1=0.72098E+00

```

Приклад 3. Користуючись підпрограмами SPLINE та SEVAL, побудуємо параметричний сплайн для функції

$$x = a \cos^3 t; \quad y = a \sin^3 t, \quad 0 \leq t \leq \pi/2,$$

після чого обчислимо його значення в точках $m = 25$ при кількості m вузлів значно меншій (наприклад, $n = 10$).

Головна програма, в якій для розв'язування цієї задачі застосовуються підпрограма SPLINE та підпрограма-функція SEVAL, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА ВИКОРИСТАННЯ ПІДПРОГРАМ                                     С
С SPLINE ТА SEVAL ДЛЯ ПОБУДОВИ ПАРАМЕТРИЧНОГО                             С
С СПЛАЙНА                                                                     С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      DIMENSION X(50),Y(50),T(50),Y1(50),Y2(50),
*              V1(50),V2(50),U(50),B1(50),C1(50),
*              D1(50),B2(50),C2(50),D2(50),V(150)
С КІЛЬКІСТЬ ТОЧОК ПО ОСІ АБСЦИС
      TYPE 101
      ACCEPT *,N
      TYPE 102

```

```

ACCEPT *,M
TYPE 103
ACCEPT *,A
TYPE 104
ACCEPT *,NW
WRITE(NW,105)N,M,A
H=3.1415/(2.*N)
H1=3.1415/(2.*M)
NP1=N+1
MP1=M+1
DO 10 I=1,NP1
    T(I)=H*FLOAT(I-1)
    AA=T(I)
    AB=COS(AA)
    AC=SIN(AA)
    X(I)=A*AB**3
    Y(I)=A*AC**3
    CONTINUE
10 CALL SPLINE(NP1,T,X,3,0,0,B1,C1,D1)
   CALL SPLINE(NP1,T,Y,3,0,0,B2,C2,D2)
   DO 20 I=1,MP1
       U(I)=H1*FLOAT(I-1)
       AA=U(I)
       Y1(I)=A*(COS(AA))**3
       Y2(I)=A*(SIN(AA))**3
20   CONTINUE
   DO 30 I=1,MP1
       V1(I)=SEVAL(NP1,U(I),T,X,B1,C1,D1)
       V2(I)=SEVAL(NP1,U(I),T,Y,B2,C2,D2)
30   CONTINUE
   WRITE(NW,106)(Y1(I),I=1,MP1)
   WRITE(NW,107)(Y2(I),I=1,MP1)
   WRITE(NW,108)(V1(I),I=1,MP1)
   WRITE(NW,109)(V2(I),I=1,MP1)
   DO 40 I=1,MP1
       V(I)=V1(MP1-I+1)
       V(I+MP1)=V2(MP1-I+1)
40   CONTINUE
   WRITE(NW,110)
   CALL GRAPH(NW,1,MP1,2,2*MP1,V)
   STOP
101  FORMAT(1H$, 'N=')
102  FORMAT(1H$, 'M=')
103  FORMAT(1H$, 'A=')
104  FORMAT(1H$, 'NW=')
105  FORMAT(4X, 'КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=',
*      12, /4X, 'КІЛЬК. ТОЧОК, У ЯКИХ ОБЧИСЛ.',
*      'ПАРАМ. СПЛАЙН, M=', /4X, 'ЗНАЧЕННЯ',
*      'ПАРАМЕТРА A=', E12.5)
106  FORMAT(4X, 'ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ X(U) Y1=',
*      (7(/4X, 4E12.5)))
107  FORMAT(4X, 'ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ Y(U) Y2=',
*      (7(/4X, 4E12.5)))
108  FORMAT(4X, 'ЗНАЧЕННЯ СПЛАЙНА ДЛЯ X(U) V1=',
*      (7(/4X, 4E12.5)))

```

109 FORMAT(4X,'ЗНАЧЕННЯ СПЛАЙНА ДЛЯ Y(U) V2=',
*(7/(4X,4E12.5)))

110 FORMAT(14X,'ГРАФІК ЗНАЧЕНЬ СПЛАЙНА')
END

Обчислення на комп'ютері ДВК-2М дають такий результат:

КІЛЬКІСТЬ ТОЧОК ІНТЕРПОЛЯЦІЇ N=10
КІЛЬК. ТОЧОК, У ЯКИХ ОБЧИСЛ. ПАРАМ. СПЛАЙН, M=25
ЗНАЧЕННЯ ПАРАМЕТРА A=0.20000E+01

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ X(U) Y1=
0.20000E+01 0.19882E+01 0.19531E+01 0.18956E+01
0.18174E+01 0.17205E+01 0.16076E+01 0.14816E+01
0.13459E+01 0.12039E+01 0.10591E+01 0.91494E+00
0.77479E+00 0.64161E+00 0.51803E+00 0.40620E+00
0.30773E+00 0.22366E+00 0.15441E+00 0.99800E-01
0.59037E-01 0.30775E-01 0.13167E-01 0.39416E-02
0.49617E-03 0.19788E-12

ТОЧНЕ ЗНАЧЕННЯ ФУНКЦІЇ Y(U) Y2=
0.00000E+00 0.49508E-03 0.39372E-02 0.13157E-01
0.30759E-01 0.59012E-01 0.99765E-01 0.15436E+00
0.22360E+00 0.30766E+00 0.40612E+00 0.51794E+00
0.64152E+00 0.77469E+00 0.91483E+00 0.10590E+01
0.12038E+01 0.13458E+01 0.14815E+01 0.16075E+01
0.17204E+01 0.18173E+01 0.18956E+01 0.19530E+01
0.19882E+01 0.20000E+01

ЗНАЧЕННЯ СПЛАЙНА ДЛЯ X(U) V1=
0.20000E+01 0.19890E+01 0.19534E+01 0.18954E+01
0.18171E+01 0.17205E+01 0.16076E+01 0.14816E+01
0.13459E+01 0.12039E+01 0.10591E+01 0.91498E+00
0.77480E+00 0.64164E+00 0.51808E+00 0.40620E+00
0.30779E+00 0.22369E+00 0.15442E+00 0.99823E-01
0.59037E-01 0.30916E-01 0.13294E-01 0.37544E-02
0.30516E-04 0.19788E-12

ЗНАЧЕННЯ СПЛАЙНА ДЛЯ Y(U) V2=
0.00000E+00 0.29564E-04 0.37502E-02 0.13284E-01
0.30899E-01 0.59012E-01 0.99788E-01 0.15438E+00
0.22363E+00 0.30773E+00 0.40612E+00 0.51799E+00
0.64154E+00 0.77470E+00 0.91487E+00 0.10590E+01
0.12037E+01 0.13458E+01 0.14815E+01 0.16075E+01
0.17204E+01 0.18171E+01 0.18954E+01 0.19533E+01
0.19889E+01 0.20000E+01

Графік значень сплайна наведено на с. 78.

Запитання для самоперевірки

1. Що таке згладжуючий інтерполяційний сплайн? 2. Коли та чому кусково-многочленне інтерполювання краще, ніж інтерполювання многочленами? 3. Які основні етапи алгоритму побудови інтерполяційного сплайна? 4. Які основні етапи алгоритму побудови згладжуючого сплайна? 5. Як побудувати інтерполяційний сплайн для функції, що має розривні похідні? 6. Який зв'язок між інтерполяційним і згладжуючим сплайнами? 7. Які екстремальні властивості мають інтерполяційні та згладжуючі сплайни?

ГРАФІК ЗНАЧЕНЬ СПЛАЙНА

ФУНКЦІЇ: 1 - *
АБСЦИСИ

ГРАФІК 1

0.1979E-12!
0.3052E-04!
0.3754E-02!
0.1329E-01!
0.3092E-01!
0.5904E-01!
0.9982E-01!
0.1544E+00!
0.2237E+00!
0.3078E+00!
0.4062E+00!
0.5181E+00!
0.6416E+00!
0.7748E+00!
0.9150E+00!
0.1059E+01!
0.1204E+01!
0.1346E+01!
0.1482E+01!
0.1608E+01!
0.1720E+01!
0.1817E+01!
0.1895E+01!
0.1953E+01!
0.1989E+01!
0.2000E+01!

ШКАЛА ОРДИНАТ
1 2 3 4 5 6 7 8 9 10 11
STEPY= 0.20000E+00

VMIN= 0.00000E+00 VMAX= 0.20000E+01

Контрольні завдання

1. Побудувати таблицю значень функції

$$f(x) = a_1 x^2 + b_1 \sin(c_1 x) + d_1 \cos(c_2 x) + a_2 \ln x, \quad x \in [a, b]$$

в точках $x_i = a + ih$, $i = \overline{1, n}$, $h = 1/n$. Обчислити значення цієї функції, а потім її першу та другу похідні в точках x_j з кроком $h/3$. Кількість точок n і крок h задаються викладачем.

2. Побудувати кубічний згладжуючий сплайн для функції

$$f(x) = a_1 (x^2)^\alpha + b_1 \sin(c_1 x) + d_1 \cos(\mu_1 x), \quad x \in [a, b].$$

Обчислити значення сплайна та його похідних в точці $x_1 \in [a, b]$. Коефіцієнти a_1 , b_1 , c_1 , d_1 , μ_1 , α , a , b , а також точка x_1 і крок дискретизації задаються викладачем. Усі коефіцієнти, крім α , можуть бути довільними. Якщо $\alpha < 0$, то d має бути відмінним від нуля.

3. За допомогою підпрограм SPLINE та SEVAL проінтерполювати функцію Рунге

$$f(x) = \frac{1}{1 + 25x^2}$$

в точках $x_i = -1; -0,9; \dots; 0,9; 1$ при їх кількості $n = 21$. Порівняти знайдені результати з інтерполяційним многочленом 20-го степеня.

4. Побудувати параметричний сплайн для функції

$$x = a(t - \sin t); \quad y = a(1 - \cos t), \quad 0 \leq t \leq 2\pi.$$

§ 8. Чисельне інтегрування

Постановка задачі. Обчислити інтеграл $I(f) = \int_a^b \rho(x) f(x) dx$ (де $\rho(x) > 0$ — задана вагова функція) зі заданим ступенем точності.

Методичні вказівки. Розрахункова формула [12, с. 85]

$$I_n(f) = \sum_{k=1}^m \sum_{i=1}^{\alpha_k} A_{k,i}^{(n+1)} f^{(i)}(x_k^{(n+1)}),$$

де $\sum_{k=1}^m \alpha_k = n + 1$; $x_k^{(n+1)} \in [a, b]$, $k = \overline{1, m}$ — вузли; $A_{k,i}^{(n+1)}$ — вагові коефіцієнти, називається *квadrатурною*.

Отже,

$$\int_a^b \rho(x) f(x) dx = I_n(f) + R_n(f),$$

де $R_n(f)$ — залишковий член квадратурної формули.

На практиці вживаються квадратурні формули без використання значень похідних функції у вузлах, тобто формули вигляду

$$I_n(f) = \sum_{i=1}^{n+1} A_i^{(n+1)} f(x_i).$$

Одним із способів побудови квадратурних формул є такий. За заданими значеннями функції $f(x)$ у вузлах x_i , $i = \overline{1, n+1}$ будують інтерполяційний многочлен степеня n , тобто $P_n(x) = \sum_{i=1}^{n+1} f(x_i) l_{n,i}(x)$, де $l_{n,i}(x)$ — фундаментальні інтерполяційні многочлени, і покладають

$$I = \int_a^b \rho(x) f(x) dx \approx \int_a^b \rho(x) P_n(x) dx = I_n; \quad (1)$$

$$I_n = \sum_{i=1}^{n+1} c_i f(x_i),$$

тобто $c_i = \int_a^b \rho(x) l_{n,i}(x) dx$. Такі квадратурні формули називаються *формулами інтерполяційного типу*.

Нехай відрізок $[a, b]$ розбито на n однакових підінтервалів точками $x_i = a + (i - 1) \frac{b-a}{n}$, $i = \overline{1, n+1}$ і $\rho(x) = 1$. Тоді формула

$$I \approx \int_a^b P_n(x) dx = \sum_{i=1}^{n+1} A_i^{(n+1)} f(x_i) \quad (2)$$

називається *формулою Ньютона — Котеса*. Її вагові коефіцієнти визначаються виразом

$$A_i^{(n+1)} = \int_a^b l_{n,i}(x) dx, \quad (3)$$

де $l_{n,i}(x) = \omega_{n+1}(x)/(x - x_i)\omega'_{n+1}(x)$, а

$$\omega_{n+1}(x) = \prod_{j=1}^{n+1} (x - x_j). \quad (4)$$

При $n = 0$ дістаємо формулу прямокутників

$$I \approx (b - a) f\left(\frac{a+b}{2}\right); \quad (5)$$

при $n = 1$ — формулу трапецій

$$I \approx \frac{b-a}{2} (f(a) + f(b)); \quad (6)$$

при $n = 2$ — формулу Сімпсона

$$I \approx \frac{b-a}{6} \left(f(a) + 4f\left(\frac{a+b}{2}\right) + f(b) \right). \quad (7)$$

Доведено [12, с. 87], що $\sum_{i=1}^{n+1} |A_i^{(n+1)}| \rightarrow \infty$ при $n \rightarrow \infty$. Якщо при обчисленні значень функції у вузлах сітки допускається похибка ε , то неусувна похибка обчислення квадратурної суми $\sum_{i=1}^{n+1} A_i^{(n+1)} f(x_i)$ може

досягати значення $\varepsilon \sum_{i=1}^{n+1} |A_i^{(n+1)}|$ і при $n \rightarrow \infty$ прямує до нескінченності. Число $\Lambda_n = \sum_{i=1}^{n+1} |A_i^{(n+1)}|$ природно назвати *числом обумовленості* задачі обчислення квадратурної суми.

Таким чином, при великих n обчислення квадратурної суми Ньютона — Котеса є погано обумовленою задачею, а число обумовленості — немовби коефіцієнтом підсилення похибки вхідних даних у остаточному результаті обчислень. Тому при великих n формула Ньютона — Котеса не застосовується.

Похибки R_j^* , $j = \overline{1, 3}$ наближених рівностей (5) — (7) визначаються відповідно виразами

$$R_1^* = \frac{1}{24} (b-a)^3 f''(\eta_1), \quad \eta_1 \in (a, b), \quad f(x) \in C^2[a, b]; \quad (5a)$$

$$R_2^* = -\frac{1}{12} (b-a)^3 f''(\eta_2), \quad \eta_2 \in (a, b), \quad f(x) \in C^2[a, b]; \quad (6a)$$

$$R_3^* = -\left(\frac{b-a}{2}\right)^5 \frac{f^{(IV)}(\eta_3)}{90}, \quad \eta_3 \in (a, b), \quad f(x) \in C^4[a, b]. \quad (7a)$$

Звідси випливає, що формули (5), (6) будуть точними для многочленів третього степеня.

З метою підвищення точності обчислення інтеграла використовують ускладнені квадратурні формули. Розіб'ємо відрізок інтегрування $[a, b]$ на n однакових частин точками $x_i = a + (i-1)h$, $i = \overline{1, n+1}$

і запишемо
$$\int_a^b f(x) dx = \sum_{i=1}^n \int_{x_i}^{x_{i+1}} f(x) dx.$$

Нехай $f(x) \in C^2[a, b]$. Тоді з (5) дістанемо

$$\int_{x_i}^{x_{i+1}} f(x) dx \approx hf\left(a + \left(i - \frac{1}{2}\right)h\right) = hf_{i-1/2}. \quad (8)$$

Згідно з (5a) маємо

$$\int_{x_i}^{x_{i+1}} f(x) dx = hf_{i-1/2} + \frac{h^3}{24} f''(\xi_i), \quad \xi_i \in (x_i, x_{i+1}). \quad (9)$$

Враховуючи (8), ускладнена формула прямокутників набуває вигляду

$$\int_a^b f(x) dx \approx h \sum_{i=1}^n f_{i-1/2}. \quad (10)$$

Підсумовуючи (9) і враховуючи [5, с. 109] рівність

$$\sum_{i=1}^n f''(\xi_i) = n \left(\frac{1}{n} \sum_{i=1}^n f''(\xi_i) \right) = n f''(\xi) = \frac{b-a}{h} f''(\xi),$$

де ξ — деяка точка з відрізка (a, b) , дістаємо ускладнену формулу прямокутників із залишковим членом

$$\int_a^b f(x) dx = h \sum_{i=1}^n f_{i-1/2} + h^2 \frac{b-a}{24} f''(\xi). \quad (11)$$

На підставі (6) запишемо ускладнену формулу трапецій

$$\int_a^b f(x) dx \approx \frac{h}{2} \sum_{i=1}^n (f_i + f_{i+1}) \quad (12)$$

і відповідну формулу з залишковим членом

$$\int_a^b f(x) dx = \frac{h}{2} \sum_{i=1}^n (f_i + f_{i+1}) - h^2 \frac{b-a}{12} f''(\xi), \quad (13)$$

де $f_i = f(a + (i-1)h)$; $\xi \in (a, b)$.

Нехай $f(x) \in C^4[a, b]$. Розіб'ємо відрізок $[a, b]$ на $2m$ однакових частин точками $x_i = a + (i-1)h$, $h = (b-a)/2m$, $i = \overline{1, 2m+1}$. Тоді, аналогічно до попереднього, з (7) дістанемо ускладнену формулу Сімпсона

$$I \approx \frac{h}{3} \sum_{j=1}^m (f(x_{2j-1}) + 4f(x_{2j}) + f(x_{2j+1})) \quad (14)$$

і відповідну формулу з залишковим членом

$$I = \frac{h}{3} \sum_{j=1}^m (f(x_{2j-1}) + 4f(x_{2j}) + f(x_{2j+1})) - \frac{h^4(b-a)}{180} f^{(IV)}(\xi), \quad (15)$$

де $\xi \in (a, b)$.

Введемо позначення

$$I_h^{np} = h \sum_{i=1}^n f_{i-1/2}; \quad I_h^{tp} = \frac{h}{2} \sum_{i=1}^n (f_i + f_{i+1}),$$

де $h = (b-a)/n$, $f_q = f(a + (q-1)h)$, а також позначення

$$I_h^c = \frac{h}{3} \sum_{i=1}^m (f(x_{2i-1}) + 4f(x_{2i}) + f(x_{2i+1})),$$

де $h = (b-a)/2m$.

Наближені рівності

$$\int_a^b f(x) dx \approx I_h^{np}; \quad \int_a^b f(x) dx \approx I_h^{tp}; \quad \int_a^b f(x) dx \approx I_h^c$$

називатимемо формулами прямокутників, трапецій, Сімпсона обчислення визначених інтегралів.

▼ У квадратурній формулі Гаусса [5, с. 113; 12, с. 89]

$$\int_a^b \rho(x) f(x) dx \approx \sum_{i=1}^n A_i^{(n)} f(x_i), \quad \rho(x) > 0 \quad (16)$$

вузли x_i та коефіцієнти $A_i^{(n)}$ вибираються, виходячи з того, що ця формула має бути точною для многочленів степеня $2n-1$.

Для їх визначення запишемо систему рівнянь

$$\int_a^b \rho(x) x^k dx = \sum_{i=1}^n A_i^{(n)} x_i^k, \quad k = \overline{0, 2n-1}. \quad (17)$$

При $a = -1$, $b = 1$, $\rho(x) = 1$ вузли квадратурної формули Гаусса є коренями многочлена Лежандра степеня n . Для найбільш поширених вагових коефіцієнтів $\rho(x)$ вузли x_i та коефіцієнти $A_i^{(n)}$ наведено в [12, с. 93] разом з відповідними ортогональними многочленами.

Для обчислення інтеграла $\int_a^b f(x) dx$ зробимо таку заміну змінної:

$$x = \frac{a+b}{2} + \frac{b-a}{2} t. \text{ Тоді}$$

$$\int_a^b f(x) dx = \frac{b-a}{2} \int_{-1}^1 f\left(\frac{a+b}{2} + \frac{b-a}{2} t\right) dt = \frac{b-a}{2} \int_{-1}^1 F(t) dt \approx \approx \frac{b-a}{2} \sum_{i=1}^n A_i^{(n)} F(t_i). \quad (18)$$

При $n = 8$ коефіцієнти $A_j^{(8)}$, $j = \overline{1, 8}$, які часто називають *коефіцієнтами Кристофеля*, і вузли t_j , $j = \overline{1, 8}$ мають такі значення: $A_1^{(8)} \approx 0,10122854$; $A_2^{(8)} \approx 0,22238104$; $A_3^{(8)} \approx 0,31370664$; $A_4^{(8)} \approx 0,36278378$; $A_{9-j} = A_j^{(8)}$, $j = \overline{1, 4}$; $t_5 \approx 0,18343464$; $t_6 \approx 0,52553242$; $t_7 \approx 0,79666648$; $t_8 \approx 0,96028986$; $t_{9-j} = -t_j$, $t = \overline{5, 8}$.

Залишковий член R_n квадратурної формули Гаусса [5, с. 117] визначається виразом

$$R_n = \frac{(b-a)^{2n+1} (n!)^4}{((2n!)^3 (2n+1)} f^{(2n)}(\xi), \quad (19)$$

якщо $f(x) \in C^{2n}[a, b]$.

Формули Гаусса можна застосовувати, коли функція $f(x)$ має похідні досить високого порядку, а значення їх є невеликими.

За допомогою квадратурної формули Гаусса, знайденої для відрізка $[-1, 1]$, можна побудувати ускладнену квадратурну формулу Гаусса для довільного відрізка $[a, b]$. Розіб'ємо останній на N однакових елементарних відрізків $[x_k^*, x_{k+1}^*]$, де $x_k^* = a + (k-1)(b-a)/N$, $k = \overline{1, N}$, $x_{N+1}^* = b$. Нехай на кожному з них задано n вузлів, тобто

$$x_{kj} = \frac{x_k^* + x_{k+1}^*}{2} + t_j \frac{b-a}{2N}, \quad j = \overline{1, n}, \quad (20)$$

де t_j — вузли формули Гаусса (18).

Розміщення вузлів x_{kj} , $j = \overline{1, n}$ на елементарному відрізку $[x_k^*, x_{k+1}^*]$ геометрично подібне до розміщення вузлів t_j у формулі Гаусса (18). При цьому остання набуває вигляду

$$\int_{x_k^*}^{x_{k+1}^*} f(x) dx = \frac{b-a}{2N} \sum_{i=1}^n A_i^{(n)} f(x_{ki}), \quad (21)$$

де $A_i^{(n)}$ — ті самі коефіцієнти, що й у формулі (18).

Підсумовуючи (21), дістаємо ускладнену квадратурну формулу Гаусса

$$\int_a^b f(x) dx \approx \frac{b-a}{2N} \sum_{i=1}^n A_i^{(n)} \sum_{k=1}^n f(k_i) = I_h^{\Gamma}, \quad (22)$$

де $h = (b-a)/N$.

У формулі (16) функція $\rho(x)$ може мати особливість. Тому ця формула застосовується для обчислення невластних інтегралів $\int_a^b \Phi(x) dx$, якщо попередньо записати функцію $\Phi(x)$ у вигляді $\Phi(x) = \rho(x) f(x)$, де $f(x)$ — досить гладка функція на відрізку $[a, b]$.

Коефіцієнти Кристофеля $A_i^{(n)} > 0$ [12, с. 91] задовольняють умову $\sum_{i=1}^n A_i^{(n)} = \int_a^b \rho(x) dx$.

Таким чином, обчислення квадратурної суми $\sum_{i=1}^n A_i^{(n)} f(x_i)$ є добре обумовленою задачею, оскільки число обумовленості $\Lambda_n = \sum_{i=1}^n A_i^{(n)} = \int_a^b \rho(x) dx$ — стала величина.

▼ Розглянемо апостеріорну оцінку похибки квадратурних формул [5, с. 118]. Нехай $f(x) \in C^4[a, b]$; тоді виконується рівність

$$\int_{x_i}^{x_{i+1}} f(x) dx = hf_{i-1/2} + \frac{h^3}{24} f''_{i-1/2} + \frac{h^5}{1920} f^{(IV)}(\xi_i). \quad (23)$$

Підсумовуючи (23) по i від 1 до n , дістаємо

$$I = I_h^{\text{np}} + \frac{h^2}{24} \left(h \sum_{i=1}^n f''_{i-1/2} \right) + h^4 \frac{b-a}{1920} f^{(IV)}(\xi). \quad (24)$$

Підставивши в (11) замість функції $f(x)$ її другу похідну, матимемо

$$\int_a^b f(x) dx = h \sum_{i=1}^n f''_{i-1/2} + h^2 \frac{b-a}{24} f^{(IV)}(\xi), \quad \xi \in [a, b]. \quad (25)$$

З (24) та (25) випливає, що

$$I = I_h^{\text{np}} + Ch^2 + O(h^4), \quad (26)$$

де $C = \frac{1}{24} \int_a^b f''(x) dx$.

Величина Ch^2 у (26) називається *головною частиною* похибки формули прямокутників.

Аналогічно можемо записати

$$I = I_h^{TP} + C_1 h^2 + O(h^4), \quad (27)$$

де $C_1 = -\frac{1}{12} \int_a^b f''(x) dx$.

Зазначимо, що коли $\int_a^b f''(x) dx \neq 0$ при $f(x) \in C^4[a, b]$, формули прямокутників і трапецій дають двостороннє наближення інтеграла $\int_a^b f(x) dx$. У припущенні $f(x) \in C^6[a, b]$ можна дістати співвідношення

$$I = I_h^C + C_2 h^4 + O(h^6), \quad (28)$$

де $C_2 = -\frac{1}{180} \int_a^b f^{(IV)}(x) dx$.

Якщо $f(x) \in C^{2n+2}[a, b]$, то виконується рівність

$$I = I_h^{\Gamma} + C_3 h^{2n} + O(h^{2n+2}), \quad (29)$$

де

$$C_3 = \frac{(n!)^4}{((2n)!)^3 (2n+1)} \int_a^b f^{(2n)}(x) dx, \quad h = (b-a)/N,$$

а N — кількість однакових елементарних відрізків, на які розбивається відрізок $[a, b]$.

Для апостеріорної оцінки похибки квадратурних формул скористаємось правилом Рунге [5, с. 120]. Нехай z — невідоме точне значення деякої величини; z_h — її відоме наближене значення, залежне від параметра h . Нехай виконується співвідношення

$$z = z_h + C h^k + O(h^{k+m}), \quad (30)$$

де C — невідома стала, що не залежить від h ; k, m — відомі числа. Тоді

$$z = z_{h/2} + C (h/2)^k + O(h^{k+m}). \quad (31)$$

На підставі (30), (31) дістаємо

$$z_{h/2} - z_h = C (h/2)^k (2^k - 1) + O(h^{k+m}),$$

звідки

$$C (h/2)^k = \frac{z_{h/2} - z_h}{2^{k-1}} + O(h^{k+m}). \quad (32)$$

Отже, з точністю до $0 (h^{k+m})$

$$z - z_{h/2} \approx \frac{z_{h/2} - z_h}{2^k - 1}. \quad (33)$$

Обчислення наближеної оцінки похибки за формулою (33) називається *правилом Рунге*. З (32) випливає, що коли $C \neq 0$, права частина (33) має k -й порядок відносно параметра h і відрізняється від $C (h/2)^k$ на величину більш високого порядку по h .

Правило Рунге застосовується, якщо $C \neq 0$. На практиці це означає виконання умови

$$\left| 2^k \frac{z_h - z_{h/2}}{z_{2h} - z_h} - 1 \right| < 0,1. \quad (34)$$

Нерівність (34) може не виконуватись з таких причин: а) параметр h має велике значення і при цьому не можна не враховувати член $0 (h^{k+m})$; б) h — дуже малий параметр, тоді впливає похибка заокруглення; в) якщо стала $C = 0$ або близька до нуля.

Позначимо через I_h наближене значення інтеграла $I = \int_a^b f(x) dx$, знайдене за однією з квадратурних формул

$$I = I_h + Ch^k + 0(h^{k+2}), \quad (35)$$

де C — стала, що не залежить від h ; $k = 2$ для формул прямокутників і трапецій; $k = 4$ для формули Сімпсона; $k = n$ для формули Гаусса (в розглядуваному випадку $n = 8$).

Очевидно, для застосування правила Рунге необхідно, щоб $f(x) \in C^{k+2} [a, b]$, причому похідні від функції $f(x)$ не повинні сильно змінюватися.

Покладаючи $z_{rh} = I_{rh}$ (де $r = 1/2; 1; 2$) і переконуючись, що виконується нерівність (34), можна наближено оцінити похибку квадратурних формул за правилом Рунге:

$$I - I_{h/2} \approx \frac{I_{h/2} - I_h}{2^k - 1}. \quad (36)$$

Таким чином, для обчислення інтеграла $I = \int_a^b f(x) dx$ можна скористатись наближеною рівністю

$$I \approx I_{h/2} + \frac{I_{h/2} - I_h}{2^k - 1}, \quad (37)$$

похибка якої для формул прямокутників і трапецій є величиною порядку $0 (h^3)$, для формули Сімпсона — $0 (h^4)$, для формули Гаусса — $0 (h^{2n})$.

Приклади використання підпрограм. Приклад 1. Чисельну реалізацію на ЕОМ формули (14) розглянемо на прикладі такої задачі: потрібно обчислити інтеграл

$\int_a^b f(x) dx$, де функція $f(x)$ визначається виразом

$$f(x) = 100 \sin x. \quad (38)$$

Головна програма, яка містить звернення до підпрограми SIMP і підпрограми-функції обчислення значення функції $f(x)$, має вигляд:

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛА C
C ФУНКЦІЇ F(X) НА ВІДРІЗКУ [A,B] ЗА ФОРМУЛОЮ C
C СІМПСОНА C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
EXTERNAL F
M=3
TYPE 201
ACCEPT *,A
TYPE 202
ACCEPT *,B
TYPE 203
ACCEPT 204,NW
TYPE 205
ACCEPT *,EPS
HMIN=1.E-6
CALL SIMP(A,B,M,EPS,HMIN,RHMIN,IERR,F,S)
WRITE(NW,206) EPS
WRITE(NW,207) A,B
WRITE(NW,208) S,RHMIN,IERR
STOP
201 FORMAT(1H$, 'A=')
202 FORMAT(1H$, 'B=')
203 FORMAT(1H$, 'ЗАДАТИ NW — КАНАЛ ВИВЕДЕННЯ',
* 'РЕЗУЛЬТАТИВ')
204 FORMAT(I)
205 FORMAT(1H$, 'ВВЕСТИ ТОЧН. ОБЧ. ІНТЕГР. EPS=')
206 FORMAT(/5X, 'S—ЗНАЧ. ІНТЕГР. ТА ФОРМ. СІМПСОНА'
* '/5X,' 3 ТОЧНІСТЮ EPS=',E13.5)
207 FORMAT(5X, ' A=',E13.5,2X,' B=',E13.5)
208 FORMAT(5X, ' S=',E13.5,2X,' RHMIN=',E13.5,2X,
* ' IERR=',I2)
END
FUNCTION F(X)
F=100.*SIN(X)
RETURN
END
```

Обчислення за цією програмою на комп'ютері ДВК-2М дають такий результат:
S — ЗНАЧ. ІНТЕГР. ЗА ФОРМ. СІМПСОНА
3 ТОЧНІСТЮ EPS=0.10000E-04

A= 0.00000E+00 B= 0.31416E+01

S= 0.20000E+03 RHMIN= 0.24544E-01 IERR= 0

Точне значення інтеграла дорівнює 200.

Приклад 2. Чисельну реалізацію на ЕОМ формули (22) розглянемо на прикладі обчислення інтеграла з попереднього прикладу.

Головна програма, яка містить звернення до підпрограми-функції обчислення значення функції $f(x)$, має вигляд:

```
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛА C
C ФУНКЦІЇ F(X) ЗА ФОРМУЛОЮ ГАУССА C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCG

EXTERNAL F
TYPE 201
ACCEPT *,A
TYPE 202
ACCEPT *,B
TYPE 203
ACCEPT *,EPS
TYPE 204
ACCEPT 205,NW
CALL GAUSS(A,B,F,G)
A1=A
B1=B
G2=0.
K=2
U=65535.
10 DO 20 I=1,K
    B=A1+I*(B1-A1)/K
    CALL GAUSS(A,B,F,G1)
    A=B
    G2=G2+G1
20 CONTINUE
V=(G2-G)/U
G2=G2+V
IF(ABS(V).LT.EPS) GO TO 30
K=K*2
G=G2
G2=0.
A=A1
GO TO 10
30 WRITE(NW,206)A1,B1,K,EPS
WRITE(NW,207) G
STOP
201 FORMAT(1H$, 'A=')
202 FORMAT(1H$, 'B=')
203 FORMAT(1H$, 'ВВЕСТИ EPS—ТОЧН. ОБЧИСЛ. ІНТЕГР.',
* 'EPS=')
204 FORMAT(1H$, 'ЗАДАТИ NW—КАНАЛ ВИВЕДЕННЯ',
* 'РЕЗУЛЬТАТИВ')
205 FORMAT(1)
206 FORMAT(5X, 'МЕЖІ ІНТЕГРУВАННЯ:/'
* 5X, 'A=', E13.5, 'B=', E13.5, /
* 5X, 'КІЛЬК. ПОДРІБНЕНЬ ВІДРІЗКА K=', 15/
* 5X, 'ТОЧН. ОБЧИСЛ. ІНТЕГР. EPS=', E13.5)
207 FORMAT(5X, 'ЗНАЧ. ІНТЕГР. ЗА ФОРМ. ГАУССА',
* 'G=', E13.5)

END
```

```

FUNCTION F(X)
F=100.*SIN(X)
RETURN
END

```

Обчислення за цією програмою на комп'ютері ДВК-2М дають такий результат:
МЕЖІ ІНТЕГРУВАННЯ:

```

A=0.00000E+00  B=0.31416E+01
КІЛЬК. ПОДРІБНЕНЬ ВІДРІЗКА  K=4
ТОЧН. ОБЧИСЛ. ІНТЕГР. EPS=0.10000E-06
ЗНАЧ. ІНТЕГР. ЗА ФОРМ. ГАУССА  G=0.20002E+03

```

Приклад 3. Досить ефективною підпрограмою обчислення визначених інтегралів є підпрограма QUAC8 [26, с. 108—122]. Її назва походить від слів Quadrature — Adaptive — Newton — Cotes'8-panel, тобто квадратура адаптивна, Ньютона — Котеса восьмого порядку (мається на увазі степінь інтерполяційного многочлена, за яким вона побудована).

Адаптивна підпрограма є реалізацією такого алгоритму чисельного інтегрування, в якому застосовуються одна чи дві квадратурні формули. За допомогою цієї підпрограми автоматично визначається довжина підінтервалів так, щоб остаточний результат обчислення мав задану точність. У різних частинах інтервалу можуть використовуватись сітки з рівними кроками, більшими там, де підінтегральна функція $f(x)$ змінюється повільно, й меншими там, де вона змінюється сильно.

Нехай $\int_{x_i}^{x_{i+1}} f(x) dx = I^{(i)}$, $x_{i+1} - x_i = h_i$; $I_h^{(i)}$, $I_{h/2}^{(i)}$ — наближені значення $I^{(i)}$,

знайдені за якою-небудь квадратурною формулою k -го порядку точності з кроками h і $h/2$. Тоді підінтервал $[x_i, x_{i+1}]$ ділиться навпіл стільки разів, поки не буде виконуватись нерівність

$$\frac{1}{2^k - 1} |I_h^{(i)} - I_{h/2}^{(i)}| \leq \frac{h_i}{b-a} \varepsilon, \quad (39)$$

де ε — потрібна точність обчислення інтеграла $\int_a^b f(x) dx$.

Головна програма, яка містить звернення до підпрограми QUAC8, підпрограми-функції обчислення значення функції $f(x)$ за формулою

$$f(x) = (2x - a - b) \sin(x - a)(x - b) + (x - a)(x - b) \cos(x - a)(x - b),$$

має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛА C
C З ВИКОРИСТАННЯМ АДАПТИВНОЇ ПІДПРОГРАМИ QUAC8, C
C ЗАПОЗИЧЕНОЇ З [26, с. 113—122] C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
COMMON /C1/A,B
C F—ІМ'Я ПІДПРОГРАМИ ФУНКЦІЇ, ВІД ЯКОЇ ОБЧИСЛЮЄТЬСЯ
C ІНТЕГРАЛ. А ТА В — МЕЖІ ІНТЕГРУВАННЯ
EXTERNAL F
TYPE 201
ACCEPT *,A
TYPE 202
ACCEPT *,B
TYPE 203
ACCEPT 204,NW

```

```

TYPE 205
ACCEPT 206,RELERR
TYPE 207
ACCEPT 206,ABSERR
C ВВЕДЕННЯ ВХІДНОЇ ІНФОРМАЦІЇ ЗАВЕРШЕНО.
C ДАЛІ ЙДЕ ЗВЕРНЕННЯ ДО АДАПТИВНОЇ ПІДПРОГРАМИ QUANC8
CALL QUANC8(F,A,B,ABSERR,RELERR,RESULT,ERREST,
* NOFUN,FLAG)
WRITE(NW,208)
WRITE(NW,209)A,B
WRITE(NW,210) RESULT, ERREST
IF(FLAG.NE.0.0) WRITE(NW,211) FLAG
STOP
201 FORMAT(1H$, 'НИЖНЯ МЕЖА ІНТЕГРУВАННЯ A=')
202 FORMAT(1H$, 'ВЕРХНЯ МЕЖА ІНТЕГРУВАННЯ B=')
203 FORMAT(1H$, 'ЗАДАТИ N—WRITE=')
204 FORMAT(I)
205 FORMAT(1H$, 'ЗАДАТИ ГРАНИЦЮ ВІДН. ПОХИБКИ',
* 'RELERR=')
206 FORMAT(E16.9)
207 FORMAT(1H$, 'ЗАДАТИ ГРАНИЦЮ АБСОЛ. ПОХИБКИ',
* 'ABSERR=')
208 FORMAT(4X, 'ЗНАЧЕННЯ ІНТЕГРАЛА, ЗНАЙДЕНЕ ЗА',
* 'ДОПОМ. QUANC8')
209 FORMAT(4X, 'A=', E12.5, 'B=', E12.5)
210 FORMAT(4X, 'RESULT=', E12.5, 'ERREST=', E12.5)
211 FORMAT(4X, 'ПОПЕРЕДЖЕННЯ: РЕЗУЛЬТАТ НЕНАДІЙНИЙ'
* ',/20X, 'FLAG=', E16.6)
END

FUNCTION F(X)
COMMON /C1/A,B
F=(2.*X-A-B)*SIN((X-B)*(X-A))+((X-A)*(X-B))*
* COS((X-A)*(X-B))*(2.*X-A-B)
RETURN
END

```

Обчислення за цією програмою на комп'ютері ДВК-2М дають такий результат
 ЗНАЧЕННЯ ІНТЕГРАЛА, ЗНАЙДЕНЕ ЗА ДОПОМ. QUANC8

A=0.10000E+01 B= 0.50000E+01

RESULT=0.00000E+00 ERREST=0.10255E-07

Точне значення інтеграла дорівнює нулю.

Запитання для самоперевірки

1. Які ви знаєте способи побудови квадратурних формул? 2. Які квадратурні формули називаються формулами відкритого та замкненого типів? 3. Які квадратурні формули називаються формулами найвищого алгебраїчного ступеня точності? 4. Які ортогональні многочлени застосовуються при побудові квадратурних формул

Гаусса для інтегралів $\int_{-1}^{+1} f(x) dx$ і $\int_{-1}^{+1} f(x) (1-x^2)^{-1/2} dx$? Який вигляд мають ці формули для трьох вузлів? 5. Чи завжди ускладнені формули чисельного інтегру-

вання мають кращий порядок збіжності порівняно з формулами Гауса? * 6. Що таке головний член похибки квадратурної формули? 7. Який вигляд мають головні члени похибок формул прямокутників, трапецій, Сімпсона? 8. В чому полягає правило Рунге апостеріорної оцінки похибки квадратурних формул? 9. Які умови дозволяють застосовувати правило Рунге?

Контрольні завдання

1. Обчислити інтеграл $\int_a^b f(x) dx$, використовуючи підпрограми SIMP, GAUSS і QUANC8, змінюючи межі інтегрування, якщо

$$f(x) = (2x - a - b) \varphi(x) + (x^2 - (a + b)x + ab) \varphi'(x),$$

де $\varphi(x)$ — довільна достатнє число разів диференційовна функція на відрізку $[a, b]$. Порівняти знайдені результати між собою.

Відповідь: $\int_a^b f(x) dx = 0$.

2. Визначити силу тиску на вертикальну стінку, що має форму півастроїди $x^{2/3} + y^{2/3} = a^{2/3}$, коли рівень води задається умовою $x = 0$.

Вказівка. Якщо довжину відрізка горизонтальної прямої, проведеної на стінці на відстані x від рівня води AB (рис. 6), позначити через $S(x)$, то можна вважати, що заштрихована область між лініями x і $x + dx$ є прямокутником з основою $S(x)$ і висотою dx .

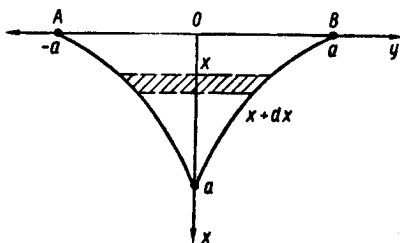


Рис. 6

Відповідно до законів гідростатики сила тиску води на прямокутну смугу завдовжки $S(x)$ висотою dx , опущену на глибину x з точністю до dx , визначається ви-

разом $\Delta p = xS(x) dx$. Оскільки $S(x) = 2\sqrt{(a^{2/3} - x^{2/3})^3}$, маємо $\Delta p = 2x\sqrt{(a^{2/3} - x^{2/3})^3} dx$. Тоді шуканий тиск води можна визначити за формулою

$$P = 2 \int_a^b x \sqrt{(a^{2/3} - x^{2/3})^3} dx.$$

3. Скласти програму розв'язування рівнянь:

а) $\int_0^x \frac{\sin^2 t}{t} dt = a$, де $a \in [0, \infty[$;

б) $\int_0^x e^{-t^2} dt = a$, де $a \in [0, \sqrt{\pi}/10]$.

* Див. [5, с. 131].

4. Скласти програму побудови графіка функції

$$F(x) = \int_a^x f(t) dt, \quad x \in [a, b],$$

де $f(t)$ — задана функція.

5. Скласти програму обчислення довжини дуги кривої, якщо її рівняння записано в параметричній формі:

$$\begin{cases} x = \varphi(t); \\ y = \psi(t), \quad t_1 \leq t \leq t_2, \end{cases}$$

де t_1, t_2 — задані числа; $\varphi(t), \psi(t)$ — задані функції.

§ 9. Найкраще середньоквадратичне наближення

Постановка задачі. Побудувати многочлен найкращого середньоквадратичного наближення (МНК) [12, с. 53] функції $f(x)$, що належить гільбертовому простору $H = L_{2,\alpha} [a, b]$, в якому скалярний добуток визначається виразом

$$(f(x), g(x)) = \int_a^b f(x) g(x) d\alpha(x). \quad (1)$$

Тут $\alpha(x)$ — неспадна функція; інтеграл береться в розумінні Лебега — Стільтьєса, а функції $\alpha(x), f(x)$ задаються викладачем.

Методичні вказівки. Відстань між функціями в термінах скалярного добутку (1) називається *середньоквадратичним відхиленням* функції $g(x)$ від функції $f(x)$:

$$\Delta(f, g) = \left(\int_a^b (f(x) - g(x))^2 d\alpha(x) \right)^{1/2}. \quad (2)$$

Якщо $\alpha(x)$ — функція стрибків, тобто є кусково-сталою і має стрибки величин ρ_i в точках $x_i \in [a, b]$, то скалярний добуток (1) набуває вигляду

$$(f, g) = \sum_j \rho_j f(x_j) g(x_j). \quad (1a)$$

Відстань між функціями дискретного аргументу в цьому разі визначається виразом

$$\Delta(f, g) = \left(\sum_j \rho_j (f(x_j) - g(x_j))^2 \right)^{1/2}. \quad (2a)$$

▼ Нехай $\{\varphi_k(x)\}, k = \overline{1, n}, x \in [a, b]$ — система лінійно незалежних функцій. Функцію $Q_{n-1}(x) = \sum_{k=1}^n c_k \varphi_k(x)$ називають *узагальненим многочленом* степеня $n-1$, побудованим за системою

$\{\varphi_k\}$. Многочлен $Q_{n-1}(x)$, який мінімізує квадрат норми (2) або (2а), називається *многочленом найкращого середньоквадратичного наближення* функції $f(x)$, заданої на відрізку $[a, b]$ чи в дискретній системі точок $x_i, i = \overline{1, m}$, причому $m \geq n$. Величина $\|f(x) - Q_{n-1}(x)\|$ називається *середньоквадратичним відхиленням* многочлена $Q_{n-1}(x)$ від функції $f(x)$.

Для визначення коефіцієнтів $c_k, k = \overline{1, n}$ МНСН $Q_{n-1}(x)$ необхідно розв'язати СЛАР

$$\sum_{k=1}^n (\varphi_k, \varphi_j) c_k = (f, \varphi_j), \quad j = \overline{1, n}, \quad (3)$$

де (\cdot) — скалярний добуток.

Систему лінійних алгебраїчних рівнянь (3) називають *нормальною*, її визначником є відмінний від нуля визначник Грама для системи лінійно незалежних функцій $\{\varphi_k(x)\}, k = \overline{1, n}$. Такими системами функцій можуть бути, наприклад, системи:

а) $\{\varphi_k(x)\} = \{(x - x_0)^{k-1}\}, k = \overline{1, n}; x \in [a, b]$, де a, b — довільні дійсні числа;

б) $\{\varphi_k(x)\} = \{e^{(k-1)x}\}, k = \overline{1, n}; x \in [a, b]$;

в) $\{\varphi_k(x)\} = \{P_{k-1}(x)\}, x \in [-1, 1], k = \overline{1, n}$,

де $P_{k-1}(x)$ — система многочленів Лежандра, ортогональних у розумінні (1) при $\alpha(x) = x$;

г) $\{\varphi_k(x)\} = \{T_{k-1}(x)\}, x \in [-1, 1], k = \overline{1, n}; T_{k-1}(x)$ — система многочленів Чебишова першого роду.

▼ Якщо система функцій $\{\varphi_k(x)\}, k = \overline{1, n}$ є ортогональною, то в цьому разі матриця СЛАР (3) діагональна, а коефіцієнти многочлена $Q_{n-1}(x)$ визначаються так:

$$c_k = (f, \varphi_k) / (\varphi_k, \varphi_k), \quad k = \overline{1, n}. \quad (4)$$

Наведемо рекурентні співвідношення для визначення многочленів Лежандра та Чебишова. Многочлени Лежандра обчислюються за рекурентними співвідношеннями

$$P_{n+1}(x) = ((2n+1)xP_n(x) - nP_{n-1}(x))(n+1),$$

де n — степінь многочлена; $P_0(x) = 1; P_1(x) = x$.

Подальші обчислення дають $P_2(x) = (3x^2 - 1)/2, P_3(x) = (5x^3 - 3x)/2, P_4(x) = (35x^4 - 70x^3 + 15x)/8$ і т. д.

Многочлени Лежандра утворюють ортогональну систему на відрізку $[-1, 1]$ з ваговою функцією $\rho(x) = 1$, причому

$$\int_{-1}^1 P_m(x) P_n(x) dx = \begin{cases} 0, & m \neq n; \\ 2/(2n+1), & m = n, n = 0, 1, \dots \end{cases}$$

Многочлени Чебишова першого роду обчислюються за рекурентними співвідношеннями

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x),$$

де n — степінь многочлена; $T_0(x) = 1$; $T_1(x) = x$.

Подальші обчислення дають $T_2(x) = 2x^2 - 1$, $T_3(x) = 4x^3 - 3x$, $T_4(x) = 8x^4 - 8x^2 + 1$ і т. д. При $|x| \leq 1$ многочлени Чебишова $T_n(x) = \cos((n-1)\arccos x)$ утворюють ортогональну систему на відрізьку $[-1, 1]$ з ваговою функцією $\rho(x) = 1/\sqrt{1-x^2}$, причому

$$\int_{-1}^1 \rho(x) T_{k-1}(x) T_{j-1}(x) dx = \begin{cases} \pi & \text{при } k = j = 1; \\ \pi/2 & \text{при } k = j \neq 1; \\ 0 & \text{при } k \neq j. \end{cases}$$

▼ Якщо система лінійно незалежних функцій $\{\varphi_k(x)\}$, $k = \overline{1, n}$ не є ортогональною, то матриця СЛАР (3) повністю заповнена і, як правило, є погано обумовленою, причому її число обумовленості зростає зі збільшенням степеня n . Тому рекомендується будувати многочлен $Q_{n-1}(x)$ невисокого степеня ($n = 3 \div 6$).

▼ При побудові МНСН $Q_{n-1}(x)$ для функції $f(x)$ у випадку скалярного добутку (1) графік $Q_{n-1}(x)$ проходить ближче до тих точок графіка функції $f(x)$, де значення вагової функції $\rho(x)$ більше. Тому останню і вибирають великою для тих значень x , при яких необхідно мати більш високу точність апроксимації функції $f(x)$.

При побудові МНСН $Q_{n-1}(x)$ для функції $f(x)$ як функції дискретного аргументу її значення $f(x_i)$, $i = \overline{1, m}$ відомо з деякою похибкою ε_i . При цьому ваговим функціям ρ_i надають змісту точності вимірювання функції $f(x_i)$, тобто чим остання вища, тим більше значення вагових функцій надають у цій точці. Часто покладають, наприклад, $\rho_i = 1/\varepsilon_i^2$.

▼ Для розв'язання СЛАР (3) можна застосувати метод Гаусса або квадратного кореня, оскільки її матриця симетрична. Але ці методи досить часто не дають задовільних результатів, бо ця матриця є погано обумовленою і розв'язок виявляється дуже чутливим до похибок вхідних даних і похибок заокруглення, які виникають у процесі розв'язування СЛАР.

Одним з найбільш надійних методів формування МНСН в розумінні скалярного добутку (1а), побудованого для функції $f(x)$, значення якої задано в дискретній системі точок, є метод, який ґрунтується на матричній факторизації і називається *сингулярним розкладом* або методом SVD [26, с. 239—256].

Нехай \vec{f} є m -вимірним вектором значень функції $f(x)$ у точках x_i , $i = \overline{1, m}$, а \vec{c} — n -вимірним вектором з компонентами c_j , $j = \overline{1, n}$

(коефіцієнти МНСН). Тоді наближені рівності

$$Q_{n-1}(x_i) = \sum_{j=1}^n c_j \varphi_j(x_i) \approx f(x_i), \quad i = \overline{1, m},$$

які називаються умовними рівняннями, можна записати у вигляді

$$A\vec{c} = \vec{f}. \quad (5)$$

Метод SVD полягає ось у чому. Сингулярним розкладом матриці A , що має розмірність $m \times n$, називається будь-яка факторизація вигляду

$$A = U\Sigma V^T,$$

де U, V — ортогональні матриці розмірністю $m \times n$; Σ — діагональна матриця розмірністю $m \times n$, в якій $\sigma_{ij} = 0$ при $i \neq j$ і $\sigma_{ii} = \sigma_i \geq 0$.

Величини σ_i називаються сингулярними числами матриці A , стовпці матриць U, V — лівими та правими сингулярними векторами.

При цьому рівняння (5) набувають вигляду

$$U\Sigma V^T \vec{c} = \vec{f}, \quad (5a)$$

звідки $\Sigma \vec{z} = \vec{d}$, де $\vec{z} = V^T \vec{c}$; $\vec{d} = U^T \vec{f}$.

Значимо, що в матриці Σ може бути не більш як n перших ненульових стовпців; тому тільки перші n стовпців матриці U вносять вклад у добуток $U\Sigma V^T$.

Приклади використання підпрограм. **Приклад 1.** Побудуємо МНСН четвертого степеня $Q_4(x)$ для функції $f(x) = |x|$, $x \in [-1, 1]$, взявши многочлени Чебишова першого роду і вважаючи їх системою лінійно незалежних функцій, тобто

$$\varphi_k(x) = T_{k-1}(x) = \cos((k-1) \arccos x), \quad k = \overline{1, 5}.$$

Отже,

$$Q_4(x) = \sum_{k=1}^5 c_k T_{k-1}(x).$$

Коефіцієнти c_k , $k = \overline{1, 5}$ визначаємо за формулою (4):

$$c_1 = \frac{1}{\pi} \int_{-1}^1 \frac{|x|}{\sqrt{1-x^2}} dx = \frac{1}{2\pi} \int_{-\pi}^{\pi} |\cos \theta| d\theta;$$

$$c_k = \frac{2}{\pi} \int_{-1}^1 \frac{|x|}{\sqrt{1-x^2}} \cos((k-1) \arccos x) dx = \frac{1}{\pi} \int_{-\pi}^{\pi} |\cos \theta| \cos(k-1) \theta d\theta.$$

Головна програма розв'язування розглядуваної задачі за допомогою підпрограми SIMP має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА ПОБУДОВИ МНСН ФУНКЦІЇ, КОЛИ ЯК
С СИСТЕМУ ЛІНІЙНО НЕЗАЛЕЖНИХ ФУНКЦІЙ ВИБРАНО
С МНОГОЧЛЕНИ ЧЕБИШОВА ПЕРШОГО РОДУ
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL TK, TF
COMMON/CQ/I
COMMON/CP/C(10),K
A=-4.*ATAN(1.)
B=-A
TYPE 101
ACCEPT 102,K
TYPE 103
ACCEPT *,EPS
TYPE 104
ACCEPT 102,NW
M=3
HMIN=1.E-4
I=1
CALL SIMP(A,B,M,EPS,HMIN,RHMIN,IERR,TK,Y0)
C(I)=Y0/(2.*B)
DO 10 I=2,K
    CALL SIMP(A,B,M,EPS,HMIN,RHMIN,IERR,TK,Y0)
    C(I)=Y0/B
10 CONTINUE
CALL SIMP(A,B,M,EPS,HMIN,RHMIN,IERR,TF,Y0)
Y0=ABS(Y0)
S=SQRT(Y0)
WRITE(NW,105)(I,C(I),I=1,K)
WRITE(NW,106)S
STOP
101 FORMAT('$ ЗАДАТИ ЧИСЛО КООРД. ФУНКЦІЙ K=')
102 FORMAT(I1)
103 FORMAT('$ ВВЕСТИ EPS—ТОЧН. ОБЧИСЛ. ІНТЕГРА',
*          ,ЛІВ EPS=')
104 FORMAT('$ ЗАДАТИ КАНАЛ ВИВЕДЕННЯ РЕЗУЛЬТ. NW=')
105 FORMAT(4X,'C(',I1,')=',E12.5)
106 FORMAT('/ЗНАЧ. СЕРЕДНЬОКВ. ВІДХ. S=',E12.5)
END

FUNCTION TK(X)
COMMON/CQ/I
TK=ABS(COS(X))*COS((I-1)*X)
RETURN
END

FUNCTION TF(X)
COMMON/CP/C(10),K
TF=0.
DO 70 J=1,K
    TF=TF-C(J)*COS((J-1)*X)
70 CONTINUE

```

```

TF=(TF+ABS(COS(X)))*2
RETURN
END

```

Обчислення за цією програмою на комп'ютері ДВК-2М дають такий результат:

```

C(1)= 0.63662E+00
C(2)= 0.84292E-06
C(3)= 0.42441E+00
C(4)= 0.90307E-06
C(5)=-0.84880E-01

```

ЗНАЧ. СЕРЕДНЬОКВ. ВІДХ. S=0.17724E-02

Як бачимо, коефіцієнти c_2 та c_4 близькі до нуля. Це свідчить про те, що наближення знайденим многочленом функції $y = |x|$, $x \in [-1, 1]$ є досить добрим, бо зазначені коефіцієнти є коефіцієнтами відповідно при x і x^3 , а функція, яку наближуємо, — парна.

Приклад 2. Нехай відомо висоту польоту H (у кілометрах) літака, що летить над землею, в момент часу t , заданого в секундах:

H		1,5	1,87	2,14	3,2	3,75	4,2	5,1	5,7
t		10	20	30	40	50	60	70	80

На підставі цих даних МНСН має вигляд

$$Q_2(t) = c_1 + c_2 \left(\frac{t-45}{10} \right) + c_3 \left(\frac{t-45}{10} \right)^2,$$

користуючись яким можна обчислити висоту польоту літака в будь-який момент часу.

Значимо, що коли наближення проводиться на відріжку $[a, b]$, то при виборі як базисних функцій $\varphi_k(x) = \left(\frac{x - (a+b)/2}{(a+b)/2} \right)^{k-1}$, $k = \overline{1, n}$, число обумовленості

матриці СЛАР (3) буде меншим, ніж при виборі функцій $\varphi_k(x) = x^{k-1}$.

Головна програма розв'язування розглядуваної задачі, тобто системи умовних рівнянь (5), за допомогою підпрограми SVD має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА ПОБУДОВИ МНСН В РАЗІ C
C ТАБЛИЧНОГО ЗАДАВАННЯ ФУНКЦІЇ C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DIMENSION A(8,3),U(8,3),SIGMA(8),V(8,3),B(8),
* T(8),C(3),WORK(30),D(30),Z(30),R1(30)
LOGICAL L1,L2
DATA B/1.500,1.870,2.140,3.200,3.750,4.200,
* 5.100,5.700/
L1=.TRUE.
L2=.TRUE.
TYPE 201
ACCEPT*,NW
NM=8
N=3
M=8
C ФОРМУВАННЯ МАТРИЦІ УМОВНИХ РІВНЯНЬ
DO 10 I=1,M
T(I)=10.+(I-1)
A(I,1)=1,

```

```

      A(I,2)=(T(I)-45.)/10.
      A(I,3)=((T(I)-45.)/10.)**2
10    CONTINUE
      CALL SVD(NM,M,N,A,SIGMA,L1,U,L2,V,IERR,WORK)
      IF(IERR.NE.0) WRITE(NW,202)IERR
C     ВИЗНАЧЕННЯ ПЕРШИХ N КОМПОНЕНТІВ ВЕКТОРА ПРАВИХ ЧАСТИН
C     СЛАР SZ=D
      DO 20 J=1,N
        D(J)=0.
        DO 20 I=1,M
          D(J)=D(J)+U(I,J)*B(I)
20    CONTINUE
C     ВИЗНАЧЕННЯ НАЙБІЛЬШОГО СИНГУЛЯРНОГО ЧИСЛА МАТРИЦІ A
      SIGMA1=0.
      DO 30 J=1,N
        IF(SIGMA(J).GT.SIGMA1)SIGMA1=SIGMA(J)
30    CONTINUE
C     ПОКЛАСТИ ТАУ = ВІДНОСНІЙ ПОХИБЦІ ВХІДНИХ ДАНИХ, ЯКЩО
C     ІНФОРМАЦІЮ ТОЧНО НЕ ЗАДАНО, ЯКЩО Ж ВХІДНУ ІНФОРМАЦІЮ
C     ЗАДАНО ТОЧНО, ТО ПОКЛАСТИ ТАУ=MAX(SIGMA(J))*DELTA; ТУТ
C     ДЕЛЬТА = ПОХИБЦІ ЗАОКРУГЛЕННЯ — ВЕЛИЧИНА, БЛИЗЬКА ДО
C     МАШИННОГО НУЛЯ.
      TAU=1.E-04
C     РОЗВ'ЯЗУВАННЯ СЛАР SZ=D
      DO 40 J=1,N
        IF(SIGMA(J).GT.TAU) GO TO 50
        Z(J)=0.
        GO TO 40
50    Z(J)=D(J)/SIGMA(J)
40    CONTINUE
C     ЗНАХОДЖЕННЯ РОЗВ'ЯЗКУ СИСТЕМИ УМОВНИХ РІВНЯНЬ AC=B
      DO 60 I=1,N
        C(I)=0.
        DO 70 J=1,N
          C(I)=C(I)+V(I,J)*Z(J)
70    CONTINUE
60    CONTINUE
C     ЗНАХОДЖЕННЯ НЕВ'ЯЗКИ РОЗВ'ЯЗКУ СИСТЕМИ УМОВНИХ
C     РІВНЯНЬ AC=E, ||R|| =SQRT(AC-B,AC-B)
      DO 80 I=1,M
        R1(I)=0.
        DO 90 J=1,N
          R1(I)=R1(I)+A(I,J)*C(J)
90    CONTINUE
      R1(I)=R1(I)-B(I)
80    CONTINUE
      R=0.
      DO 100 I=1,M
        R=R+R1(I)**2
100   CONTINUE
      R=SQRT(R)
      WRITE(NW,203)
      DO 110 I=1,M
        WRITE(NW,204)(A(I,J),J=1,N),B(I)
110   CONTINUE
      WRITE(NW,205)

```

```

WRITE(NW,206)(J,SIGMA(J),J=1,N)
WRITE(NW,207)
DO 120 I=1,M
120   WRITE(NW,208)(U(I,J),J=1,N)
WRITE(NW,209)
DO 130 I=1,N
130   WRITE(NW,208)(V(I,J),J=1,N)
WRITE(NW,210)
WRITE(NW,211)(J,C(J),J=1,N)
WRITE(NW,212)R
STOP

201  FORMAT('$ ВВЕСТИ КАНАЛ ВИВЕД. РЕЗУЛЬТ. NW=')
202  FORMAT('K—ТЕ СИНГ. ЧИСЛО НЕ ОБЧИСЛЕНЕ ЗА 30',
*      ' IТЕР.; IERR=',I2)
203  FORMAT(/4X,'МАТР. А І ВЕКТ. ПРАВИХ ЧАСТ. В')
204  FORMAT(' ',4E12.5)
205  FORMAT(/4X,'SIGMA')
206  FORMAT(6X,'SIGMA'(I1,')=',E12.5)
207  FORMAT(/4X,'МАТР. U ІЗ СИНГУЛ. ЗАПИС.')
208  FORMAT(' ',3E12.5)
209  FORMAT(/4X,'МАТР. V ІЗ СИНГУЛ. ЗАПИС.')
210  FORMAT(/4X,'КОЕФІЦІЄНТИ C')
211  FORMAT(6X,'C'(I1,')=',E12.5)
212  FORMAT(/4X,' ЗНАЧ. СЕРЕДНЬОКВ. ВІДХ. R=',
*      E12.5)
END

```

Обчислення за цією програмою на комп'ютері ДВК-2М дають такий результат:

```

МАТР. А І ВЕКТ. ПРАВИХ ЧАСТ. В
0.10000E+01  -0.35000E+01  0.12250E+02  0.15000E+01
0.10000E+01  -0.34000E+01  0.11560E+02  0.18700E+01
0.10000E+01  -0.33000E+01  0.10890E+02  0.21400E+01
0.10000E+01  -0.32000E+01  0.10240E+02  0.32000E+01
0.10000E+01  -0.31000E+01  0.96100E+01  0.37500E+01
0.10000E+01  -0.30000E+01  0.90000E+01  0.42000E+01
0.10000E+01  -0.29000E+01  0.84100E+01  0.51000E+01
0.10000E+01  -0.28000E+01  0.78400E+01  0.57000E+01

```

```

| SIGMA
SIGMA(1)=0.30000E+02
SIGMA(2)=0.11013E-01
SIGMA(3)=0.71912E+00

```

```

МАТР. U ІЗ СИНГУЛ. ЗАПИС.
-0.42581E+00  -0.51122E+00  -0.51544E+00
-0.40296E+00  -0.57330E-01  -0.33778E+00
-0.38075E+00  0.24248E+00  -0.16847E+00
-0.35917E+00  0.38820E+00  -0.74829E-02
-0.33822E+00  0.37986E+00  0.14516E+00
-0.31791E+00  0.21744E+00  0.28948E+00
-0.29823E+00  -0.99064E-01  0.42545E+00
-0.27918E+00  -0.56963E+00  0.55309E+00

```

```

МАТР. V ІЗ СИНГУЛ. ЗАПИС.
-0.93409E-01  -0.84031E+00  0.53400E+00

```

0.29717E+00	-0.53543E+00	-0.79057E+00
-0.95024E+00	-0.84843E-01	-0.29973E+00

КОЕФІЦІЄНТИ C

C(1)=0.44187E+02
 C(2)=0.19724E+02
 C(3)=0.21428E+01

ЗНАЧ. СЕРЕДНЬОКВ. ВІДХ. R=0.40789E+00

Запитання для самоперевірки

1. Як ставиться задача наближення функцій в лінійному нормованому просторі?
2. Як розв'язується задача найкращого наближення функцій в гільбертовому просторі?
3. Чому при побудові елемента найкращого наближення на ЕОМ необхідно обережно користуватись системою функцій $\{(x - x_0)^{k-1}\}$, $k = \overline{1, n}$?
4. Які причини обумовили широке практичне застосування середньоквадратичних наближень?
5. За допомогою яких многочленів зручно будувати елемент найкращого наближення і чому?
6. Як вибираються на практиці вагові функції ρ_j , $j = \overline{1, n}$ при побудові елемента найкращого наближення, коли функцію задано в дискретній системі точок?

Контрольні завдання

1. Побудувати МНСН $Q_n(x)$ функції $y = k|x|^j$, $x \in [-1, 1]$ за допомогою многочленів Лежандра чи Чебишова першого роду, де $j = 1, 3, 5$; n, k — задані цілі числа.

2. Побудувати МНСН $Q_n(x)$ функції з попереднього завдання, виходячи з такої системи лінійно незалежних функцій:

$$U_{k-1}(x) = \frac{1}{k} T'_k(x), \quad k = 1, 2, \dots,$$

де $T_k(x) = \cos(k \arccos x)$ — многочлен Чебишова, ортогональні з ваговою функцією $\rho(x) = \sqrt{1-x^2}$.

3. Побудувати МНСН $Q_n(x)$ функції $f(x) = kx^2 + b \sin \omega x$ за її значеннями у вузлах $x_i = (i-1)h$, $i = \overline{1, m}$, $m \geq n$. Параметри x_1, h, k, ω, m, n задаються викладачем.

4. Побудувати МНСН $Q_n(x)$ функції $f(x) = k \ln x + b \cos \omega x$ за її значеннями у вузлах $x_i = x_1 + (i-1)h$, $i = \overline{1, m}$, $m \geq n$; $x_i \in [1/2, b]$. Параметри b, x_1, m, n задаються викладачем.

§ 10. Розв'язування задачі Коші для системи звичайних диференціальних рівнянь методом Рунге — Кутти

Постановка задачі. На відріжку $x_0 \leq x \leq \xi$ знайти розв'язок системи звичайних диференціальних рівнянь (ЗДР)

$$\begin{aligned} y_1' &= f_1(x, y_1, y_2, \dots, y_n); \\ &\dots \dots \dots \\ y_n' &= f_n(x, y_1, y_2, \dots, y_n). \end{aligned} \tag{1}$$

з початковими умовами

$$y_1(x_0) = y_1^{(0)}, \dots, y_n(x_0) = y_n^{(0)}. \quad (2)$$

Методичні вказівки. Система (1) з умовами (2) називається *задачею Коші* для системи ЗДР. Загальний розв'язок її в аналітичному вигляді можна знайти в окремих випадках лінійних систем із сталими коефіцієнтами. Тому універсальними методами розв'язування задачі Коші (кажуть ще інтегрування задачі Коші) є чисельні методи.

Для великого класу систем ЗДР, які не є жорсткими (див. § 11), найбільше поширення дістали явні однокрокові методи Рунге — Кутти [12, с. 255—272]. Розглянемо їх загальну схему.

Для кожної l -ї компоненти системи ЗДР (1) введено набір параметрів

$$\alpha_i^{(l)}, \beta_{i,j}^{(l)}, p_i^{(l)}, \quad l = 1, 2, \dots, n; \quad 0 < j < i \leq r^{(l)}.$$

Наближений розв'язок $\vec{y}_n(x+h)$ у точці $x+h$ шукатимемо у вигляді

$$y_n^{(l)}(x+h) = y_l(x) + \sum_{m=1}^{r^{(l)}} p_m^{(l)} k_m^{(l)}, \quad (3)$$

де

$$k_1^{(l)} = h f_l(x, y_1(x), \dots, y_n(x));$$

$$k_m^{(l)} = h f_l \left(x + \alpha_m^{(l)} h, y_1(x) + \sum_{j=1}^{m-1} \beta_{m,j}^{(l)} k_j^{(l)}, \dots, \right. \quad (4)$$

$$\left. y_n(x) + \sum_{j=1}^{m-1} \beta_{m,j}^{(n)} k_j^{(n)} \right), \quad l = 1, 2, \dots, n.$$

Часто параметри $\alpha_i^{(l)}, \beta_{i,j}^{(l)}, p_i^{(l)}$ вибираються однаковими для всіх значень $l = 1, 2, \dots, n$. У цьому разі принцип вибору їх найбільш зручно розглянути на прикладі скалярного рівняння, для якого формули (3), (4) мають вигляд

$$y_h(x+h) = y(x) + \sum_{m=1}^r p_m k_m; \quad (3a)$$

$$k_m = h f \left(x + \alpha_m h, y(x) + \sum_{j=1}^{m-1} \beta_{m,j} k_j \right). \quad (4a)$$

При фіксованому значенні r (як правило, значення r вибираються в межах $1 \leq r \leq 8$) параметри в методах Рунге — Кутти підбираються так, щоб забезпечити найбільшу точність наближеного розв'язку ЗДР за один крок обчислення.

Назвемо вираз

$$y(x+h) - y_h(x+h) = y(x+h) - y(x) - \sum_{i=1}^r p_i k_i = R(h) \quad (5)$$

похибкою методу на кроці. За формулою Тейлора маємо

$$R(h) = \sum_{k=0}^s \frac{h^k R^{(k)}(0)}{k!} + h^{(s+1)} \frac{R^{(s+1)}(\theta h)}{(s+1)!}. \quad (6)$$

Вибором параметрів p_i , α_i , β_{ij} забезпечимо виконання умови

$$R(0) = R'(0) = \dots = R^{(s)}(0) = 0 \quad (7)$$

до можливо більш високого порядку s .

Таким чином, умова (7) дає змогу побудувати клас методів s -го порядку точності (або $(s+1)$ -го порядку на кроці) і тим самим зафіксувати параметри їх. При цьому похибка

$$R(h) = \frac{h^{(s+1)} R^{(s+1)}(\theta h)}{(s+1)!} \quad (8)$$

є похибкою методів Рунге — Кутти s -го порядку точності на кроці.

▼ Однією з головних позитивних рис методів Рунге — Кутти є простота їх реалізації. Фактично для визначення наближеного розв'язку ЗДР у точці $x+h$ необхідно, вибравши крок інтегрування h , знайти наближений розв'язок $y_h(x+h)$ за формулами (3) та (4), використовуючи відомі значення $y_h(x)$. Після цього треба оцінити похибку наближеного розв'язку. Якщо необхідну точність розв'язку досягнуто, то слід прийняти цей крок, а якщо ні, то потрібно провести обчислення з меншим значенням h .

Разом з цим методи Рунге — Кутти мають і деякі недоліки. Головним з них є умова стійкості методів, тобто практичне використання їх можливе лише при досить малих кроках інтегрування h . Значення h визначається, з одного боку, варіантом методу, а з другого, власними значеннями матриці Якобі правих частин системи ЗДР.

Стійкість методів досліджується на модельному рівнянні [12, с. 271]

$$y' = -\lambda y, \quad \operatorname{Re} \lambda > 0.$$

Для апостеріорного оцінювання похибки наближеного розв'язку ЗДР на кроці застосовується правило Рунге або принцип «вкладених» методів [26, с. 365—367]. Розглянемо конкретний метод, який реалізовано в підпрограмі RKF45 [26, с. 146—164], що потребує шести обчислень правої частини на одному кроці. Чотири з них беруться з одним набором коефіцієнтів, завдяки чому досягається четвертий порядок точності розв'язку. Позначимо наближений розв'язок ЗДР за цим методом через $y_{1h}(x)$.

Усі шість значень k_i у комбінації з іншими коефіцієнтами дають метод п'ятого порядку точності.

Нехай розв'язком ЗДР за цим методом є $y_{2h}(x)$. Апостеріорною оцінкою похибки на кроці вважається величина $y_{1h}(x) - y_{2h}(x)$, яку використовують для контролю довжини кроку. В цьому й полягає суть принципу «вкладених» методів. Набір параметрів, які при цьому застосовуються, наведено в табл. 8.

Таблиця 8

α_i	β_{ij}					p_i	p_i^*
0						$\frac{16}{135}$	$\frac{25}{216}$
$\frac{1}{4}$	$\frac{1}{4}$					0	0
$\frac{3}{8}$	$\frac{3}{32}$	$\frac{9}{32}$				$\frac{6656}{12\ 825}$	$\frac{1408}{2565}$
$\frac{12}{13}$	$\frac{1932}{2197}$	$\frac{7200}{2197}$	$\frac{7296}{2197}$			$\frac{28\ 561}{56\ 480}$	$\frac{2197}{4104}$
1	$\frac{439}{216}$	-8	$\frac{3680}{513}$	$-\frac{845}{4104}$		$-\frac{9}{50}$	$-\frac{1}{5}$
$\frac{1}{2}$	$-\frac{8}{27}$	2	$-\frac{3544}{2565}$	$\frac{1859}{4104}$	$-\frac{11}{40}$	$\frac{2}{55}$	0

Наближений розв'язок ЗДР визначається виразом

$$y_{n+1} = y_n + \sum_{i=1}^6 p_i k_i,$$

який дає метод п'ятого порядку точності розв'язку. В той же час вираз

$$y_{n+1}^* = y_n + \sum_{i=1}^6 p_i^* k_i$$

застосовується для забезпечення четвертого порядку точності розв'язку. Але підпрограму RK45 складено так, що замість знаходження цього розв'язку визначається його похибка відповідно до виразу

$$\sum_{i=1}^6 (p_i - p_i^*) k_i.$$

Явні однокрокові методи Рунге — Кутти, для яких і розроблено підпрограму RK45, є ефективними для нежорстких або слабо жорстких систем ЗДР.

Приклади використання підпрограм. Приклад 1. Застосування підпрограми RK45 проілюструємо на розв'язуванні задачі Коші для такої системи ЗДР:

$$y_1' = \mu_0 y_1;$$

$$y_2' = (\mu_0 - \mu_1) y_1 + (\mu_1 + \nu_1) y_2 - \nu_1 y_3;$$

$$y_3' = (\mu_0 - \mu_1 - \nu_1) y_1 + 2\nu_1 y_2 + (\mu_1 - \nu_1) y_3; \quad (9)$$

$$y_4' = (\mu_0 - \mu_1 - \nu_1) y_1 + 2\nu_1 y_2 + (\mu_1 - \nu_1 - \mu_2) y_3 + 2\nu_2 y_4 + (\mu_2 - \nu_2) y_5;$$

$$y_5' = (\mu_0 - \mu_1 - \nu_1) y_1 + 2\nu_1 y_2 + (\mu_1 - \nu_1 - \mu_2 - \nu_2) y_3 + 2\nu_2 y_4 + (\mu_2 - \nu_2) y_5.$$

Покладемо

$$\mu_0 = -2, \mu_1 = -1, \mu_2 = -1, \nu_1 = 1, \nu_2 = 10 \quad (10)$$

і виберемо такий вектор початкових умов:

$$y_0^T = (1; 1,5; 1,5; 2,5; 2,5) \quad 0 \leq t \leq 1. \quad (11)$$

Обчислення виконаємо з абсолютною похибкою $\varepsilon_a = 10^{-2}$ та з відносною похибкою $\varepsilon_b = 10^{-3}$.

Знайдений наближений розв'язок можна порівняти з точним розв'язком системи ЗДР (9), який для початкових умов $y_2(0) = y_3(0)$, $y_4(0) = y_5(0)$ має вигляд

$$\begin{aligned} y_1(x) &= y_1(0) e^{\mu_0 x}; \\ y_2(x) &= y_1(x) + [y_2(0) - y_1(0)] e^{\mu_1 x} \cos v_1 x; \\ y_3(x) &= y_1(x) + \sqrt{2} [y_2(0) - y_1(0)] e^{\mu_1 x} \sin(v_1 x + \pi/4); \\ y_4(x) &= y_3(x) + [y_4(0) - y_2(0)] e^{\mu_2 x} \cos(v_2 x); \\ y_5(x) &= y_3(x) + \sqrt{2} [y_4(0) - y_2(0)] e^{\mu_2 x} \sin(v_2 x + \pi/4). \end{aligned} \quad (12)$$

Головна програма, яка містить звернення до підпрограми RKF45, а також до підпрограми SOL обчислення прямих частин системи ЗДР (9), і підпрограма YU визначення точного розв'язку (12) мають вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ                               С
С СИСТЕМИ ЗДР                                                 С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      EXTERNAL SOL
      COMMON/A/M0,M1,M2,N1,N2
      REAL Y(5),WORK(33),YT(5),Y0(5),
      *      M0,M1,M2,N1,N2
      INTEGER IWORK(5)
С ЗАДАВАННЯ КІЛЬКОСТІ РІВНЯНЬ
      NEQN=5
С ЗАДАВАННЯ ПАРАМЕТРІВ СИСТЕМИ
      TYPE *, 'M0='
      ACCEPT *, M0
      TYPE *, 'M1='
      ACCEPT *, M1
      TYPE *, 'M2='
      ACCEPT *, M2
      TYPE *, 'N1='
      ACCEPT *, N1
      TYPE *, 'N2='
      ACCEPT *, N2
      TYPE *, 'NF='
      ACCEPT *, NF
      WRITE(NF,100)M0,M1,M2,N1,N2
С ЗАДАВАННЯ ПОЧАТКОВИХ ЗНАЧЕНЬ
      TYPE *, 'Y(1)='
      ACCEPT *, Y(1)
      TYPE *, 'Y(2)='
      ACCEPT *, Y(2)
      TYPE *, 'Y(3)='
      ACCEPT *, Y(3)
      TYPE *, 'Y(4)='
      ACCEPT *, Y(4)
      TYPE *, 'Y(5)='

```

```

ACCEPT*,Y(5)
DO 10 I = 1,NEQN
10   Y0(I)=Y(I)
C   ЗАДАВАННЯ ЗНАЧЕННЯ АБСОЛЮТНОЇ ТА ВІДНОСНОЇ ПОХИБОК;
C   ЗНАЧЕННЯ RELERR МАЄ БУТИ ЗМІННОЮ, ЗНАЧЕННЯ
C   ABSERR МОЖЕ БУТИ КОНСТАНТОЮ;
C   ЗАДАВАННЯ ЗНАЧЕННЯ ПОЧАТКОВОЇ ТА КІНЦЕВОЇ ТОЧОК
C   ІНТЕГРУВАННЯ;
C   ЗАДАВАННЯ КРОКУ ВИДАЧІ РЕЗУЛЬТАТІВ
      TYPE *,RELERR,ABSERR,T,TFINAL,TPRINT-?
      ACCEPT *,RELERR,ABSERR,T,TFINAL,TPRINT
      IFLAG=1
      TOUT=T
      WRITE(NF,101)
      WRITE(NF,102)
20   CALL RKF45(S0L,NEQN,Y,T,TOUT,RELERR,ABSERR,
      * IFLAG,WORK,IWORK)
      CALL YY(T,Y0,YT)
      TYPE 103,T
      WRITE(NF,104)T,Y
      WRITE(NF,104)T,YT
      GO TO (90,30,40,50,60,70,80,90),IFLAG
C   ОБЧИСЛИТИ РОЗВ'ЯЗОК У ТОЧЦІ TOUT=T+TPRINT
30   TOUT=T+TPRINT
C   ПЕРЕВІРИТИ ДОСЯГНЕННЯ КІНЦЯ ПРОМІЖКУ ІНТЕГРУВАННЯ
      IF(T.LT.TFINAL) GO TO 20
      STOP
40   WRITE (NF,105) RELERR,ABSERR
      GO TO 20
50   WRITE (NF,106)
      GO TO 20
60   ABSERR=1.0E-7
      WRITE (NF,105) RELERR,ABSERR
      GO TO 20
70   RELERR=10.0*RELERR
      WRITE(NF,105) RELERR,ABSERR
      IFLAG=2
      GO TO 20
80   WRITE(NF,107)
      IFLAG=2
      GO TO 20
90   WRITE(NF,108)
      STOP
100  FORMAT(2X,'M0=',E12.5,2X,'M1=',E12.5,2X,
      * 'M2=',E12.5/5X,'N1=',E12.5,2X,'N2=',E12.5)
101  FORMAT(/20X,'НАБЛИЖЕНИЙ РОЗВ'ЯЗОК')
102  FORMAT(3X,'T',6X,'Y1',7X,'Y2',7X,'Y3',7X,
      * 'Y4',7X,'Y5')
103  FORMAT(2X,'T=',E12.5)
104  FORMAT(1X,F5.3,5(1X,F8.5))
105  FORMAT(5X,'ВЕЛИКІ ВИМОГИ ДО ТОЧНОСТІ',
      * 2X,2E15.5)
106  FORMAT(5X,'ДУЖЕ МАЛИЙ КРОК ІНТЕГРУВАННЯ')
107  FORMAT(5X,'ВЕЛИКЕ ВИВЕДЕННЯ ІНФОРМАЦІЇ')
108  FORMAT(5X,'НЕПРАВИЛЬНИЙ ВИКЛИК RKF45')
      END

```

```

FUNCTION SOL(T,Y,YP)
С ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯННЯ
REAL M0,M1,M2,N1,N2,Y(5),YP(5)
COMMON /A/M0,M1,M2,N1,N2
YP(1)=M0*Y(1)
YP(2)=(M0-M1)*Y(1)+(M1+N1)*Y(2)-N1*Y(3)
YP(3)=(M0-M1-N1)*Y(1)+2.*N1*Y(2)+(M1-N1)*Y(3)
YP(4)=(M0-M1-N1)*Y(1)+2.*N1*Y(2)+(M1-N1-M2)*
* Y(3)+(M2+N2)*Y(4)-N2*Y(5)
* YP(5)=(M0-M1-N1)*Y(1)+2.*N1*Y(2)+(M1-N1-M2-
* N2)*Y(3)+2.*N2*Y(4)+(M2-N2)*Y(5)
RETURN
END

```

```

FUNCTION YY(T,Y0,Y)
С ОБЧИСЛЕННЯ ТОЧНОГО РОЗВ'ЯЗКУ РІВНЯННЯ
REAL M0,M1,M2,N1,N2,Y0(5),Y(5)
COMMON /A/M0,M1,M2,N1,N2
PI=4.*ATAN(1.)
Y(1)=Y0(1)*EXP(M0*T)
Y(2)=Y(1)+(Y0(2)-Y0(1))*EXP(M1*T)*COS(N1*T)
Y(3)=Y(1)+SQRT(2.)*(Y0(2)-Y0(1))*EXP(M1*T)*
* SIN(N1*T+PI/4.)
Y(4)=Y(3)+(Y0(4)-Y0(2))*EXP(M2*T)*COS(N2*T)
Y(5)=Y(3)+SQRT(2.)*(Y0(4)-Y0(2))*EXP(M2*T)*
* SIN(N2*T+PI/4.)
RETURN
END

```

Результат обчислень на комп'ютері ДВК-2М зведено в таку таблицю:
 $M0 = -0.20000E+01$ $M1 = -0.10000E+01$ $M2 = -0.10000E+01$
 $N1 = 0.10000E+01$ $N2 = 0.10000E+02$

НАБЛИЖЕНИЙ РОЗВ'ЯЗОК					
T	Y1	Y2	Y3	Y4	Y5
0.000	1.00000	1.50000	1.50000	2.50000	2.50000
0.000	1.00000	1.50000	1.50000	2.50000	2.50000
0.100	0.81873	1.26889	1.31406	1.80295	2.56435
0.100	0.81873	1.26889	1.31406	1.80294	2.56434
0.200	0.67032	1.07153	1.15285	0.81213	1.55661
0.200	0.37032	1.07153	1.15285	0.81214	1.55661
0.300	0.54881	0.90268	1.01214	0.27871	0.38325
0.300	0.54881	0.90268	1.01214	0.27874	0.38328
0.400	0.44933	0.75803	0.88855	0.45039	-0.05694
0.400	0.44933	0.75803	0.88855	0.45040	-0.05690
0.500	0.36788	0.63402	0.77941	0.95148	0.36984
0.500	0.36788	0.63402	0.77941	0.95146	0.36985
0.600	0.30119	0.52767	0.68261	1.20960	1.05625
0.600	0.30119	0.52767	0.68261	1.20957	1.05622
0.700	0.24660	0.43650	0.59646	0.97086	1.29715
0.700	0.24660	0.43650	0.59646	0.97083	1.29708
0.800	0.20190	0.35842	0.51959	0.45419	0.89878
0.800	0.20190	0.35842	0.51959	0.45421	0.89876
0.900	0.16530	0.29166	0.45090	0.08042	0.24798
0.900	0.16530	0.29166	0.45090	0.08046	0.24802
1.000	0.13534	0.23472	0.38950	0.08079	-0.11938
1.000	0.13534	0.23472	0.38950	0.08082	-0.11931

Непарні рядки її містять наближений розв'язок, а парні — точний.

Приклад 2. Знайдемо наближений розв'язок системи ЗДР (9), якщо

$$\mu_0 = -100, \quad \mu_1 = -1, \quad \mu_2 = -1 \cdot 10^5, \quad \nu_1 = 1, \quad \nu_2 = 100, \quad (10a)$$

а вектор початкових умов має вигляд

$$y_0^T = (10, 11, 11, 111, 111). \quad (11a)$$

Обчислення виконаємо з абсолютною похибкою $\varepsilon_a = 0$ та відносною похибкою $\varepsilon_B = 10^{-3}$.

За цих вхідних даних задача Коші є жорсткою, що проявляється в значному зростанні часу інтегрування, а також в переході підпрограми RK45 на покроковий режим інтегрування, внаслідок чого її ефективність стосовно розглядуваної задачі є малою.

Запитання для самоперевірки

1. Як можна використати підпрограму RK45 при розв'язуванні задачі Коші для рівняння n -го порядку $y^{(n)} = F(x, y, y^{(1)}, \dots, y^{(n-1)})$, $x_0 < x < \xi$, $y(x_0) = y_0$, $y'(x_0) = y_{1,0}, \dots, y^{(n-1)}(x_0) = y_{n-1,0}$? 2. Чому методи, розглянуті в § 10, називаються явними? Який ви знаєте приклад неявного однокрокового методу? 3. Як застосувати правило Рунге для оцінювання похибки методу Рунге — Кутти четвертого порядку точності? Який вигляд мають відповідні розрахункові формули? 4. Як можна провести дослідження стійкості методу Рунге — Кутти другого порядку точності

$$y_{n+1} = y_n + hf(t_n + h/2, y_n + k_1/2), \quad k_1 = hf(t_n, y_n) \quad (13)$$

на модельному рівнянні $y' = -\lambda y$, $\text{Re} \lambda > 0$? 5. Як можна за допомогою методу Рунге — Кутти (13) побудувати систему вкладених методів першого-другого порядку точності? Як при цьому оцінюється похибка методу?

Контрольні завдання

1. Процес нанесення тонких рідких плівок на поверхню, що витягується з в'язкої рідини, описується задачею Коші для такого диференціального рівняння другого порядку:

$$y'' = -\frac{2P(H, \alpha)}{\sqrt{y}}, \quad P(H, \alpha) = 2 + \frac{3\alpha - 2\alpha^3}{H^3} - \frac{3}{H^2}; \quad (14)$$

$$y(\alpha) = y'(\alpha) = 0, \quad 0 < \alpha < 1/\sqrt{2}.$$

На відріжку $[\alpha, \alpha]$ знайти розв'язок задачі Коші з відносною похибкою $\varepsilon_B = 10^{-4}$ та абсолютною похибкою $\varepsilon_a = 10^{-5}$. Чисельно дослідити характер залежності цього розв'язку від параметра α .

Зуваження. Точка $H = \alpha$ є особливою точкою рівняння другого порядку

$$\lim_{H \rightarrow \alpha} -\frac{2P(H, \alpha)}{\sqrt{y}} = \frac{0}{0}.$$

Цю невизначеність необхідно розкрити, використовуючи розкладання розв'язку в ряд по степенях $(H - \alpha)$ в околі особливої точки. Легко перевірити, що при досить малих значеннях $(H - \alpha)$ справджується рівність

$$y(H, \alpha) = \left(\frac{3 - 6\alpha^2}{\alpha^3} \right)^{2/3} (H - \alpha)^2 + 0 (H - \alpha)^3. \quad (15)$$

Інтегрувати (14), користуючись цією рівністю, треба, починаючи з точки $H = \alpha + \varepsilon$.

2. Траєкторія руху ракети невеликого радіуса дії при відсутності бокового вітру описується такою системою двох ЗДР другого порядку:

$$\begin{aligned}\ddot{x} &= \frac{1}{m} (T - 0,5c\rho s v^2) \cos \theta - \frac{\dot{m}}{m} \dot{x}; \\ \ddot{y} &= \frac{1}{m} (T - 0,5c\rho s v^2) \sin \theta - \frac{\dot{m}}{m} \dot{y} - g,\end{aligned}\quad (16)$$

де

$$v(t) = [\dot{x}^2(t) + \dot{y}^2(t)]^{1/2}; \quad \theta(t) = \arctg(\dot{y}(t)/\dot{x}(t)), \quad (17)$$

за початкових умов

$$x(0) = x_0, \quad y(0) = y_0, \quad v(0) = v_0, \quad \theta(0) = \theta_0. \quad (18)$$

Тут $x(t)$, $y(t)$ — горизонтальна та вертикальна координати ракети в момент часу t ; $m = m(t)$ — відома функція, що описує масу ракети в момент часу t ; $T = T(t)$ — відома функція, що описує силу тяжіння ракети в момент часу t ; $c = c(y)$, $\rho = \rho(y)$ — відомі функції опору й густини повітря; s — площа поперечного перерізу ракети; $\theta(t)$ — кут між вектором швидкості та площиною землі; $g = 9,8 \text{ м/с}^2$ — прискорення вільного падіння.

Виходячи з цього, потрібно:

а) обчислити траєкторію руху ракети з моменту часу $t = 0$ до моменту її удару об поверхню землі. Знайти висоту і дальність польоту ракети. Вихідні дані наведено в табл. 9;

б) знайти траєкторію польоту ракети з моменту пуску $t = 0$ до моменту досягнення нею висоти H . Обчислити швидкість і кут θ між вектором швидкості та лінією горизонту в точках $H/4$, $H/2$, $3H/4$. Вихідні дані наведено в табл. 10.

Значення $T_i(t)$, $m_i(t)$ обчислюються за формулами

$$\begin{aligned}T_i(t) &= \begin{cases} T_0^{(i)} \left(1 - \left(\frac{t - t_0^{(i)}}{t_0^{(i)}} \right)^2 \right), & 0 \leq t \leq 2t_0^{(i)}; \\ 0, & t \geq 2t_0^{(i)}; \end{cases} \\ m_i(t) &= \begin{cases} m_3^{(i)} + m_T^{(i)} \left(1 - \left(\frac{t}{2t_0^{(i)}} \right)^2 \right), & 0 \leq t \leq 2t_0^{(i)}; \\ m_3^{(i)}, & t > 2t_0^{(i)}. \end{cases}\end{aligned}$$

Значення безрозмірних параметрів $m_3^{(i)}$, $m_T^{(i)}$, $T_0^{(i)}$, $t_0^{(i)}$ наведено в табл. 11.

3. Динаміка популяції двох видів тварин, які взаємодіють між собою за типом «хижак — жертва», описується системою таких двох ЗДР:

$$\begin{aligned}\frac{dx}{dt} &= \alpha x + \beta xy; \\ \frac{dy}{dt} &= \gamma y + \delta xy\end{aligned}\quad (19)$$

в початковими умовами

$$x(0) = x_0, \quad y(0) = y_0; \quad \alpha = x_b - x_d > 0, \quad \beta < 0, \quad \gamma < 0, \quad \delta > 0, \quad (20)$$

Таблиця 9

Номер ва-ріанта	x_0 , м	y_0 , м	v_0 , м/с	θ_0 , рад	m , кг	T , Н	$c(y)$	$\rho(y)$, кг/м ³	ε_B	ε_d
1	0	0	50	0,6	15	0	0,2	1,29	10^{-3}	10^{-1}
2	0	5	75	1,2	15	0	0,2	1,29	10^{-3}	10^{-1}
3	0	0	100	1,2	10	0	$\frac{0,1}{1+y}$	$\frac{1,29}{1+0,5y}$	10^{-3}	10^{-1}
4	0	200	25	0	50	0	$\frac{0,1}{1+2y}$	$\frac{1}{1+2y}$	10^{-3}	10^{-1}
5	0	500	40	-0,1	60	0	$\frac{0,5}{1+10y}$	$\frac{1,3}{1+3y}$	10^{-3}	10^{-1}

Таблиця 10

Номер ва-ріанта	x_0 , м	y_0 , м	v_0 , м/с	θ_0 , рад	$m(t)$, кг	$T(t)$, Н	$c(y)$	$\rho(y)$, кг/м ³	H , м	ε_B	ε_d
1	0	0	0	0,6	$m_1(t)$	$T_1(t)$	0,2	1,3	500	10^{-3}	10^{-2}
2	0	100	0	0,6	$m_2(t)$	$T_2(t)$	0,2	1,2	800	10^{-3}	10^{-1}
3	0	100	10	1,2	$m_3(t)$	$T_3(t)$	$\frac{0,2}{1+0,25y}$	$\frac{1,29}{1+y}$	1000	10^{-2}	10^{-2}
4	0	1500	15	0	$m_4(t)$	$T_4(t)$	$\frac{0,1}{1+2y}$	$\frac{1,29}{1+2y}$	800	10^{-2}	10^{-2}

де $x(t)$, $y(t)$ — кількість особин у популяціях жертв і хижаків; x_0 — норма народження жертв; x_d — норма природної смертності жертв; γ — коефіцієнт природного зменшення кількості хижаків; β — коефіцієнт зменшення популяції жертв через взаємодію з хижаком; δ — коефіцієнт зростання популяції хижаків через взаємодію з жертвами.

Визначити кількість популяції хижаків і жертв за даними, наведеними в табл. 12. Знайти стаціонарний розв'язок розглядуваної задачі, тобто розв'язок для $t \rightarrow \infty$.

4. Літак летить горизонтально зі швидкістю V на висоті h над поверхню землі. В деякий момент часу пілот літака катапультиється. Визначити початкову швидкість катапульти V_l , при якій вона не зіткнеться з хвостовим оперенням літака. Рів-

Таблиця 11

t	$m_3^{(i)}$	$m_T^{(i)}$	$T_0^{(i)}$	$t_0^{(i)}$
1	100	200	2000	0,1
2	150	250	50	20
3	100	300	100	10
4	200	400	150	300
5	250	250	500	100

Таблиця 12

Номер варіанта	x_0	y_0	α	β	γ	δ
1	200	200	0,25	-0,01	-1,0	0,01
2	300	500	0,5	-0,001	-2,0	0,05
3	700	10	0,1	-0,1	-3,5	0,1
4	150	10	1,5	-0,5	-2,0	0,3
5	10	15	0,7	-0,1	-1,5	0,25

няння руху системи «літак — катапульта» в системі координат, зв'язаної з центром мас катапульти з пілотом, мають вигляд

$$\frac{dV}{dt} = \frac{-0,5\gamma c s V^2}{M} - g \sin \theta; \quad \frac{dy}{dt} = V \sin \theta;$$

$$\frac{d\theta}{dt} = -\frac{g \cos \theta}{V}; \quad \frac{dx}{dt} = V \cos \theta - V_l,$$

а початковими умовами є такі:

$$x(0) = y(0) = 0; \quad V(0) = [(V_0 - V_l \sin \theta_l)^2 + (V_l \cos \theta_l)^2]^{1/2};$$

$$\theta(0) = \operatorname{tg}^{-1} \frac{V_l \cos \theta_l}{V_0 - V_l \sin \theta_l},$$

де γ — густина повітря; c — коефіцієнт його опору; s — площа опору катапульти з пілотом; g — прискорення вільного падіння; M — маса катапульти з пілотом; θ — кут нахилу траєкторії швидкості катапульти; x, y — її координати; θ_l — початковий кут катапультивання.

Катапультивання можна вважати безпечним, якщо при $x = 5$ м координата $y \geq 4$ м (висота хвостового оперення плюс 1,5 м).

Виконати обчислення $M = 150$ кг, $V_l = 12$ м/с, $c = 1$, $\theta_l = 15^\circ$, $s = 1$ м², $g = 9,8$ м/с². Залежність густини повітря γ від висоти польоту літака h має вигляд:

$h, \text{ м}$	0	100	200	600	1000	2000	6000	8000	10 000	12 000
$\gamma \cdot 10^3, \text{ кг/м}^3$	2,377	2,308	2,241	2,117	1,987	1,447	1,126	0,871	0,857	0,304

Дослідити безпеку катапультивання при зміні швидкості літака від 30 до 300 м/с і висоти його польоту від рівня моря до 10 000 м з кроком 100 м.

§ 11. Задача Коші для жорстких систем звичайних диференціальних рівнянь

Постановка задачі. Нехай на деякому відрізку $t_0 \leq t \leq T$ необхідно знайти розв'язок задачі Коші для такої системи l ЗДР:

$$\begin{aligned} y_i^{(1)} &= f_1(t, y_1, y_2, \dots, y_l); \\ y_i^{(2)} &= f_2(t, y_1, y_2, \dots, y_l); \\ y_i^{(l)} &= f_l(t, y_1, y_2, \dots, y_l) \end{aligned} \quad (1)$$

з початковими умовами

$$y_1(t_0) = y_{1,0}, \quad y_2(t_0) = y_{2,0}, \quad \dots, \quad y_l(t_0) = y_{l,0}.$$

У векторній формі розглядувана задача записується так:

$$\vec{y}'(t) = \vec{f}(t, \vec{y}); \quad \vec{y}(t_0) = \vec{y}_0, \quad (2)$$

де y, f, y_0 — l -вимірні вектор-функції.

Із системи ЗДР (1) виділимо клас жорстких систем. Жорсткість системи ЗДР було виявлено в зв'язку з неефективністю використання методів Рунге — Кутти для її розв'язування.

Розглянемо досить простий приклад жорсткої системи ЗДР, а саме:

$$\begin{aligned}\frac{dy_1}{dt} &= -501y_1 + 500y_2; \\ \frac{dy_2}{dt} &= 500y_1 - 501y_2.\end{aligned}\quad (3)$$

Розв'язок цієї системи має вигляд

$$\begin{aligned}y_1(t) &= 0,5 [y_1(0) - y_2(0)] e^{-1001t} + 0,5 [y_1(0) + y_2(0)] e^{-t}; \\ y_2(t) &= -0,5 [y_1(0) - y_2(0)] e^{-1001t} + 0,5 [y_1(0) + y_2(0)] e^{-t}.\end{aligned}\quad (4)$$

Головна особливість системи ЗДР (3) полягає в тому, що спектральне число її обумовленості є досить великим, причому визначається воно виразом

$$K(A) = \frac{\max_i |\lambda_i|}{\min_i |\lambda_i|} \quad (5)$$

і для цієї задачі набуває значення $K(A) = 1001$, де A — матриця розглядуваної лінійної системи ЗДР; λ_i — власні числа матриці A з номером i .

При досить малому інтервалі $0 \leq t \leq t_n$ розв'язок (4) визначається максимальним за модулем власним числом матриці, але, не дивлячись на це, в усьому інтервалі інтегрування допустимий крок інтегрування явним методом Рунге — Кутти визначається значенням λ_1 , тобто $h \leq c / |\lambda_1|$, де c — стала, що залежить від застосовуваного методу.

Очевидно, таке обмеження кроку інтегрування h є досить жорстким і, як правило, визначається умовою стійкості методу, а не вимогою його точності.

ОЗНАЧЕННЯ. Система ЗДР (1) називається *жорсткою* на відріжку $[a, b]$, який належить інтервалу існування її розв'язку, якщо при довільному векторі початкових значень (t_0, y_0) і на довільному відріжку $[t_0, t_0 + T] \subset [a, b]$ знайдуться такі числа τ_n, L, N , які задовольняють умови

$$\begin{aligned}\tau_n &\ll (b - a); \\ 0 < L &\leq \rho \left(\frac{\partial f}{\partial x} \right) \leq \left\| \frac{\partial f}{\partial x} \right\|,\end{aligned}$$

що

$$\left| \frac{dy_k}{dt} \right|_{t > t_0 + \tau_n} \leq \frac{L}{N} \max_{t \in [t_0, t_0 + T]} |y_k(t)|, \quad k = \overline{1, l}; \quad (6)$$

$$t_0 + \tau_n \leq t \leq t_0 + T, \quad N \gg 1, \quad (7)$$

де $\rho \left(\frac{\partial f}{\partial x} \right)$ — спектральний радіус матриці Якобі.

Методичні вказівки. При інтегруванні жорстких систем ЗДР на великих інтервалах зміни незалежної змінної, коли явні однокрокові методи Рунге — Кутти стають неефективними, слід використовувати інший клас методів, зокрема неявні одно- та багатокрокові методи.

Розглянемо клас таких методів, які зручно подати у вигляді

$$y_n = \sum_{j=1}^{k_1} \alpha_j y_{n-j} + h \sum_{j=0}^{k_2} \beta_j y'_{n-j}. \quad (8)$$

До цих методів належать як традиційні багатокрокові методи Адамса q -го порядку ($k_1 = 1, k_2 = q - 1$), які є ефективними для розв'язування нежорстких систем рівнянь, так і методи Гіра порядку q ($k_1 = q, k_2 = 0$), а також їх модифікації. У всіх випадках α_j, β_j — коефіцієнти, що визначаються використанням методом. При $\beta_0 \neq 0$ рівняння (8) стає неявним відносно y_n .

Методи вигляду (8) реалізовано в підпрограмі GEARZ [9], причому порядок методів Адамса дорівнює 12, а Гіра — 5. Ця підпрограма є деякою модифікацією підпрограми GEAR [27]. Розроблено ще кілька підпрограм, близьких за своєю ідеологією до підпрограми GEAR. Серед них виділимо такі, як STIFF [7], ERISODE [29], GENRZ [8], LSODE [28].

Вирази коефіцієнтів α_j, β_j мають досить громіздкий вигляд. Значення їх можуть бути знайдені в масиві EL підпрограми COSET. Метод (8) можна записати у вигляді

$$q(y_n, t_n) = y_n - h\beta_0 f(y_n, t_n) - \sum_{j=1}^{k_1} \alpha_j y_{n-j} - h \sum_{j=1}^{k_2} \beta_j y'_{n-j} = 0. \quad (9)$$

Як і у (8), тут y_n є вектором наближеного розв'язку системи ЗДР (1) у точці t_n .

Для розв'язування нелінійної системи рівнянь (9) у пакеті підпрограм GEARZ застосовуються методи простої ітерації, Ньютона та його модифікації. Так, ітераційний процес Ньютона можна відобразити рівнянням

$$y_n^{(m+1)} = y_n^{(m)} - (P_n^{(m)})^{-1} q(y_n^{(m)}, t_n), \quad (10)$$

де

$$P_n^{(m)} = \left. \frac{\partial q}{\partial y} \right|_{\substack{y=y_n^{(m)} \\ t=t_n}} = I - h\beta_0 \left. \frac{\partial f}{\partial y} \right|_{\substack{y=y_n^{(m)} \\ t=t_n}} \quad (11)$$

— матриця Якобі.

Серед модифікацій методу Ньютона використовують такі, в яких матриця обчислюється чисельним диференціюванням. Застосовується також діагональна апроксимація матриці Якобі за формулою

$$\begin{aligned} P_n &\approx I - h\beta_n D_n, \quad D_n = \text{diag}(d_1, \dots, d_l); \\ d_i &= [f_i(y_n^{(0)} + u, t_n) - f_i(y_n^{(0)}, t_n)]/u_i; \\ u &= (u_i)_{i=1, \dots, l} = -0,1q(y_n^{(0)}). \end{aligned} \quad (12)$$

У методі простої ітерації

$$P_n = I. \quad (13)$$

При використанні будь-якого ітераційного методу потрібно знати початкове наближення $y_n^{(0)}$ — предиктор, який визначається з точністю порядку q на підставі відомої інформації про попередні $k_1 + k_2 + 1 = q + 1$ значення функції y_h та її похідної y'_k , що використовуються в методі (8). Побудувавши таким способом інтерполяційний многочлен степеня q , за його допомогою можна обчислити матрицю

$$z_n = (y_n, hy'_n, h^2 y''_n/2!, \dots, h^q y_n^{(q)}/q!)^T, \quad (14)$$

яка називається *матрицею (предиктором) Нордсіка*. Її компоненти, звичайно, є неточними, проте вони апроксимують відповідні похідні точного розв'язку.

Для визначення компонентів вектора розв'язку \vec{y}_n і його похідних $h^i y_n^{(i)}/i!$, $i = \overline{1, q}$ (тобто матриці Нордсіка) в підпрограмі застосовуються співвідношення

$$z_n^{(0)} = Az_{n-1};$$

$$y_n^{(m+1)} = y_n^{(m)} + l_0 P_n^{-1} F^{(m)}; \quad (15)$$

$$F^{(m)} = hf(y_n^{(m)}, t_n) - hy_n^{(m)};$$

$$y_n = y_n^{(M)} = y_n^{(0)} + l_0 E_M^{(n)}; \quad E_M^{(n)} = \sum_{m=0}^{M-1} P_n^{-1} F^{(m)}; \quad (16)$$

$$z_n = z_n^{(0)} + E_M^{(n)} \vec{l},$$

де M — кількість виконаних ітерацій; $\vec{l} = (l_i)^q$ — вектор-стовпець, який визначає основний метод (8).

Добуток $P_n^{-1} F^{(m)}$ обчислюється за допомогою LU -факторизації, де L і U — трикутні матриці.

Обчислення на кожному кроці матриці Нордсіка дає змогу змінювати крок інтегрування без використання інтерполяцій, легко переходити до методу іншого порядку, а також значно спростити стартову процедуру для застосування багатокрокових методів. Підпрограма GEARZ починає інтегрування системи ЗДР з використання методу першого порядку, поступово збільшуючи його до максимально допустимого. При цьому застосовується деяка стратегія вибору змінного порядку й кроку інтегрування, що при заданій локальній точності забезпечує максимально допустимий крок інтегрування.

Оцінювання локальної похибки ε_{q-1} , ε_q , ε_{q+1} методів $(q-1)$ -, q - та $(q+1)$ -го порядків точності проводиться за допомогою різницевої апроксимації головного члена похибки

$$\varepsilon_q = c_{q+1}^* h^{q+1} \|\vec{y}^{(q+1)}\| + O(h^{q+2}), \quad (17)$$

де c_{q+1}^* — складова локальної похибки дискретизації на кроці, яка задається в підпрограмі для відповідного методу, причому

$$\begin{aligned} e_{q-1} &\approx |c_q^*| \|\vec{z}_n\|_{q+1} \|q\|; \\ e_q &\approx |c_{q+1}^*| \|\nabla \vec{z}_n\|_{q+1} \|q\|; \\ e_{q+1} &\approx |c_{q+2}^*| \|\nabla^2 \vec{z}_n\|_{q+1} \|q\|, \end{aligned} \quad (18)$$

де

$$\begin{aligned} \nabla \vec{z}_n &= h^q (y_n^{(q)} - y_{n-1}^{(q)})/q! \\ \nabla^2 \vec{z}_n &= \nabla \vec{z}_n - \nabla \vec{z}_{n-1}. \end{aligned}$$

Приклади використання підпрограми. Методику застосування підпрограми GEARZ для інтегрування жорсткої системи ЗДР розглянемо на прикладі головної програми TSTKL, яка дає змогу інтегрувати систему лінійних диференціальних рівнянь

$$y_1' = \mu_0 y_1;$$

$$y_2' = (\mu_0 - \mu_1) y_1 + (\mu_1 + \nu_1) y_2 - \nu_1 y_3;$$

$$y_3' = (\mu_0 - \mu_1 - \nu_1) y_1 + 2\nu_1 y_2 + (\mu_1 - \nu_1) y_3;$$

$$y_4' = (\mu_0 - \mu_1 - \nu_1) y_1 + 2\nu_1 y_2 + (\mu_1 - \nu_1 - \mu_2) y_3 + (\mu_2 + \nu_2) y_4 - \nu_2 y_5; \quad (19)$$

$$y_5' = (\mu_0 - \mu_1 - \nu_1) y_1 + 2\nu_1 y_2 + (\mu_1 - \nu_1 - \mu_2 - \nu_2) y_3 + 2\nu_2 y_4 - (\mu_2 - \nu_2) y_5$$

з початковими умовами

$$y_j(0) = c_j; \quad i = \overline{1, 5}; \quad c_2 = c_3; \quad c_4 = c_5.$$

При різних значеннях параметрів задачі Коші (19) дістаємо тести з різною поведінкою її розв'язку, а саме:

1. Погано обумовлена система, коли $c_1 = 0,1$; $c_2 = c_3 = 1$; $c_4 = c_5 = 0,5$; $\mu_0 = 10$, $\mu_1 = 4$, $\mu_2 = 5$; $\nu_1 = 20\pi$, $\nu_2 = 100$.

2. М'яка система, коли $c_1 = 1$; $c_2 = c_3 = 1,5$; $c_4 = c_5 = 2,5$; $\mu_0 = -2$, $\mu_1 = 1$, $\mu_2 = -1$; $\nu_1 = 1$, $\nu_2 = 10$.

3. Швидко осцилююча система, коли $c_1 = 0,5$; $c_2 = c_3 = 0,8$; $c_4 = c_5 = 2$; $\mu_0 = -2$, $\mu_1 = 1$, $\mu_2 = -1$; $\nu_1 = 1$, $\nu_2 = 1000$.

4. Жорстка система 1, коли $c = 10$, $c_2 = c_3 = 11$, $c_4 = c_5 = 111$; $\mu_0 = -100$, $\mu_1 = -1$, $\mu_2 = -10\,000$; $\nu_1 = 1$, $\nu_2 = 10$.

5. Жорстка система 2, коли $c_1 = 10$, $c_2 = c_3 = 11$, $c_4 = c_5 = 111$; $\mu_0 = -100$, $\mu_1 = -1$, $\mu_2 = -10\,000$; $\nu_1 = 1$, $\nu_2 = 10\,000$.

6. Жорстка осцилююча система, коли $c_1 = 100$, $c_2 = c_3 = 101$, $c_4 = c_5 = 201$; $\mu_0 = -1 \cdot 10^5$, $\mu_1 = 1$, $\mu_2 = -100$; $\nu_1 = 1$, $\nu_2 = 1000$.

Щоб розв'язати одну з цих задач, досить у режимі діалога ввести з терміналу величини: NUM — номер системи 1–6; MF — ціле двозначне число, що визначає метод інтегрування та спосіб обчислення матриці Якобі; EPS — потрібну локальну

точність інтегрування; DELTAT — параметр, який визначає інтервал видавання результату.

При MF = 0 за програмою TSTKL проводиться інтегрування задачі Коші методом Рунге — Кутти четвертого порядку точності; при MF < 0 — обчислення й друкування точного розв'язку задачі Коші, який має вигляд

$$y_1(t) = c_1 e^{\mu_0 t};$$

$$y_2(t) = y_1(t) + (c_2 - c_1) e^{\mu_1 t} \cos(v_1 t);$$

$$y_3(t) = y_1(t) + \sqrt{2} (c_2 - c_1) e^{\mu_1 t} \sin(v_1 t + \pi/4);$$

$$y_4(t) = y_3(t) + (c_4 - c_2) e^{\mu_2 t} \cos(v_2 t);$$

$$y_5(t) = y_3(t) + \sqrt{2} (c_4 - c_2) e^{\mu_2 t} \sin(v_2 t + \pi/4),$$

причому

$$MF = 10 \cdot METH + MITER,$$

де METH — перша цифра числа (METH = 1 — формули Адамса; METH = 2 — жорстко стійкі формули Гіра; METH = 3 — жорстко стійкі модифікації цих формул); MITER — друга цифра числа, що вказує на спосіб розв'язування невязних рівнянь коректора (MITER = 0 — метод простих ітерацій; MITER = 1 — метод Ньютона з аналітичним записом матриці Якобі та використанням підпрограми PEDERV; MITER = 2 — метод Ньютона з автоматичним обчисленням цієї матриці чисельним диференціюванням; MITER = 3 — метод Ньютона з діагональною апроксимацією матриці Якобі).

За допомогою тестуючої програми TSTKL визначається також і глобальна похибка розв'язку ЗДР.

Тестуюча програма TSTKL має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА TSTKL. РОЗГЛЯДАЄТЬСЯ ШІСТЬ С
C ЛІНІЙНИХ СИСТЕМ ЗДР, ЗВЕДЕНИХ ДО НОРМАЛЬНОГО ВИГЛЯДУ. С
C КОЖНА З НИХ СКЛАДАЄТЬСЯ З П'ЯТИ ДИФЕРЕНЦІАЛЬНИХ С
C РІВНЯНЬ ПЕРШОГО ПОРЯДКУ І Є ТЕСТОВОЮ ЗАДАЧЕЮ ДЛЯ С
C АПРОБУВАННЯ ПРОГРАМНИХ ЗАСОБІВ ЧИСЕЛЬНОГО С
C РОЗВ'ЯЗУВАННЯ ЗАДАЧІ КОШІ. С
C ЛІТЕРАТУРА: С
C АРУШАНЯН О. Б., ЗАЛЕТКИН С. Ф., ЗАХАРОВ Ф. Ю., С
C КАЛИТКИН Н. Н. О ТЕСТИРОВАНІЙ ПРОГРАММ РЕШЕНИЯ С
C ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ. — М., С
C 1983, ПРЕПРИНТ 139 ИМП ИМ. М. В. КЕЛДЫША АН СССР.
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
DIMENSION Y0(5)
DIMENSION RK(5,4),YARG(5),COF(3)
COMMON/FS2/RM0,Z2,Z3,V1,Z5,Z6,Z7,Z8,Z9,V2,Z11,
* Z12,Z13
COMMON/GEAR9/HUSED,NQUSED,NSTEP,NFE,NJE
COMMON/NOUSTR/NF
DATA COF/2.,2.,1./
PI=3.14159265
TYPE *,'NF='
ACCEPT *,'NF'

```

```

C ----- C
C ВХІДНІ ДАНІ: C
C NUM — НОМЕР СИСТЕМИ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ C
C Н — ПОЧАТКОВИЙ КРОК ІНТЕГРУВАННЯ. C
C MF — ЦІЛЕ ДВОЗНАЧНЕ ЧИСЛО — ВХІДНИЙ ПАРАМЕТР ДЛЯ C
C ППП GEARZ (ДИВ. ОПИС ППП GEARZ). C
C MF = 0 — МЕТОД РУНГЕ — КУТТИ З ФІКСОВАНИМ КРОКОМ Н. C
C MF < 0 — ВИВЕДЕННЯ ТОЧНОГО РОЗВ'ЯЗКУ З КРОКОМ Н. C
C EPS — ПОТРІБНА ЛОКАЛЬНА ТОЧНІСТЬ МЕТОДУ (ДИВ. C
C ОПИС ППП GEARZ). ПРИ MF = 0 І MF < 0 ЗНАЧЕННЯ C
C EPS НЕ ВИКОРИСТОВУЄТЬСЯ. C
C DELTAT — ПАРАМЕТР, ЯКИЙ ВИЗНАЧАЄ ІНТЕРВАЛ ВИДАЧІ C
C РЕЗУЛЬТАТУ: C
C TOUT = DELTAT, DELTAT*S, DELTAT*S**2, C
C ..., 1.0, ДЕ S = SQRT(10). ПРИ C
C MF < 0 ЗНАЧЕННЯ DELTAT НЕ C
C ВИКОРИСТОВУЄТЬСЯ. C
C ПРОГРАМА З ПРИКЛАДОМ РОЗВ'ЯЗУВАННЯ ДИФЕРЕНЦІАЛЬ- C
C НИХ СИСТЕМ ЗА ДОПОМОГОЮ ППП GEARZ. C
C ----- C

```

```

TYPE *,'NUM=      ,MF='
ACCEPT *,NUM,MF
TYPE *,' EPS=      ,DELTAT='
ACCEPT *,EPS,DELTAT
H=1.E-5
WRITE(NF,200)NUM,MF,EPS,DELTAT,H
S=SQRT(10.)
GO TO (10,20,30,40,50,60),NUM

```

```

10 CONTINUE
CCCCCCCCCCCCCCCCCCCC
C НОМЕР 1. C
C ПОГАНО ОБУМОВЛЕ- C
C НА СИСТЕМА C
CCCCCCCCCCCCCCCCCCCC

```

```

RMO=10.
RM1=4.
RM2=5.
V1=20.*PI
V2=100.
C1=0.1
C2=1.
C4=0.5
GO TO 70
20 CONTINUE

```

```

C ----- C
C НОМЕР 2. C
C М'ЯКА СИСТЕМА C
C ----- C

```

```

RMO=-2.
RM1=1.
RM2=-1.
V1=1.
V2=10.
C1=1.
C2=1.5
C4=2.5

```

30 GO TO 70
CONTINUE

C-----C
C НОМЕР 3. C
C ШВИДКО ОСЦИЛЮЮЧА C
C СИСТЕМА C
C-----C

RM0=-2.
RM1=1.
RM2=-1.
V1=1.
V2=1000.
C1=0.5
C2=0.8
C4=2.

GO TO 70
40 CONTINUE

C-----C
C НОМЕР 4. C
C ЖОРСТКА СИСТЕМА 1 C
C-----C

RM0=-100.
RM1=-1.
RM2=-10000.
V1=1.
V1=1.
V1=1.
V2=10.
C1=10.
C2=11.
C4=111.
GO TO 70

50 CONTINUE

C-----C
C НОМЕР 5. C
C ЖОРСТКА СИСТЕМА 2 C
C-----C

RM0=-100.
RM1=-1.
RM2=-10000.
V1=1.
V2=1000.
C1=10.
C2=11.
C4=111.
GO TO 70

60 CONTINUE

C-----C
C НОМЕР 6. C
C ЖОРСТКА ОСЦИЛЮЮЧА C
C СИСТЕМА C
C-----C

RM0=-1.E+5
RM1=1.

RM2=-100.
V1=1.
V2=1000.
C1=100.
C2=101.
C4=201.

C-----G
C ВИЗНАЧЕННЯ ІНШИХ C
C ПАРАМЕТРІВ G
C-----G

70 N=5
T0=0.
Z2=RM0-RM1
Z3=RM1+V1
Z5=Z2-V1
Z6=V1+V1
Z7=RM1-V1
Z8=Z7-RM2
Z9=RM2+V2
Z11=Z8-V2
Z12=V2+V2
Z13=RM2-V2
H0=H
Y0(1)=C1
Y0(2)=C2
Y0(3)=C2
Y0(4)=C4
Y0(5)=C4
C3=C2
C5=C4
INDEX=1
TOUT=DELTA
C ПРИ MF<0 ДРУКУЄТЬСЯ ТОЧНИЙ РОЗВ'ЯЗОК ДЛЯ
C T=H, T=2*H, T=3*H, ..., T=1.0.
C ПРИ MF=0 СИСТЕМА РОЗВ'ЯЗУЄТЬСЯ МЕТОДОМ
C РУНГЕ - КУТТИ З ФІКСОВАНИМ КРОКОМ H.
IPRIZ=3
CF=0.1666666666666666
T=0.
H2=H/2.
NSTEP=0
NFE=0
NJE=0
IF(MF)80,100,150
80 IPRIZ=1
GO TO 160
90 WRITE(NF,201)R1,Y2,Y3,Y4,Y5
T=T+H
IF(T.LT.1.00001)GO TO 160
STOP
100 WRITE(NF,202)
INDEX=0
IPRIZ=2
110 CALL DIFFUN(N,T,Y0,RK)
DO 130 J=1,3
CFK=1./COF(J)*H

```

TARG=T+CFK
DO 120 I=1,5
120   YARG(I)=Y0(I)+CFK*RK(I,J)
130   CALL DIFFUN(N,TARG,YARG,RK(1,J+1))
DO 140 I=1,N
140   Y0(I)=Y0(I)+CF*H*(RK(I,1)+2.*RK(I,2)+2.*
*   RK(I,3)+RK(I,4))
T=T+H
NSTEP=NSTEP+1
NFE=NFE+4
IF(ABS(T-TOUT).GT.H2)GO TO 110
NQUSED=4
GO TO 160
C ЗВЕРНЕННЯ ДО ГОЛОВНОЇ ПІДПРОГРАМИ GEARZ
150 CALL DRIVE(N,T0,H0,Y0,TOUT,EPS,MF,INDEX)
T=TOUT
160 R1=EXP(RM0*T)*C1
R3=EXP(RM1*T)
R4=COS(V1*T)
R5=COS(V2*T)
R6=SIN(V1*T)
R7=SIN(V2*T)
R8=EXP(RM2*T)
Y2=R1+(C2-C1)*R3*R4+(C2-C3)*R3*R6
Y3=R1+(C3-C1)*R3*R4+(2.*C2-C1-C3)*R3*R6
Y4=Y3+(C4-C3)*R8*R5+(C4-C5)*R8*R7
Y5=Y3+(C5-C3)*R8*R5+(2.*C4-C3-C5)*R8*R7
IF(IPRIZ.EQ.1)GO TO 90
PP1=(Y0(1)-R1)**2+(Y0(2)-Y2)**2+(Y0(3)-Y3)**2
PP2=(Y0(4)-Y4)**2+(Y0(5)-Y5)**2
C ОБЧИСЛЕННЯ НОРМИ ГЛОБАЛЬНОЇ ПОХИБКИ РЕЗУЛЬТАТУ В
C МОМЕНТ T=TOUT
D=SQRT(PP1+PP2)
WRITE(NF,203)T,D,NQUSED
IF(INDEX.EQ.0)GO TO 170
WRITE(NF,204)INDEX
GO TO 180
170 TOUT=TOUT*S
IF(TOUT.LE.1.00001)GO TO(110,110,150),IPRIZ
C ДРУКУВАННЯ ЧИСЕЛЬНИХ ЗНАЧЕНЬ РЕЗУЛЬТАТУ Y0(J),
C ПАРАМЕТРІВ NSTEP, NFE I NJE В МОМЕНТ T=TOUT=1.0.
180 WRITE(NF,201)(Y0(J),J=1,N)
WRITE(NF,205)NSTEP,NFE,NJE
C
CALL CLOSE(2)
STOP
200 FORMAT(/1X 'НОМЕР СИСТЕМИ=',I2, 'MF=',I3/
* 1X,'EPS=',E10.2,' DELTAT=',F9.6,
* ' H=',F12.9/)
201 FORMAT(1X,5(F9.6,1X))
202 FORMAT(11X,'МЕТОД РУНГЕ — КУТТИ'/)
203 FORMAT(11X,'T=',E12.4,' D=',E12.4,' NQ=',I3)
204 FORMAT(11X,'АВАРІЙНЕ ПОВЕРНЕННЯ, INDEX=',I3/)
205 FORMAT(11X,'NSTEP=',I8,' NFE=',I7,' NJE=',
* I7//)
END

```



```

SUBROUTINE DIFFUN(N,T,Y,DX)
C ПІДПРОГРАМА DIFFUN ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ
  DIMENSION Y(N),DX(N)
  COMMON/FS2/Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9,
  * Z10,Z11,Z12,Z13
  DX(1)=Z1*Y(1)
  DX(2)=Z2*Y(1)+Z3*Y(2)-Z4*Y(3)
  BUF=Z5*Y(1)+Z6*Y(2)
  DX(3)=BUF+Z7*Y(3)
  DX(4)=BUF+Z8*Y(3)+Z9*Y(4)-Z10*Y(5)
  DX(5)=BUF+Z11*Y(3)+Z12*Y(4)+Z13*Y(5)
  RETURN
  END

```

```

SUBROUTINE PEDERV(N,T,Y,PW,N0)
C ПІДПРОГРАМА PEDERV ОБЧИСЛЕННЯ МАТРИЦІ
C ЯКОБІ ПРАВОЇ ЧАСТИНИ
  DIMENSION Y(N),PW(N0,N0)
  COMMON/FS2/Z1,Z2,Z3,Z4,Z5,Z6,Z7,Z8,Z9,
  * Z10,Z11,Z12,Z13
  DO 10 I=1,N
  DO 10 J=1,N
10    PW(I,J)=0.0
  PW(1,1)=Z1
  PW(2,1)=Z2
  PW(2,2)=Z3
  PW(2,3)=-Z4
  PW(3,1)=Z5
  PW(3,2)=Z6
  PW(3,3)=Z7
  PW(4,1)=Z5
  PW(4,2)=Z6
  PW(4,3)=Z8
  PW(4,4)=Z9
  PW(4,5)=Z10
  PW(5,1)=Z5
  PW(5,2)=Z6
  PW(5,3)=Z11
  PW(5,4)=Z12
  PW(5,5)=Z13
  RETURN
  END

```

Обчислення за цією програмою на комп'ютері ДВК-2М дають такий результат:

НОМЕР СИСТЕМИ = 1 MF= 12
 EPS= 0.10E-05 DELTAT= 0.200000 H= 0.000010000

T= 0.2000E+00 D= 0.9267E-04 NQ= 6
 T= 0.6325E+00 D= 0.7426E-02 NQ= 6
 55.810863 50.710922 60.790127 49.976669 45.232834
 NSTEP= 245 NFE= 509 NJE= 16

НОМЕР СИСТЕМИ = 2 MF= 22
 EPS= 0.10E-04 DELTAT= 0.100000 H= 0.000010000
 T= 0.1000E+00 D= 0.3184E-04 NQ= 4
 T= 0.3162E+00 D= 0.9366E-04 NQ= 5
 T= 0.1000E+01 D= 0.5815E-03 NQ= 5

0.135335 0.869683 2.013361 1.705004 1.505033
NSTEP= 59 NFE= 142 NJE= 6

НОМЕР СИСТЕМИ = 3 MF= 21
EPS= 0.10E-04 DELTAT= 0.100000 H= 0.000010000

KFLAG=-1 - ДЛЯ T= 0.00000000E+00
НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ ABS(H)=HMIN

Н ЗМЕНШИЛОСЬ ДО 0.10000000E-05
І КРОК БУДЕ ПОВТОРЕНИЙ

T= 0.1000E+00 D= 0.4938E-02 NQ= 5
T= 0.3162E+00 D= 0.2391E-01 NQ= 5
T= 0.1000E+01 D= 0.8096E-01 NQ= 5

0.067669 0.508279 1.194474 1.400318 1.738836
NSTEP= 4291 NFE= 4405 NJE= 220

НОМЕР СИСТЕМИ = 4 MF= 22
EPS= 0.10E-03 DELTAT= 0.050000 H= 0.000010000

KFLAG=-1 - ДЛЯ T= 0.00000000E+00
НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ ABS(H)=HMIN

Н ЗМЕНШИЛОСЬ ДО 0.10000000E-05
І КРОК БУДЕ ПОВТОРЕНИЙ

T= 0.5000E-01 D= 0.1396E-02 NQ= 4
T= 0.1581E+00 D= 0.1186E-02 NQ= 4
T= 0.5000E+00 D= 0.4263E-03 NQ= 2

0.000002 0.532325 0.823312 0.823312 0.823312
NSTEP= 95 NFE= 248 NJE= 17

НОМЕР СИСТЕМИ = 5 MF= 31
EPS= 0.10E-04 DELTAT= 0.050000 H= 0.000010000

KFLAG=-1 - ДЛЯ T= 0.00000000E+00
НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ ABS(H)=HMIN

Н ЗМЕНШИЛОСЬ ДО 0.10000000E-05
І КРОК БУДЕ ПОВТОРЕНИЙ

KFLAG=-1 - ДЛЯ T= 0.00000000E+00
НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ ABS(H)=HMIN

Н ЗМЕНШИЛОСЬ ДО 0.10000000E-06
І КРОК БУДЕ ПОВТОРЕНИЙ

T= 0.5000E-01 D= 0.6887E-04 NQ= 6
T= 0.1581E+00 D= 0.9582E-04 NQ= 5
T= 0.5000E+00 D= 0.2910E-04 NQ= 6

-0.000000 0.532281 0.823067 0.823091 0.823083
NSTEP= 342 NFE= 694 NJE= 29

НОМЕР СИСТЕМИ = 6 MF= 33
EPS= 0.10E-04 DELTAT= 0.050000 H= 0.000010000

KFLAG=-3 - ПРИ T= 0.00000000E+00
НЕМАЄ ЗБІЖНОСТІ ПРИ РОЗВ'ЯЗУВАННІ РІВНЯНЬ КОРЕКТОРА

Н ЗМЕНШИЛОСЬ ДО 0.10000000E-05
І КРОК БУДЕ ПОВТОРЕНИЙ

KFLAG=-1 — ДЛЯ Т= 0.00000000E+00
НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ ABS(H)=HMIN

Н ЗМЕНШИЛОСЬ ДО 0.10000000E-06
І КРОК БУДЕ ПОВТОРЕНИЙ

KFLAG=-1 — ДЛЯ Т= 0.00000000E+00
НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ ABS(H)=HMIN

Н ЗМЕНШИЛОСЬ ДО 0.99999999E-08
І КРОК БУДЕ ПОВТОРЕНИЙ

T= 0.5000E-01 D= 0.9745E-02 NQ= 3

T= 0.1581E+00 D= 0.4444E-01 NQ= 2

T= 0.5000E+00 D= 0.3199E-01 NQ= 3

0.000000 1.449932 2.228073 2.253662 2.263056

NSTEP= 3341 NFE= 10002 NJE= 1835

Запитання для самоперевірки

1. Які методи інтегрування необхідно використовувати для розв'язування жорстких систем ЗДР і чому? 2. Які особливості виникають при інтегруванні жорсткої системи ЗДР неявними методами? 3. Що таке матриця Нордсіка й для чого вона використовується? 4. Як проводиться оцінювання локальної похибки методу на кроці інтегрування? 5. Як визначається обернена матриця Якобі? 6. При яких значеннях параметра MF потрібне виконання підпрограми PEDERV?

Контрольні завдання

1. Поява багатьох жорстких систем ЗДР пов'язана з математичним моделюванням складних хімічних реакцій. Наприклад, розкладання етану відбувається за дев'ять елементарних стадій:

- 1) $C_2H_6 \rightarrow CH_3 + CH_3$;
- 2) $C_2H_6 + CH_3 \rightarrow C_2H_5 + CH_4$;
- 3) $C_2H_5 \rightarrow C_2H_4 + H$;
- 4) $C_2H_6 + H \rightarrow C_2H_5 + H_2$;
- 5) $CH_3 + C_2H_5 \rightarrow C_3H_8$;
- 6) $CH_3 + H \rightarrow CH_4$;
- 7) $C_2H_5 + H \rightarrow H_2 + C_2H_4$;
- 8) $C_2H_6 + C_2H_5 \rightarrow C_2H_4 + C_2H_6$;
- 9) $H + H \rightarrow H_2$.

Швидкість кожної з цих реакцій визначається добутком концентрації вихідних речовин на відповідну сталу реакції, тобто

$$r_1 = k_1 [C_2H_6]; \quad r_2 = k_2 [C_2H_6] [CH_3]; \quad r_3 = k_3 [C_2H_5]; \quad r_4 = k_4 [C_2H_6] [H];$$

$$r_5 = k_5 [CH_3] [C_2H_5]; \quad r_6 = k_6 [CH_3] [H]; \quad r_7 = k_7 [C_2H_5] [H];$$

$$r_8 = k_8 [C_2H_6] [C_2H_5]; \quad r_9 = k_9 [H] [H],$$

де [·] — концентрація речовини.

Для складання системи ЗДР, які описують швидкість накопичення та витрачання однієї речовини, відібрати ті з наведених елементарних стадій, в яких вона утворюється чи витрачається. Якщо ця речовина є в лівій частині рівняння реакції (початковий продукт), то швидкість відповідної елементарної стадії r_j входить у кінетичне рівняння для даної речовини зі знаком «мінус» і, навпаки, якщо ця речовина є в правій частині рівняння реакції (кінцевий продукт), то швидкість відповід-

ної елементарної стадії входить у кінетичне рівняння зі знаком «плюс». Стосовно розглядуваної задачі система ЗДР має вигляд

$$\begin{aligned} d [C_2H_6]/dt &= -r_1 - r_2 - r_4 + r_8; \\ d [CH_3]/dt &= -r_2 - r_5 - r_6 + r_1 + r_1; \\ d [CH_4]/dt &= r_2 + r_6; \\ d [CH_4]/dt &= r_3 + r_7 + r_8; \\ d [H]/dt &= -r_4 - r_6 - r_7 - r_9 + r_3; \\ d [H_2]/dt &= r_4 + r_7 + r_6; \\ d [C_3H_8]/dt &= r_5. \end{aligned} \quad (20)$$

Проінтегрувати систему (8), якщо $[C_2H_6]_{t=0} = 1$, а решта речовин мають нульову концентрацію. Сталі реакції мають такі значення: $k_1 = 0,1$; $k_2 = 10$; $k_3 = 10^4$; $k_4 = 10^{-3}$; $k_5 = 1$; $k_6 = 25,3$; $k_7 = 10$; $k_8 = 1$; $k_9 = 15$.

2. Знайти розв'язок жорсткої системи лінійних ЗДР

$$\begin{aligned} y_1' &= a_1 y_1, \quad \vec{y}(0) = (1; 1; 0,5; 0,5; 0,5; 0,5)^T; \\ y_2' &= a_3 y_1 + a_1 y_2, \quad a_1 = -1, \quad a_2 = a, \quad a_3 = a_4 = 1; \\ y_3' &= a_2 y_3, \quad a_5 = 2, \quad a_6 = 3; \\ y_4' &= a_4 y_3 + a_2 y_4; \\ y_5' &= a_5 y_4 + a_2 y_5; \\ y_6' &= a_6 y_5 + a_2 y_6. \end{aligned}$$

Власні числа матриці цієї системи мають такі значення:

$$\lambda_1 = \lambda_2 = -1; \quad \lambda_3 = \lambda_4 = \lambda_5 = \lambda_6 = a_2 = a.$$

Точний розв'язок розглядуваної системи рівнянь має вигляд

$$\begin{aligned} y_1(t) &= y_1(0) e^{\lambda_1 t}; \quad y_2(t) = (y_2(0) + a_3 y_1(0) t) e^{\lambda_2 t}; \\ y_3(t) &= y_3(0) e^{\lambda_3 t}; \quad y_4(t) = (y_4(0) + a_4 y_3(0) t) e^{\lambda_4 t}; \\ y_5(t) &= (y_5(0) + a_5 t (y_4(0) + 0,5 a_4 y_3(0) t)) e^{\lambda_5 t}; \\ y_6(t) &= (y_6(0) + a_6 t (y_5(0) + 0,5 a_5 t (y_4(0) + \frac{a_4}{3} y_3(0) t))) e^{\lambda_6 t}. \end{aligned}$$

Розглянути випадки, коли $a = -10^{-4}$ і $a = -10^{-1}$, користуючись методами Адамса та Гіра. Зробити висновок про ефективність цих методів.

3. З точністю ε на відрізку $[0, T]$ знайти розв'язок задачі Коші для такої системи лінійних ЗДР:

$$\begin{aligned} y_1' &= [(\lambda_1 - c\lambda_2) y_1 + \gamma y_2 + (\beta - \lambda_1 + c\gamma) y_3] (1 - c)^{-1}; \\ y_2' &= [c(\lambda_1 - \lambda_2) y_1 + \beta y_2 + (\gamma - c\lambda_1 + c\beta) y_3] (1 - c)^{-1}; \\ y_3' &= [c(\lambda_1 - \lambda_2) y_1 + \gamma y_2 + (\beta - c\lambda_1 + c\gamma) y_3] (1 - c)^{-1}; \\ \vec{y}(0) &= (2, 1; c - 0,9; c + 1, 1)^T. \end{aligned}$$

Точний розв'язок задачі Коші має вигляд

$$y_1(t) = e^{\lambda_1 t} + 0,1e^{\lambda_2 t} + e^{\lambda_3 t};$$

$$y_2(t) = ce^{\lambda_1 t} + 0,1e^{\lambda_2 t} - e^{\lambda_3 t};$$

$$y_3(t) = ce^{\lambda_1 t} + 0,1e^{\lambda_2 t} + e^{\lambda_3 t}; \quad \gamma = 0,5(\lambda_2 - \lambda_3)(1 - c);$$

$$\lambda_1 = a, \quad \lambda_2 = -30, \quad \lambda_3 = -1; \quad c = 0,5; \quad \beta = 0,5(\lambda_2 + \lambda_3)(1 - c).$$

Провести інтегрування задачі Коші для таких значень її параметрів:

а) $a = -100, \quad T = 100, \quad \varepsilon = 10^{-3};$

б) $a = -10^{-4}, \quad T = 10, \quad \varepsilon = 10^{-3};$

в) $a = -10^{-6}, \quad T = 1, \quad \varepsilon = 10^{-2}.$

Порівняти результати чисельного інтегрування з точним розв'язком задачі.

4. Знайти розв'язок задачі Коші для системи нелінійних ЗДР

$$y_1' = 1 - y_1^2;$$

$$y_2' = 1 - a - (1 + a)y_1^2 + ay_2^2 - 2ay_1(y_2 - y_1);$$

$$\vec{y}(0) = (0, 0), \quad t \in [0, 10].$$

Провести інтегрування задачі Коші для таких значень її параметрів: $a = 1, 10, 100; \varepsilon = 10^{-3}.$

Для розв'язування заданої системи рівнянь застосувати метод Ньютона з різними способами обчислення матриці Якобі. Порівняти результати чисельного інтегрування з точним розв'язком задачі

$$y_1(t) = \operatorname{tg} t; \quad y_2(t) = \operatorname{tg} t + \operatorname{tg}(-at).$$

5. Знайти розв'язок нелінійної задачі Коші для системи ЗДР

$$y_1' = y_2, \quad \vec{y}(0) = (0, 1, 1, 1)^T;$$

$$y_2' = y_1, \quad t \in [0, 10\pi];$$

$$y_3' = y_2 y_3; \quad a = 1, 10, -10, -100;$$

$$y_4' = (a + y_2)y_4.$$

Порівняти знайдені результати при використанні формул Адамса та неявних формул Гіра. Для обчислення матриці Якобі скласти підпрограму PEDERV.

Результати чисельного інтегрування порівняти з точним розв'язком задачі

$$y_1(t) = \sin t; \quad y_2(t) = \cos t; \quad y_3(t) = e^{\sin t}; \quad y_4(t) = e^{at + \sin t}.$$

6. З локальною точністю $\varepsilon = 10^{-4}$ знайти розв'язок сингулярно збуреної системи рівнянь

$$\delta y_1' = -y_1 y_2 + \frac{5}{2} y_2; \quad t \in [0, 100]$$

$$y_2' = 1$$

з початковими умовами $y_1(0) = y_2(0) = 1$. Інтегрування провести для параметра δ в інтервалі $\delta \in [10^{-3}, 1]$. Дослідити залежність мінімального кроку інтегрування від цього параметра.

Знайдений чисельний розв'язок порівняти з точним розв'язком задачі Коші

$$y_1^{(x)} = \left(y_1(0) - \frac{5}{2} \right) e^{\frac{1-(1+x)^2}{26}} + \frac{5}{2};$$

$$y_2(x) = 1 + x.$$

§ 12. Метод стрільби розв'язування крайових задач для звичайних диференціальних рівнянь

Постановка задачі. Розв'язати лінійну крайову задачу для системи ЗДР з розділеними крайовими умовами, тобто

$$y'(x) = A(x)y(x) + f(x), \quad x \in (a, b); \quad (1)$$

$$Py(a) = y_a; \quad (2)$$

$$Qy(b) = y_b, \quad (3)$$

де $y(x) = (y^{(1)}(x), \dots, y^{(n)}(x))$ — невідома вектор-функція; $A(x) = (a_{ij}(x))$, $P = (p_{ij})$, $Q = (q_{ij})$ — задані матриці розмірністю $n \times n$, $k \times n$, $(n-k) \times n$ відповідно, причому ранг матриці P дорівнює k , а ранг матриці Q становить $n-k$; $f(x) = (f^{(1)}(x), \dots, f^{(n)}(x))$ — задана вектор-функція; $y_a = (y_a^{(1)}, \dots, y_a^{(k)})$, $y_b = (y_b^{(1)}, \dots, y_b^{(n-k)})$ — задані вектори.

Методичні вказівки. Виходитимемо з методу стрільби, який належить до класу методів розв'язування крайових задач і ґрунтується на ідеї зведення останніх до послідовності задач Коші [8, с. 287; 2, с. 434].

Нехай $y_{a,i}$, $i = \overline{1, n-k}$ — лінійно незалежні розв'язки однорідної СЛАР

$$Pz = 0; \quad (4)$$

$y_{a,n-k+1}$ — який-небудь окремий розв'язок неоднорідної СЛАР

$$Pz = y_a; \quad (5)$$

$y_i(x)$ — розв'язки задачі Коші для однорідної системи

$$y'_i(x) = A(x)y_i(x), \quad y_i(a) = y_{a,i}, \quad (6)$$

а y_{n-k+1} — розв'язок задачі Коші для неоднорідної системи (1), причому

$$y_{n-k+1}(x) = A(x)y_{n-k+1}(x) + f(x);$$

$$y_{n-k+1}(a) = y_{a,n-k+1}. \quad (7)$$

Тоді, очевидно, лінійна комбінація

$$y(x) = \sum_{j=1}^{n-k} c_j y_j(x) + y_{n-k+1}(x) \quad (8)$$

з довільними сталими c_j задовольняє систему ЗДР (1) і крайову умову (2).

Щоб задовольнити ще крайову умову (3), виберемо ці сталі, виходячи з такої системи $n - k$ лінійних алгебраїчних рівнянь:

$$Qy(b) \equiv Q \left(\sum_{j=1}^{n-k} c_j y_j(b) + y_{n-k+1}(b) \right) = y_b, \quad (9)$$

яка має єдиний розв'язок за умови єдиності розв'язку задачі (1) — (3) [2, 3].

Систему (9) можна переписати у вигляді

$$Rc = t, \quad (10)$$

де $R = QV$, $c = (c_1, \dots, c_{n-k})$, $t = y_b - \omega$, а $V = (y_j(b))_{j=1, n-k}$, $\omega = Qy_{n-k+1}(b)$.

Отже, алгоритм стрільби можна розділити на прямий і зворотний ходи. Перший складається з таких кроків:

1) визначення лінійно незалежних розв'язків однорідної СЛАР (4) й окремого розв'язку неоднорідної СЛАР (5);

2) відшукування яким-небудь чисельним методом (наприклад, методом Рунге — Кутти) розв'язків задачі Коші (6), (7) у точках x_i ($i = \overline{1, N}$), $x_0 = a$, $x_N = b$; $x_i \in (a, b)$, $i = \overline{1, N-1}$;

3) розв'язування СЛАР (10).

Зворотний хід можливий двома варіантами: з відновленням розв'язку за формулою (8) і з повторним розв'язуванням задачі Коші. Перший застосовується тоді, коли на другому кроці прямого ходу розв'язки задачі Коші (6), (7) у точках x_j ($j = \overline{1, N}$), тобто вектори $y_i(x_j)$ ($j = \overline{1, n-k}$) та $y_{n-k+1}(x_j)$ залишалися в пам'яті ЕОМ. Тоді зворотний хід зводиться до обчислення розв'язку задачі (1) — (3) в точках x_j ($j = \overline{1, N}$) за формулою (8). У цьому разі чисельний метод в цілому називатимемо *методом стрільби з відновленням розв'язку*. Він хоч і потребує більшої ємності пам'яті ЕОМ, але реалізується за менший машинний час.

Зворотний хід з повторним розв'язуванням задачі Коші (в цьому випадку весь чисельний метод називатимемо *методом стрільби з повторним розв'язуванням задачі Коші*) використовується тоді, коли розв'язок задачі (6), (7) у проміжних точках x_i ($i = \overline{1, N-1}$) не запам'ятовується і реалізується так:

1) за знайденими довільними сталими c_j ($j = \overline{1, n-k}$) за формулою (8) обчислюється розв'язок задачі (1) — (3) в точці a , тобто

$$y(a) = \sum_{j=1}^{n-k} c_j y_{a,j} + y_{a,n-k+1}; \quad (11)$$

2) розв'язується задача Коші (1), (11).

Метод стрільби з повторним розв'язуванням задачі Коші потребує більше машинного часу, але реалізується з меншою ємністю пам'яті ЕОМ.

▼ Якщо серед розв'язків системи ЗДР (6), (7) є такі компоненти, які змінюються швидко, а також такі, що змінюються повільно (наприклад, матриця A має великий інтервал зміни власних значень), то вектори $y_j(x_i)$ ($j = \overline{1, n-k}$) зі зростанням i стають майже лінійно залежними. В результаті матриці V і R у (10) стають близькими до виродження, через що розв'язок СЛАР (10), а отже, й задачі (1) — (3), визначається з великою обчислювальною похибкою.

Щоб усунути це явище, застосовується варіант методу стрільби, який називається *методом стрільби з ортогоналізацією*. Для пояснення його ідеї позначимо через $\bar{L}(x_i)$ лінійний многовид, який утворюється векторами $y_j(x_i)$, $j = \overline{1, n-k+1}$, $i = \overline{1, N}$. Тоді повний хід стрільби зводиться до поступового переходу від многовиду $L(x_0) \equiv \equiv L(a)$, що задається векторами $y_{a,i}$ ($i = \overline{1, n-k+1}$), до многовиду $L(x_1)$ з базисом $y_i(x_1)$, $i = \overline{1, n-k+1}$ і т. д. При цьому зазначені базиси многовидів вважаються початковими для задач Коші (6), розв'язуючи які й переходимо до наступного многовиду $L(x_{i+1})$.

Можна, проте, довести, що який би інший базис многовиду $L(x_i)$ не взяли як початкові вектори для задачі Коші (6), після розв'язання їх відстанемо деякий базис многовиду $L(x_{i+1})$ [3]. Тому виникає ідея щоб уникнути ситуації, коли вектори $y_j(x_i)$ стають майже лінійно залежними, треба час від часу ортогоналізувати базиси в многовидах $L(x_i)$: при переході до многовиду $L(x_{i+1})$ за допомогою (6) за початкове значення слід брати ортогоналізований базис $\tilde{y}_j(x_i)$, $j = \overline{1, n-k+1}$.

Позначимо точки, в яких проводиться ортогоналізація цього базису, через x_s , $s = \overline{1, M}$, $x_1 = a$, $x_M = b$. Тоді на кожному відрізку $[x_{i-1}, x_i]$, $i = \overline{1, M}$ замість (8) матимемо такий розв'язок задачі (1) — (3):

$$y(x) = \sum_{j=1}^{n-k} c_j^{(i)} y_j(x) + y_{n-k+1}(x), \quad x \in [x_{i-1}, x_i],$$

де $y_j(x)$, $j = \overline{1, n-k}$ — розв'язки задач Коші (6) на відрізку $[x_{i-1}, x_i]$ за початкових умов $\tilde{y}_j(x_{i-1})$, $j = \overline{1, n-k}$, а $y_{n-k+1}(x)$ — розв'язок (7) за початкової умови $\tilde{y}_{n-k+1}(x_{i-1})$.

Між векторами $c^{(s)} = (c_1^{(s)}, \dots, c_{n-k}^{(s)}, 1)$ та матрицями $\tilde{Y}(x_s) = (\tilde{y}_j(x_s))$, $j = \overline{1, n-k+1}$ можна встановити зручні для обчислення рекурентні співвідношення.

У цілому алгоритм реалізації методу стрільби з ортогоналізацією зводиться до виконання таких дій:

1. Розбиваємо відрізок інтегрування $[a, b]$ на частини точками ортогоналізації x_s ($s = \overline{1, M}$), $x_1 = a$, $x_M = b$.

2. Розв'язуючи СЛАР (4) та (5), знаходимо систему векторів $z(x_1) = (y_{a,j})$, $j = \overline{1, n-k+1}$.

3. Нехай розв'язками задач Коші (6), (7) у точці $x_s \in z(x_s) = (y_i(x_s))$, $j = \overline{1, n-k+1}$, причому в точці x_1 це відома з кроку 2 система векторів. У кожній точці x_s ($s = \overline{1, M}$) ортонормуємо систему векторів $z(x_s)$, причому останній вектор-стовпець не нормується. Знайдену таким чином систему векторів $Y(x_s) = (\tilde{y}_j(x_s))$, $j = \overline{1, n-k}$, використовуємо як початкові вектори для задач Коші (6), (7) на відрізку $[x_s, x_{s+1}]$.

Формули для визначення векторів $Y(x_s)$ мають вигляд (це формули відомого процесу ортогоналізації Шмідта)

$$\tilde{y}_j(x_s) = \omega_{jj}^{-1} \left(y_j(x_s) - \sum_{i=1}^{j-1} \omega_{ij} \tilde{y}_i(x_s) \right),$$

де ω_{ij} — елементи трикутної матриці $\Omega^{(s)}$, які визначаються так:

$$\omega_{ij} = \begin{cases} (\tilde{y}_i(x_s), y_j(x_s)), & i < j; \\ \left(\|y_j(x_s)\|^2 - \sum_{i=1}^{j-1} \omega_{ij}^2 \right)^{1/2}, & i = j, \quad j \neq n-k+1; \\ 1, & i = j = n-k+1; \\ 0, & i > j. \end{cases}$$

4. В точці $x_M = b$ для визначення векторів $c^{(M)} = (c_1^{(M)}, \dots, c_{n-k}^{(M)}, 1)$ розв'язуємо СЛАР вигляду (10), де $V = (\tilde{y}_j(x_M))$, $j = \overline{1, n-k}$, а $W = Q y_{n-k+1}(x_M)$.

5. Розв'язок задачі (1) — (3) в точках x_s , $s = M, M-1, \dots, 1$ послідовно визначаємо за формулою

$$y(x_s) = \tilde{Y}(x_s) c^{(s)},$$

де $c^{(s)}$ обчислюється за відомим вектором $c^{(s')}$ при $s < s' \leq M$ так:

$$\Delta c^{(s)} = c^{(s')}.$$

Тут Δ — матриця трикутного вигляду, що визначається виразом

$$\Delta = \Omega^{(s')} \Omega^{(s'-1)} \dots \Omega^{(s+1)}.$$

Зробимо ще такі зауваження: 1) з метою економії машинного часу та ємності пам'яті ЕОМ бажано як початкову точку інтегрування задач Коші (6) вибирати ту з точок a чи b , в якій задано більшу кількість граничних умов (тоді кількість цих задач буде меншою); 2) з метою

економії ємності пам'яті ЕОМ серед точок ортогоналізації доцільно виділити точки x_{sp} , де виводяться результати обчислень, тобто матриці $\tilde{Y}(x_s)$ зберігаються в точках $x_{sp} \in (x_1, x_M)$, а замість матриць $\Omega^{(s)}$ у точках x_{sp} будуть матриці трикутного вигляду; 3) кількість точок ортогоналізації в областях, де розв'язок задачі (1) — (3) може мати особливості, слід збільшити.

Приклади використання підпрограм. **Приклад 1.** Знайдемо розв'язок такої крайової задачі:

$$\frac{d^4 u}{dx^4} - a^4 u = -f(x), \quad x \in (0, l);$$

$$u(0) = 0, \quad u'(0) = -bl, \quad u(l) = 0, \quad u'(l) = bl, \quad (12)$$

де a, b, l — задані числа; $f(x) = a^4 bx(x-l)$.

Неважко перевірити, що її розв'язком є функція

$$u(x) = bx(x-l).$$

Складемо програму розв'язування цієї задачі методом стрільби. Для цього подамо її у вигляді системи ЗДР першого порядку, а саме:

$$\frac{dY}{dx} = AY + F(x), \quad x \in (0, l);$$

$$PY(0) = y_a, \quad QY(l) = y_b, \quad (12a)$$

де

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a^4 & 0 & 0 & 0 \end{pmatrix}; \quad P = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}; \quad Q = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix};$$

$$Y \equiv Y(x) = (y_1(x), y_2(x), y_3(x), y_4(x)); \quad y_1(x) = u(x);$$

$$y_2(x) = u'(x); \quad y_3(x) = u''(x); \quad y_4(x) = u'''(x);$$

$$F(x) = (0, 0, 0, f(x)); \quad y_a = (0, -bl); \quad y_b = (0, bl).$$

За цим методом спочатку розв'язуємо однорідну систему рівнянь

$$PX = 0,$$

де $x = (x_1, x_2, x_3, x_4)$. Вона має такі два лінійно незалежних розв'язки:

$$Y^{(1)} = (0, 0, 1, 0); \quad Y^{(2)} = (0, 0, 0, 1).$$

Окремим розв'язком неоднорідної системи рівнянь $PX = y_a$ є

$$Y^{(3)} = (0, -bl, 0, 0).$$

Далі виконуємо такі кроки методу стрільби:

1) розв'язуємо задачі Коші

$$Y_1'(x) = AY_1(x), \quad x \in (0, l]; \quad Y_1(0) = Y^{(1)} = (0, 0, 1, 0);$$

$$Y_2'(x) = AY_2(x), \quad x \in (0, l]; \quad Y_2(0) = Y^{(2)} = (0, 0, 0, 1); \quad (13)$$

$$Y_3'(x) = AY_3(x) + F(x), \quad x \in (0, l]; \quad Y_3(0) = Y^{(3)} = (0, -bl, 0, 0),$$

де

$$Y_i(x) = (y_{1i}(x), y_{2i}(x), y_{3i}(x), y_{4i}(x)), \quad t = \overline{1, 3};$$

2) розв'язуємо систему двох лінійних алгебраїчних рівнянь відносно невідомих c_1 і c_2 , тобто систему рівнянь

$$c_1 QY_1(t) + c_2 QY_2(t) + QY_3(t) = y_b, \quad (14)$$

або

$$\begin{aligned} y_{11}(t) c_1 + y_{12}(t) c_2 + y_{13}(t) &= 0; \\ y_{21}(t) c_1 + y_{22}(t) c_2 + y_{23}(t) &= bl. \end{aligned}$$

За формулами Крамера маємо

$$\begin{aligned} \Delta &= y_{11}(t) y_{22}(t) - y_{21}(t) y_{12}(t); \\ \Delta_1 &= -y_{13}(t) y_{22}(t) - y_{12}(t) (bl - y_{23}(t)); \\ \Delta_2 &= y_{11}(t) (bl - y_{23}(t)) + y_{21}(t) y_{13}(t). \end{aligned} \quad (15)$$

Тоді

$$c_1 = \Delta_1 / \Delta, \quad c_2 = \Delta_2 / \Delta;$$

3) знаходимо розв'язок $y(0)$ за формулою

$$y(0) = c_1 Y_1(0) + c_2 Y_2(0) + Y_3(0),$$

тобто

$$y(0) = 0, \quad y_2(0) = -bl, \quad y_3(0) = c_1, \quad y_4(0) = c_2;$$

4) розв'язуємо задачу Коші

$$\begin{aligned} \frac{dy(x)}{dx} &= Ay(x) + F(x), \quad x \in (0, l]; \\ y(0) &= (0, -bl, c_1, c_2), \end{aligned} \quad (16)$$

для чого користуємося такою програмою:

```
PROGRAM TST1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА МЕТОДУ СТРІЛЬБИ ДЛЯ ПРИКЛАДУ 1 C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL F
REAL*4 Y(4),WORK(27),A,B,T,ABSERR,RELERR,
* GRAF(93),C,C1,C2,TPRINT,TOUT
INTEGER IWORK(5),INDF,IFLAG,N,N1
LOGICAL*1 IND
COMMON /INDF/ INDF /PARAM/ A,B,AL
C
10 TYPE 101
ACCEPT *,A
TYPE 102
ACCEPT *,B
TYPE 103
ACCEPT *,AL
TYPE 104
ACCEPT *,ABSERR
TYPE 105
ACCEPT *,RELERR
```

```

TYPE 106
ACCEPT *,NW
N=31
N1=N-1
TPRINT=AL/FLOAT(N1)
BL=B*AL
DO 20 I=1,N
    GRAF(I)=(I-1)*TPRINT
20  INDF=0
    Y(1)=0.
    Y(2)=0.
    Y(3)=1.
    Y(4)=0.
    T=0.
    IFLAG=1
    CALL RKF45 ( F, 4, Y, T, AL, RELERR, ABSERR,
                IFLAG, WORK, IWORK )
    IF ( IFLAG.NE.2 ) GO TO 50
    A11=Y(1)
    A21=Y(2)
    Y(1)=0.
    Y(2)=0.
    Y(3)=0.
    Y(4)=1.
    T=0.
    IFLAG=1
    CALL RKF45(F,4,Y,T,AL,RELERR,ABSERR,
                IFLAG,WORK,IWORK)
    * IF(IFLAG.NE.2)GO TO 50
    A12=Y(1)
    A22=Y(2)
    INDF=1
    Y(1)=0.
    Y(2)=-BL
    Y(3)=0.
    Y(4)=0.
    T=0.
    IFLAG=1
    CALL RKF45(F,4,Y,T,AL,RELERR,ABSERR,
                IFLAG,WORK,IWORK)
    * IF(IFLAG.NE.2)GO TO 50
    A13=Y(1)
    A23=Y(2)
    C=A11*A22-A21*A12
    C1=(-A13*A22-A12*(BL-A23))/C
    C2=(A11*(BL-A23)+A21*A13)/C
    Y(1)=0.
    Y(2)=-BL
    Y(3)=C1
    Y(4)=C2
    T=0.
    TOUT=T
    IFLAG=1
    I=N+1
30  CALL RKF45(F,4,Y,T,TOUT,RELERR,ABSERR,
    *      IFLAG,WORK,IWORK)

```

```

IF(IFLAG.NE.2)GO TO 50
GRAF(1)=Y(1)
I=I+1
TOUT=GRAF(I-N)
IF(T.LT.AL)GO TO 30
C CALL GRAPH(NW,1,N,2,2*N,GRAF)
GRAF(2*N+1)=0.
GRAF(3*N)=0.
DO 40 J=2,N1
    T=GRAF(J)
    I=2*N+J
40    GRAF(I)=B*T*(T-AL)
CALL GRAPH(NW,1,N,3,3*N,GRAF)
TYPE 107
ACCEPT 108,IND
IF(IND.EQ.'Y')GO TO 10
STOP
50 TYPE 109,IFLAG
101 FORMAT(1H$, 'ВВЕДИТЬ ЗНАЧЕННЯ А:')
102 FORMAT(1H$, 'ВВЕДИТЬ ЗНАЧЕННЯ В:')
103 FORMAT(1H$, 'ВВЕДИТЬ ЗНАЧЕННЯ L:')
104 FORMAT('ВВЕДИТЬ ЗНАЧЕННЯ АВС. ПОХИБКИ/'
* 'ІНТЕГРУВАННЯ ЗАДАЧІ КОШІ')
105 FORMAT('ВВЕДИТЬ ЗНАЧЕННЯ ВІДН. ПОХИБКИ/'
* 'ІНТЕГРУВАННЯ ЗАДАЧІ КОШІ')
106 FORMAT(1H$, 'NW=')
107 FORMAT(1H$, 'ПРОДОВЖИТИ?')
108 FORMAT(A1)
109 FORMAT('ПЕРЕРИВАННЯ ПРИ ІНТЕГРУВАННІ',
* 'ЗАДАЧІ КОШІ')
* STOP
END

SUBROUTINE F(X,Y,YP)
REAL*4 X,Y(1),YP(1)
INTEGER INDF
COMMON/INDF/INDF/PARM/A,B,AL
YP(1)=Y(2)
YP(2)=Y(3)
YP(3)=Y(4)
YP(4)=(A**4)*Y(1)+
* INDF*(-B*X*(X-AL)*(A**4))
* RETURN
END

```

Графік наближеного розв'язку задачі Коші (1), знайденого за допомогою методу стрільби, який реалізує програма TST1 при $a = 10$, $b = 1$, $l = 1$, а також графік точного розв'язку задачі Коші (2) при тих самих значеннях цих параметрів зображено на с. 133.

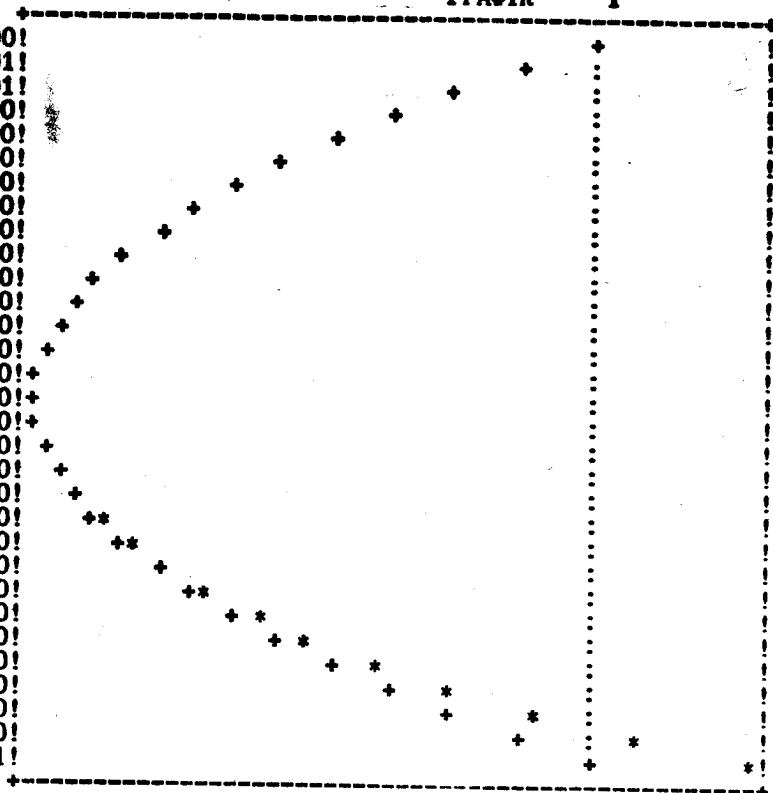
Як бачимо, наближений розв'язок навіть якісно відрізняється від точного: в той час як точний розв'язок завжди від'ємний, наближений при деяких значеннях аргументу набуває додатних значень. Ця різниця стає ще більш контрастною при $a = 13$, $b = 1$, $l = 1$, що ілюструє графік, показаний на с. 134.

У цьому прикладі спостерігається явище, про яке йшлося вище. Дійсно, характеристичне рівняння $\lambda^4 - a^4 = 0$ для однорідного рівняння (12) має корені $\lambda_{1,2} =$

ФУНКЦІЇ: 1 - * 2 - +
 АБСЦИСИ

ГРАФІК 1

0.0000E+00!
 0.3333E-01!
 0.6667E-01!
 0.1000E+00!
 0.1333E+00!
 0.1667E+00!
 0.2000E+00!
 0.2333E+00!
 0.2667E+00!
 0.3000E+00!
 0.3333E+00!
 0.3667E+00!
 0.4000E+00!
 0.4333E+00!
 0.4667E+00!
 0.5000E+00!
 0.5333E+00!
 0.5667E+00!
 0.6000E+00!
 0.6333E+00!
 0.6667E+00!
 0.7000E+00!
 0.7333E+00!
 0.7667E+00!
 0.8000E+00!
 0.8333E+00!
 0.8667E+00!
 0.9000E+00!
 0.9333E+00!
 0.9667E+00!
 0.1000E+01!



ШКАЛА ОРДИНАТ
 1 2 3 4 5 6 7 8 9 10 11
 STEP= 0.31945E-01
 VMIN=-0.25000E+00 VMAX= 0.69448E-01

$= \pm a, \lambda_{3,4} = \pm ia, i = \sqrt{-1}$, а загальним розв'язком однорідного рівняння (12) є

$$u_0(x) = M_1 \operatorname{sh} ax + M_2 \operatorname{ch} ax + M_3 \cos ax + M_4 \sin ax, \quad (17)$$

де $M_j, j = \overline{1,4}$ — довільні сталі. Присутні тут компоненти $\operatorname{sh} ax, \operatorname{ch} ax$ змінюються швидко (тим швидше, чим більше значення має параметр a), а компоненти $\cos ax, \sin ax$ — повільно. Це свідчить про жорсткість задач Коші (13). Пояснимо, що це означає, не вдаючись до складних математичних викладок.

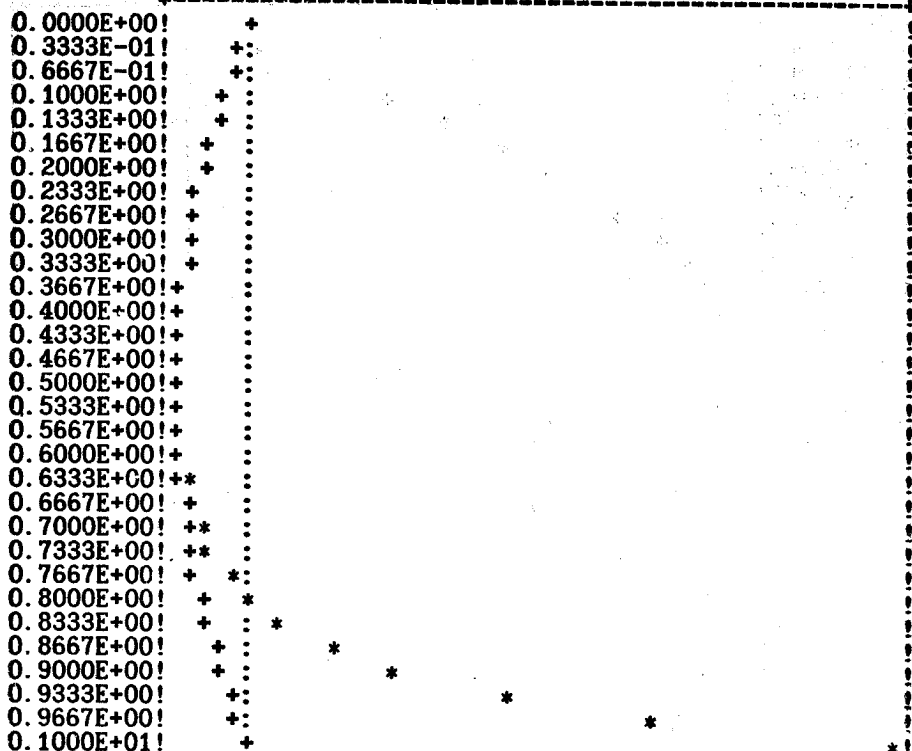
Які б не були початкові умови, внаслідок похибок заокруглення розв'язки задач Коші (13) міститимуть швидкозмінні компоненти $\operatorname{sh} ax, \operatorname{ch} ax$ з ненульовими коефіцієнтами $M_1 \neq 0, M_2 \neq 0$. При великих значеннях параметра a саме вони і вноситимуть основний вклад у ці розв'язки. Це може призвести до того, що вектори $Y_1(t), Y_2(t)$ будуть майже паралельними (тобто майже однаковими), внаслідок чого

ФУНКЦІІ: 1 - * 2 - +

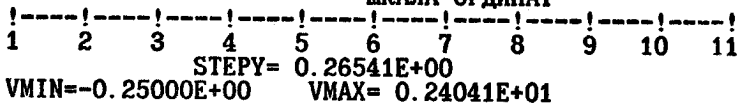
АБСЦИСИ

ГРАФІК

1



ШКАЛА ОРДИНАТ



СЛАР (14) матиме близький до нуля визначник або буде погано обумовленою, що при «недбалому» її чисельному розв'язуванні з застосуванням формул Крамера (15) може спричинити великі похибки розв'язку (порівняйте з прикладом 2).

Ці похибки при визначенні сталих c_1 , c_2 призводять до похибок розв'язку задачі Коші (16), який і є наближеним розв'язком крайової задачі. Крім того, наявність швидкозмінних розв'язків задачі Коші (13) може призвести до того, що при користуванні підпрограмою RKF45 для досягнення заданої точності розв'язку вибиратиметься досить малий крок інтегрування. Це спричинить зростання кількості кроків на відріжку $[0, l]$, а отже, й збільшення впливу похибки заокруглення (такий самий ефект може дати також зростання l).

Приклад 2. Зазначені недоліки певною мірою притаманні підпрограмі SHTNG, призначеній для розв'язування лінійної крайової задачі з системою ЗДР при засто-

суванні різних варіантів методу стрільби. Зміст її параметрів розкрито в тексті підпрограми.

Для реалізації підпрограми SHTNG потрібно в головній програмі задати робочі файли прямого доступу:

3 — з довжиною запису $2 * N * (N - K + 1)$ і кількістю записів 1, якщо CASE=0 (в протилежному разі — на одиницю меншою від числа точок виведення);

4 — (лише при CASE=2) з довжиною запису $(N - K + 1) * (N - K + 2)$ слів і кількістю записів на дві одиниці меншою від числа точок виведення.

На алгоритмічній мові ФОРТРАН треба задати

```
DIFINE FINE 3 (M1, N1, U, IF1, 4 (M2, N2, U, IF2).
```

де M1, M2 — цілі сталі, що визначають кількість записів у файлі; N1, N2 — цілі сталі, які задають довжину (кількість слів, 1 слово = 2 байта) кожного запису; U — буква, яка вказує, що файл є неформативним; IF1, IF2 — зв'язні змінні файла.

Зауважимо, що структура запису з прямим доступом від розміру фізичного блоку пристрою введення-виведення не залежить. Однак більш ефективні результати досягаються, якщо розмір запису є дільником чи кратним 256 словам.

Змінні зв'язку для файлів 3, 4 задаються оператором загальних областей з іменем FAIL, тобто

```
COMMON |FAIL| IF1, IF2.
```

Для реалізації підпрограми SHTNG потрібні такі функції та підпрограми, як DECOMP, SOLVE, RKF45, RSTM, PRPM, ORT, PRTM. Крім того, необхідно задати підпрограму-функцію F з метою обчислення вектора похідних для системи рівнянь (1), (6), (7). При цьому вектор похідних визначається як для однорідної, так і для неоднорідної систем рівнянь, якщо змінна INDE, яка передається в підпрограму F через спільну область пам'яті EOM, приймає відповідно значення 0 і 1.

Матриці та праві частини крайових умов (2), (3) зберігаються по стовпцях робочого файла FTN9.DAT у вигляді, який забезпечує читання їх за такими форматами:

Матриця лівої крайової умови

```
* FORMAT 5F6.3 *  
1., 0., 0., 1., 0.  
0., 0., 0., 0., 1.
```

Матриця правої крайової умови

```
* FORMAT 5F6.3 *  
1., 0., 0., 1., 0.  
0., 0., 0., 0., 1.
```

Цей файл створюється за допомогою редактора текстів (наприклад, TED, WED тощо).

Перед пуском програми, що викликає підпрограму SHTNG, необхідно присвоїти логічне ім'я дев'ятого пристрою, в якому зберігається файл FTN9.DAT з матрицями крайових умов. Наприклад, якщо цей файл знаходиться в пристрої з фізичним ім'ям DX, то можна використати таку команду монітора операційної системи РАФОС(RT-11):

```
·ASS DX:9
```

Зауважимо, що з цією метою можна скористатися також відповідними можливостями алгоритмічної мови ФОРТРАН.

Результати розрахунків за програмою виводяться по другому каналу і записуються у файл FTN2.DAT на робочий диск,

Зазначимо, що перед пуском програми необхідно зарезервувати вільний простір на робочому диску для створення файла FTN2.DAT з результатами розрахунків і файлів FTN3.DAT та FTN4.DAT прямого доступу. Застосування підпрограми SHTNG проілюструємо на прикладі розв'язування крайової задачі (12). Головна програма та підпрограма, що описує праву частину (12а), мають вигляд:

```
С ГОЛОВНА ПРОГРАМА ДО ПІДПРОГРАМИ SHTNG С
```

```
LOGICAL*1 TVIV(101)
REAL*4 TORT(101),A1(10),OM(6),
* Z(12),Z1(27),Z2(4)
INTEGER IPVT(5),CASE
EXTERNAL F
COMMON/FAIL/IF1,IF2
* /PARM/A,B,AL
DEFINE FILE 3(10,24,U,IF1), 4(9,12,U,IF2)
```

```
G
N=4
K=2
TYPE 101
ACCEPT *,KT
TYPE 102
ACCEPT *,A
TYPE 103
ACCEPT *,B
TYPE 104
ACCEPT *,AL
KTORT=KT*10+1
H=AL/FLOAT(KTORT-1)
DO 10 I=1,KTORT
10 TORT(I)=(I-1)*H
DO 20 I=1,KTORT
20 TVIV(I)=.TRUE.
DO 30 I=1,11
30 TVIV(1+(I-1)*KT)=.FALSE.
EPS=0.1E-05
AEPS=0.1E-05
TYPE 105
ACCEPT *,CASE
CASE=0
```

```
C
WRITE(2,106)N,K,EPS,AEPS,CASE
IF(CASE.GT.1)GO TO 40
WRITE(2,108)
WRITE(2,109)(TORT(I),I=1,KTORT)
GO TO 50
40 WRITE(2,107)KTORT
WRITE(2,109)(TORT(I),I=1,KTORT)
WRITE(2,108)
WRITE(2,110)(TVIV(I),I=1,KTORT)
```

```
C
50 CALL SHTNG(CASE,F,N,K,KTORT,EPS,AEPS,
* TVIV,IPVT,TORT,A1,OM,
* Z,Z1,Z2)
IF(CASE.NE.2)GO TO 70
```

```

WRITE(2,111)
DO 60 I=1,11
    X=TORT(1+(I-1)*KT)
    Z(I)=B*X*(X-AL)
    WRITE(2,112)X,Z(I)
60 CONTINUE
70 STOP
101 FORMAT(1H$, 'КІЛЬКІСТЬ ТОЧОК ОРТОГОНАЛІЗАЦІЇ',
*          'N*10=' )
102 * FORMAT(1H$, 'A=' )
103 * FORMAT(1H$, 'B=' )
104 * FORMAT(1H$, 'L=' )
105 * FORMAT(1H$, 'ЗАДАЙТЕ РЕЖИМ РОБОТИ *CASE*:')
106 * FORMAT(/5X, 'ЧИСЛО РІВНЯНЬ СИСТЕМИ:', I3/
*          5X, 'КІЛЬКІСТЬ КРАЙОВИХ УМОВ ПРИ X=A:', I3/
*          5X, 'ТОЧНІСТЬ ІНТЕГРУВАННЯ ЗА P.—K.=', E12.4/
*          5X, 'ABS ТОЧНІСТЬ ІНТЕГРУВАННЯ =', E12.4/
*          5X, 'РЕЖИМ РОБОТИ *CASE*:', I3)
107 * FORMAT(5X, 'КІЛЬКІСТЬ ТОЧОК ОРТОГОНАЛІЗАЦІЇ=',
*          I3/7X, 'ТОЧКИ ОРТОГОНАЛІЗАЦІЇ')
108 * FORMAT(10X, 'ТОЧКИ ВИВЕДЕННЯ')
109 * FORMAT(1X, 4F12.5)
110 * FORMAT(L8, 4L10)
111 * FORMAT(/"ТОЧНИЙ РОЗВ'ЯЗОК"/)
112 * FORMAT('РОЗВ'ЯЗОК У ТОЧЦІ X=',
*          E12.5/(4E12.5))

```

END
SUBROUTINE F(X, Y, YP)

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ПІДПРОГРАМА ОБЧИСЛЕННЯ ВЕКТОРА ПОХІДНИХ ДЛЯ                                C
C СИСТЕМ ЗДР                                                                C
C ВХІДНІ ПАРАМЕТРИ:                                                        C
C _____                                                                  C
C X — ЗНАЧЕННЯ АРГУМЕНТУ                                                  C
C Y — МАСИВ ЗНАЧЕНЬ ФУНКЦІЇ В ТОЧЦІ X                                    C
C ВИХІДНІ ПАРАМЕТРИ:                                                      C
C _____                                                                  C
C YP — МАСИВ ЗНАЧЕНЬ ПОХІДНИХ У ТОЧЦІ X                                  C
C ЗАУВАЖЕННЯ: ЗМІННА INDF ПЕРЕДАЄТЬСЯ ЧЕРЕЗ СПІЛЬНУ                     C
C ПАМ'ЯТЬ З ПІДПРОГРАМИ SHTNG І НАБУВАЄ                                  C
C ЗНАЧЕНЬ 0 ТА 1. ПРИ ЦЬОМУ ВЕКТОР ПОХІД-                               C
C НИХ ОБЧИСЛЮЄТЬСЯ ВІДПОВІДНО ДЛЯ                                       C
C ОДНОРІДНОЇ ТА НЕОДНОРІДНОЇ СИСТЕМ ЗДР                                C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

```

REAL*4X, Y(1), YP(1)
INTEGER INDF
COMMON/INDF/INDF
* /PARM/A, B, AL
YP(1)=Y(2)
YP(2)=Y(3)
YP(3)=Y(4)
YP(4)=(A**4)*Y(1)+
* INDF*(-B*X*(X-AL)*(A**4))
RETURN
END

```

Обчислення на комп'ютері ДВК-2М у режимі CASE = 0 при $a = 13$, $b = 1$,
 $j = 1$ дають такий результат:

ЧИСЛО РІВНЯНЬ СИСТЕМИ: 4
КІЛЬКІСТЬ КРАЙОВИХ УМОВ ПРИ $X=A$: 2
ТОЧНІСТЬ ІНТЕГРУВАННЯ ЗА $P.-K.=0.1000E-05$
ABS ТОЧНІСТЬ ІНТЕГРУВАННЯ $=0.1000E-05$
РЕЖИМ РОБОТИ *CASE*: 0
ТОЧКИ ВИВЕДЕННЯ
0.00000 0.10000 0.20000 0.30000
0.40000 0.50000 0.60000 0.70000
0.80000 0.90000 1.00000

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.00000E+00$
0.000000 -1.00000 2.00000 0.000000
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.10000E+00$
-0.900000E-01 -0.800000 2.00000 -0.955963E-05
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.20000E+00$
-0.160000 -0.600000 2.00000 -0.407728E-04
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.30000E+00$
-0.210000 -0.400001 1.99999 -0.102543E-03
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.40000E+00$
-0.240000 -0.200003 1.99997 -0.423982E-03
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.50000E+00$
-0.250001 -0.921637E-05 1.99988 -0.158592E-02
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.60000E+00$
-0.240003 0.199966 1.99955 -0.588737E-02
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.70000E+00$
-0.210010 0.399873 1.99835 -0.214405E-01
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.80000E+00$
-0.160036 0.599534 1.99395 -0.786950E-01
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.90000E+00$
-0.901315E-01 0.798291 1.97778 -0.288945
ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.10000E+01$
-0.482699E-03 0.993727 1.91844 -1.06046
ТОЧНИЙ РОЗВ'ЯЗОК

РОЗВ'ЯЗОК У ТОЧЦІ $X=0.00000E+00$
0.00000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.10000E+00$
0.90000E-01
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.20000E+00$
0.16000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.30000E+00$
0.21000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.40000E+00$
0.24000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.50000E+00$
0.25000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.60000E+00$
0.24000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.70000E+00$
0.21000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.80000E+00$
0.16000E+00
РОЗВ'ЯЗОК У ТОЧЦІ $X=0.90000E+00$
0.90000E-01

РОЗВ'ЯЗОК У ТОЧЦІ $X=0.10000E+01$
 $0.00000E+00$

Як бачимо, підпрограма SHTNG дає кращий результат, ніж підпрограма TST1, що пояснюється використанням у ній для розв'язування системи рівнянь (14) підпрограм DECOMP та SOLVE, які реалізують метод Гауса з вибором головного елемента в стовпцю. При застосуванні ж для розв'язування розглядуваної задачі підпрограми SHTNG у режимі CASE-2 (метод стрільби з ортогоналізацією) знайдений розв'язок ще ближче наближається до точного.

Але при $a = 20$, $b = i$, $l = i$ підпрограма SHTNG в режимі CASE = 0 (метод стрільби без ортогоналізації) дає результат, який відрізняється від точного в околі точки $x = i$:

ЧИСЛО РІВНЯНЬ СИСТЕМИ: 4

КІЛЬКІСТЬ КРАЙОВИХ УМОВ ПРИ $X=A$: 2

ТОЧНІСТЬ ІНТЕГРУВАННЯ ЗА Р.—К.= $0.1000E-05$

ABS ТОЧНІСТЬ ІНТЕГРУВАННЯ = $0.1000E-05$

РЕЖИМ РОБОТИ *CASE*: 0

ТОЧКИ ВИВЕДЕННЯ

0.00000	0.10000	0.20000	0.30000
0.40000	0.50000	0.60000	0.70000
0.80000	0.90000	1.00000	

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.00000E+00$

0.000000 -1.00000 2.00000 0.000000

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.10000E+00$

-0.900000E-01 -0.800000 2.00000 -0.809067E-04

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.20000E+00$

-0.160000 -0.600002 1.99996 -0.815450E-03

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.30000E+00$

-0.210001 -0.400014 1.99972 -0.545913E-02

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.40000E+00$

-0.240005 -0.200101 1.99798 -0.402778E-01

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.50000E+00$

-0.250037 -0.745926E-03 1.98508 -0.298584

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.60000E+00$

-0.240276 0.194486 1.88971 -2.20573

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.70000E+00$

-0.212037 0.359256 1.18512 -16.2973

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.80000E+00$

-0.175053 0.298943 -4.02113 -120.423

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.90000E+00$

-0.201226 -1.42453 -42.4906 -889.812

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.10000E+01$

-0.821857 -15.4371 -326.743 -6574.85

Ця сама підпрограма в режимі CASE = 2 (з використанням ортогоналізації) при $a = 20$, $b = i$, $l = i$ дає цілком задовільний результат:

ЧИСЛО РІВНЯНЬ СИСТЕМИ: 4

КІЛЬКІСТЬ КРАЙОВИХ УМОВ ПРИ $X=A$: 2

ТОЧНІСТЬ ІНТЕГРУВАННЯ ЗА Р.—К.= $0.1000E-05$

ABS ТОЧНІСТЬ ІНТЕГРУВАННЯ = $0.1000E-05$

РЕЖИМ РОБОТИ *CASE*: 2

КІЛЬКІСТЬ ТОЧОК ОРТОГОНАЛІЗАЦІЇ=11

ТОЧКИ ОРТОГОНАЛІЗАЦІЇ

0.00000	0.10000	0.20000	0.30000
0.40000	0.50000	0.60000	0.70000
0.80000	0.90000	1.00000	

ТОЧКИ ВИВЕДЕННЯ

F F F F F
F F F F F
F

ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.10000E+01$
 0.931323—09 1.00000 2.00004 0.933275E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.90000E+00$
 —0.899999E—01 0.800000 1.99997 —0.721519E—05
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.80000E+00$
 —0.160000 0.600001 2.0001 —0.543961E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.70000E+00$
 —0.210000 0.399999 2.00002 0.443291E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.60000E+00$
 —0.240000 0.200000 1.99997 0.194048E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.50000E+00$
 —0.250000 0.191104E—05 2.00001 —0.651682E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.40000E+00$
 —0.240000 —0.200001 2.00003 0.282809E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.30000E+00$
 —0.210000 —0.400001 1.99997 0.380225E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.20000E+00$
 —0.160000 —0.599998 1.99999 —0.661892E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.10000E+00$
 —0.900001E—01 —0.800000 2.00003 0.196896E—03
 ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ $X=0.00000E+00$
 0.000000 —1.00000 1.99995 0.106095E—02

Зуважимо, що число обумовленості системи рівнянь (14) у режимі CASE = 0 дорівнює $0,684E+10$, в режимі CASE = 2 воно становить $0,434E+0,4$, що відповідно і впливає на кінцевий результат.

Запитання для самоперевірки

1. Як можна записати загальний розв'язок системи ЗДР (1), який задовольняє умову (2)? 2. З яких кроків складається прямий хід методу стрільби? 3. Як здійснюється зворотний хід методу стрільби? Які переваги та недоліки мають методи стрільби з відновленням розв'язку і з повторним розв'язуванням задачі Коші? 4. З якого кінця відрізка $[a, b]$ слід починати прямий хід методу стрільби і чому саме? 5. Які недоліки має метод стрільби при розв'язуванні крайових задач і коли саме вони проявляються? 6. У чому суть методу стрільби з ортогоналізацією? 7. Як можна записати алгоритм ортогоналізації систем векторів?

Контрольні завдання

1. Методами стрільби та стрільби з ортогоналізацією знайти розв'язок лінійної крайової задачі (1) — (3) при $a = 0, b = 1$, якщо

$$A(x) = \begin{vmatrix} 1 & 0 & x & 0 & 0 & 0 & 4 & 0 & - \\ 0 & 1-x & 0 & 2 & -1 & 0 & 0 & 0 & \\ 0 & 0 & -3 & 0 & 0 & 2 & 0 & 0 & \\ 0 & 0 & 0 & -4x & -1 & 0 & -1 & 0 & \\ 0 & 0 & 0 & 0 & \beta x & 0 & 0 & 2 & \\ 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 0 & \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & x-3 & - \end{vmatrix};$$

$$f(x) = \begin{bmatrix} 2 - 3x + 2x^2 - 4x \ln(1+x) \\ -3 - (1-x)(x+3) + \ln(1+x) \\ 1 - 6x - 2xe^{\alpha x} \\ 8x + (1+x) \ln(1+x) \\ (1+x)^{-1} - \beta x \ln(1+x) - 2(1+x) \\ (2-x)e^{\alpha x} \\ (1+2x) \ln(1+x) + x(x+1)^{-1} \\ -1 - (x-3)(1-x) \end{bmatrix};$$

$$P = \begin{bmatrix} 1 & 0 & 1 & 0 & 2 & -1 & 0 & 1 \\ 0 & 1 & 0 & -1 & 1 & -2 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 & 2 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 2 \end{bmatrix}; \quad y_a = \begin{bmatrix} 1 \\ -1 \\ 0 \\ 2 \\ 4 \end{bmatrix};$$

$$Q = \begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 5 & 2 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}; \quad y_b = \begin{pmatrix} 1 \\ 3 \\ 1 \end{pmatrix}.$$

Порівняти його з точним розв'язком $y(x) = (2x, x+3, 1-2x, 2, \ln(1+x), e^{\alpha x}, x \ln(1+x), 1-x)$ при різних значеннях параметрів α, β .

2. Прогин балки, що лежить на пружній основі, під дією поперечного навантаження описується рівнянням

$$(p(x) y''(x))'' - q(x) y(x) = f(x), \quad x \in (a, b)$$

і, залежно від способу закріплення її кінців, такими граничними умовами:

- а) $y''(a) = y''(b) = y'(a) = y'(b) = 0$;
 - б) $y''(a) = y''(b) = 0, y(b) = \alpha_1, y'(b) = \beta_1$;
 - в) $y'(a) = \alpha_0, y''(a) = \beta_0, y'''(a) = \gamma_0, y'''(b) = 0$;
 - г) $y(a) = \alpha_0, y'(a) = \beta_0, y'(b) = \alpha_1, y''(b) = \beta_1$;
 - д) $y''(a) = 0, y'(b) = \alpha_1, y''(b) = \beta_1, y'''(b) = \gamma_1$.
- Вхідні дані вибрати з множин

$$p(x) \in \{x^{-1}; 2(1+x)^{-1}; 1; x; 6x(3x^2+1)^{-1}; 2 + \sin(2-x)\};$$

$$q(x) \in \{\ln(2+x); |\cos x|; x^{-1}; 1; 3x; e^{x-5}\};$$

$$f(x) \in \{-3xe^x; -2x^{-2}; 6x+1; 2; -5x; 2 \cos x\};$$

$$a \in \{1; 1,5; 2; 2,5\}; \quad b \in \{a+1, a+2, a+3\};$$

$$\alpha_i, \beta_i, \gamma_i \in \{0; 1; 2; 3; 4; 5; 6\}, \quad i = 0, 1.$$

Кількість точок ортогоналізації $I = 6; 15$.

Розв'язуючи задачу за допомогою ЕОМ, надрукувати:

- а) лише значення функції $y(x)$;
- б) графік функції $y(x)$;
- в) значення функції $y(x)$ та її похідних;
- г) графіки функції $y(x)$ та її похідних;
- д) значення функції $y(x)$ у початковій та кінцевій точках інтегрування, а також у всіх внутрішніх точках ортогоналізації;

- е) значення функції $y(x)$ у кожній третій точці ортогоналізації;
 е) значення функції $y(x)$ у середній точці ортогоналізації та в кожній другій точці ліворуч і праворуч від неї.

§ 13. Методи Рітца та Бубнова — Гальоркіна розв'язування звичайних диференціальних рівнянь

Постановка задачі. Методом Рітца розв'язати крайову задачу для рівняння

$$Lu \equiv -(ku)'' + qu = f(x), \quad x \in [a, b]; \quad (1)$$

$$\alpha u'(a) - \beta u(a) = 0, \quad \gamma u'(b) + \delta u(b) = 0, \quad (2)$$

де $\alpha, \beta, \gamma, \delta$ — невід'ємні числа — такі, що $\alpha + \beta > 0, \gamma + \delta > 0, f(x) \in L_2(a, b), k(x) \in C^1[a, b], k(x) \geq k_0 > 0, q(x) \in C[a, b]$, причому $q(x) \geq q_0 > 0$ у випадку крайових умов другого роду ($\beta = \delta = 0$), $q(x) \geq 0$ в інших випадках.

▼ Методом Бубнова — Гальоркіна розв'язати крайову задачу

$$Mu \equiv Lu + pu' = f(x), \quad x \in [a, b]; \quad (3)$$

$$\alpha u'(a) - \beta u(a) = 0, \quad \gamma u'(b) + \delta u(b) = 0, \quad (4)$$

де $p(x) \in C[a, b]$.

Методичні вказівки. Метод Рітца для знаходження наближених розв'язків крайових задач ґрунтується на відомій теоремі про те, що коли розв'язок операторного рівняння

$$Lu = f \quad (5)$$

із самоспряженим і додатно визначеним у дійсному гільбертовому просторі H оператором існує, то цей розв'язок надає функціоналу енергії

$$F(u) = (Lu, u) - 2(f, u) \quad (6)$$

найменшого значення і, навпаки, елемент гільбертового простору, що реалізує мінімум функціонала (6), задовольняє рівняння (5) [13, с. 24].

Суть цього методу полягає ось у чому. Вибирається послідовність функцій $\varphi_1, \varphi_2, \dots, \varphi_n, \dots$, які задовольняють умови:

1) всі вони належать енергетичному простору H_L , який є поповненням області $D(L)$ визначення оператора L за нормою $\|u\|_L = [(Lu, u)]^{1/2}$;

2) при будь-якому n ці функції є лінійно незалежними.

Позначивши через H_n лінійну оболонку функцій $\varphi_1, \varphi_2, \dots, \varphi_n$, припустимо, що послідовність скінченновимірних підпросторів H_1, \dots, H_n є гранично щільною в просторі H_L , тобто кожний елемент $u \in H_L$ з будь-якою точністю може бути наближеним елементом $u_n \in H_n$, якщо n має велике значення. Тоді n -м наближенням буде елемент $u_n \in H_n$, який мінімізує функціонал (6) на цьому підпросторі.

Алгоритм пошуку цього елемента зводиться до виконання таких дій:

- 1) задаються конкретні значення n і $\{\varphi_i\}$, $\varphi_i \in H_L$;
- 2) наближений розв'язок відшукується у вигляді

$$u_n = \sum_{j=1}^n c_j \varphi_j; \quad (7)$$

3) коефіцієнти c_j визначаються з умови мінімізації функціонала $F(u_n)$ як функції c_j , що приводить до системи рівнянь $\frac{\partial F(u_n)}{\partial c_j} = 0$, $j = \overline{1, n}$. Знайшовши ці похідні, дістаємо СЛАР

$$\sum_{i=1}^n c_i [\varphi_i, \varphi_j] = (f, \varphi_j), \quad j = \overline{1, n}, \quad (8)$$

де $[\varphi_i, \varphi_j] = (L\varphi_i, \varphi_j)$.

Матриця цієї системи симетрична, оскільки оператор L самоспряжений, а її визначник є визначником Грама лінійно незалежних елементів і тому відмінний від нуля.

ТЕОРЕМА 1 [13, с. 29]. Якщо послідовність підпросторів $\{H_n\}$ є гранично щільною в просторі H_L , то наближені розв'язки u_n , побудовані за методом Рітца, при $n \rightarrow \infty$ збігаються до узагальненого розв'язку u_0 рівняння (5) за нормою простору H_L .

Елемент $u_0 \in H_L$, що реалізує мінімум функціонала (6), називається *узагальненим розв'язком* рівняння (5). Такий елемент існує і він єдиний [13, с. 27]. Якщо ж $u_0 \in D(L)$, то такий елемент буде *класичним розв'язком* рівняння (5).

▼ Розглянемо застосування методу Рітца для знаходження наближеного розв'язку крайової задачі (1), (2). Ця задача при крайових умовах першого роду ($\alpha = \gamma = 0$) еквівалентна задачі відшукування елемента $u \in W_2^1(a, b)$, який мінімізує функціонал

$$F(u) = \int_a^b (k(u')^2 + qu^2) dx - 2 \int_a^b f u dx, \quad (9)$$

а при крайових умовах другого роду ($\beta = \delta = 0$) та третього роду ($\alpha > 0$, $\beta > 0$, $\gamma > 0$, $\delta > 0$) — задачі відшукування елемента $u \in W_2^1(a, b)$, який мінімізує функціонал

$$F(u) = \int_a^b (k(u')^2 + qu^2) dx + \frac{\delta}{\gamma} k(b) u^2(b) + \frac{\beta}{\alpha} k(a) u^2(a) - 2 \int_a^b f u dx. \quad (10)$$

Тут $W_2^1(a, b)$ — гільбертів простір дійсних функцій, які є абсолютно неперервними на відрізку $[a, b]$ і мають інтегровну з квадратом першу похідну. Скалярний добуток у цьому просторі визначається виразом

$$(u, v) = \int_a^b (u'v' + uv) dx,$$

і з ним пов'язана норма $\|u\|_{W_2^1} = (u, u)^{1/2}$. Простір $W_2^1(a, b)$ складається з функцій простору $W_2^1(a, b)$, які перетворюються в нуль на кінцях відрізка $[a, b]$.

Система лінійних алгебраїчних рівнянь (8) для крайової задачі (1), (2) має вигляд

$$B\vec{c} = \vec{\Phi}, \quad (11)$$

де $\vec{c} = (c_1, \dots, c_n)^T$; $\vec{\Phi} = (\Phi_1, \dots, \Phi_n)^T$, $\Phi_i = \int_a^b f\varphi_i dx$, $i = \overline{1, n}$, а коефіцієнти b_{ij} матриці B порядку n визначаються виразом

$$b_{ij} = \int_a^b (k\varphi_j'\varphi_i' + q\varphi_j\varphi_i) dx \quad (12)$$

за крайових умов першого роду ($\alpha = \gamma = 0$) і виразом

$$b_{ij} = \int_a^b (k\varphi_j'\varphi_i' + q\varphi_j\varphi_i) dx + \frac{\delta}{\gamma} k(b) \varphi_j(b) \varphi_i(b) + \frac{\beta}{\alpha} k(a) \varphi_j(a) \varphi_i(a) \quad (13)$$

за крайових умов другого роду ($\beta = \delta = 0$) та третього роду ($\alpha > 0$, $\beta > 0$, $\gamma > 0$, $\delta > 0$).

▼ На відміну від методу Рітца метод Бубнова — Гальоркіна застосовується також для розв'язування несамоспряжених і незнаковизначених крайових задач. Його суть при розв'язуванні операторного рівняння

$$Mu = f \quad (14)$$

у гільбертовому просторі H , де $M = L + K$, а L — самоспряжений і додатно визначений оператор, полягає ось у чому. Вибирається послідовність функцій $\varphi_1, \varphi_2, \dots, \varphi_n, \dots$, які задовольняють умови:

- 1) всі вони належать енергетичному простору H_L ;
- 2) при будь-якому n ці функції є лінійно незалежними.

Припустимо, що послідовність скінченновимірних підпросторів H_1, \dots, H_n , де H_n — лінійна оболонка функцій $\varphi_1, \varphi_2, \dots, \varphi_n$, є щільною в просторі H_L . Тоді n -м наближенням буде елемент $u_n \in H_n$, для якого нев'язка $Mu_n - f$ є ортогональною до всіх функцій $\varphi_1, \varphi_2, \dots, \varphi_n$.

Реалізація цього алгоритму полягає ось у чому:

- 1) задаються конкретні значення n і $\{\varphi_i\}$, $\varphi_i \in H_L$;
- 2) наближений розв'язок відшукується у вигляді

$$u_n = \sum_{i=1}^n c_i \varphi_i; \quad (15)$$

3) коефіцієнти c_i визначаються з умови ортогональності нев'язки до функцій $\varphi_1, \varphi_2, \dots, \varphi_n$.

У результаті дістаємо таку СЛАР

$$\sum_{i=1}^n c_i ([\varphi_i, \varphi_j] + (k\varphi_i, \varphi_j)) = (f, \varphi_j)_{j=\overline{1, n}}, \quad (16)$$

де $[\varphi_i, \varphi_j] = (L\varphi_i, \varphi_j)$.

ТЕОРЕМА 2 [13, с. 49]. Нехай: 1) рівняння (14) має єдиний узагальнений розв'язок u_0 , який належить простору H_L ; 2) форма $L(u, v) = [u, v] + (Ku, v) \in H_L$ -визначеною і H_L -обмеженою, тобто виконуються співвідношення $L(u, u) \geq \gamma_0^2 \|u\|_L^2$, $|(Lu, v)| \leq \gamma_1^2 \|u\|_L \|v\|_L$ (де $\gamma_0, \gamma_1 = \text{const}$); 3) послідовність підпросторів $\{H_n\}$ є гранично щільною в просторі H_L . Тоді при будь-якому скінченному n СЛАР (16) матиме єдиний розв'язок, наближений розв'язок u_n при $n \rightarrow \infty$ збігається до u_0 в нормі простору H_L .

Елемент $u_0 \in H_L$ називається *узагальненим розв'язком* рівняння (14), якщо при будь-якому $v \in H_L$ виконується рівність $[u, v] + (Ku, v) = (f, v)$.

▼ Розглянемо застосування методу Бубнова — Гальоркіна для знаходження наближеного розв'язку крайової задачі (3), (4). СЛАР (16) для неї має вигляд

$$A\vec{c} = \vec{\Phi}. \quad (17)$$

Тут $\vec{c} = (c_1, \dots, c_n)^T$; $\vec{\Phi} = (\Phi_1, \dots, \Phi_n)^T$, $\Phi_i = \int_a^b f \varphi_i dx$, $i = \overline{1, n}$;

коефіцієнти матриці A порядку n визначаються виразом $a_{ij} = b_{ij} + c_{ij}$, де $c_{ij} = \int_a^b \rho \varphi_i' \varphi_j dx$, а значення b_{ij} залежно від типу крайових умов обчислюються за формулою (12) або (13).

▼ Наведемо приклад системи функцій $\{\varphi_n(x)\}$, яка задовольняє умови, зазначені вище, і називається *поліномною*:

$$\varphi_x(x) = (x-a)^n (b-x), \quad n = 1, 2, 3, \dots \quad (\alpha = \gamma = 0);$$

$$\varphi_1(x) = 1; \quad \varphi_2(x) = (x-a)^2 \left(x-b - \frac{b-a}{2}\right);$$

$$\varphi_n(x) = (x-a)^{n-1} (b-x)^2, \quad n = 3, 4, 5, \dots \quad (\beta = \delta = 0);$$

$$\varphi_1(x) = (x-a)^2 \left(x-b - \frac{\gamma(b-a)}{2\gamma + \delta(b-a)} \right);$$

$$\varphi_2(x) = (b-x)^2 \left(x-a + \frac{\alpha(b-a)}{2\alpha + \beta(b-a)} \right);$$

$$\varphi_n(x) = (x-a)^{n-1} (b-x)^2, \quad n = 3, 4, 5, \dots$$

$$(\alpha > 0, \beta > 0, \gamma > 0, \delta > 0).$$

▼ Основний обсяг обчислень при реалізації методів Рітца та Бубнова — Гальоркіна припадає на визначення компонентів вектора $\vec{\Phi}$ і коефіцієнтів b_{ij} (a_{ij}) матриць B (A), а також на чисельне розв'язування СЛАР (11), (17). У зв'язку з цим важливо дослідити вплив похибок, допущених при побудові СЛАР (11), (17) і чисельному розв'язуванні їх, на точність наближеного розв'язку крайової задачі. Цю проблему докладно розглянуто в [15, с. 16—94, с. 157—171; 18, с. 224—237].

Звичайно до системи функцій $\{\varphi_n\}$, крім вимог, наведених вище, ставляться ще додаткові вимоги, а саме: вона має бути такою, щоб чисельний процес був стійким, числа обумовленості матриць B (A) були рівномірно обмеженими відносно n , нев'язка рівняння при $n \rightarrow \infty$ прямувала до нуля.

Способи побудови систем таких функцій, які задовольняють зазначені вимоги, описано в [18, с. 224—237; 15, с. 157—171]. Як приклад наведемо таку систему функцій:

$$\varphi_n(x) = \frac{1}{n} \sin \frac{\pi n (x-a)}{b-a}, \quad n = 1, 2, 3, \dots \quad (\alpha = \gamma = 0);$$

$$\varphi_1(x) = 1; \quad \varphi_n(x) = \frac{1}{n-1} \cos \frac{(n-1)\pi(x-a)}{b-a}, \quad n = 2, 3, \dots$$

$$(\beta = \delta = 0); \quad \varphi_1(x) = 1; \quad \varphi_2(x) = x;$$

$$\varphi_n(x) = \frac{1}{n-2} \sin \frac{(n-2)\pi(x-a)}{b-a}, \quad n = 3, 4, 5 \dots$$

$$(\alpha > 0, \beta > 0, \gamma > 0, \delta > 0).$$

Приклад використання підпрограм. Для обчислення коефіцієнтів матриці і вектора правих частин СЛАР (17) призначена підпрограма GALOR. У ній з використанням підпрограм SIMP обчислюються інтеграли за формулою Сімсона, а підпрограми-функції AIJ та FII застосовуються відповідно для обчислення значень функції $k(x)$, $\varphi_i(x)$, $\varphi_j(x) + q(x)\varphi_i(x)\varphi_j(x) + p(x)\varphi_j(x)\varphi_i(x)$ і функції $f(x)\varphi_i(x)$.

▼ Використання підпрограм GALOR проілюструємо на такому прикладі, Методом Бубнова — Гальоркіна розв'яжемо крайову задачу (3), (4) при $k(x) = 1$, $p(x) = 1$, $q(x) = 1$, $f(x) = -x^2 - x + 4$, $\alpha = \beta = \gamma = \delta = 1$, $a = 0$, $b = 1$, точний розв'язок якої має вигляд

$$u^*(x) = -x^2 + x + 1.$$

Узявши чотири функції з поліномної координатної системи, для знаходження розв'язку СЛАР (17) використаємо підпрограму SISVD. Головна програма, що містить підпрограми GALOR і SISVD, а також підпрограми-функції, які описують коефіцієнти $k(x)$, $p(x)$, $q(x)$, $f(x)$ і задають поліномну координатну систему з її похідними, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ КРАЙОВИХ ЗАДАЧ
С ДЛЯ ЗДР ДРУГОГО ПОРЯДКУ МЕТОДОМ
С БУБНОВА — ГАЛЬОРКІНА
С
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
EXTERNAL QA,P,QU,F0
REAL C(4),AM(4,4),FR(4),SIGMA(10),U(100),V(100)
С МАСИВИ SIGMA,U,V — ВИКОРИСТОВУЮТЬСЯ В ПІДПРОГРАМИ
С SISVD
С МАСИВ С МІСТИТЬ ВЕКТОР РОЗВ'ЯЗКІВ СИСТЕМИ МЕТОДУ
С БУБНОВА — ГАЛЬОРКІНА
COMMON/B1/ICOND,NF
COMMON/BB/A,B,ALFA,BETA,GAMA,DELTA
С ПАРАМЕТРИ КРАЙОВИХ УМОВ
DATA ALFA,BETA,GAMA,DELTA/4*1./
С NF — НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ РЕЗУЛЬТАТІВ
NF=3
С ЧИСЛО КООРДИНАТНИХ ФУНКЦІЙ
N=4
С [A,B] — ВІДРІЗОК, НА ЯКОМУ ВІДШУКУЄТЬСЯ РОЗВ'ЯЗОК
A=0.
B=1.
С ТОЧНІСТЬ ОБЧИСЛЕННЯ ІНТЕГРАЛІВ
EPS=1.E-05
С ПІДПРОГРАМА GALOR ОБЧИСЛЮЄ МАТРИЦЮ AM І ВЕКТОР FR
CALL GALOR(A,B,ALFA,BETA,GAMA,DELTA,N,EPS,QA,
* P,QU,F0,AM,FR)
* IF(ICOND.EQ.1)STOP
С ПІДПРОГРАМА SISVD ЗНАХОДИТЬ РОЗВ'ЯЗОК СЛАП
С AM*C=FR
CALL SISVD(AM,FR,N,N,C,SIGMA,U,V,R)
WRITE(NF,100)R
SIGMAX=SIGMA(1)
SIGMIN=SIGMAX
DO 10 L=2,N
IF(SIGMA(L).GT.SIGMAX) SIGMAX=SIGMA(L)
IF(SIGMA(L).LT.SIGMIN) SIGMIN=SIGMA(L)
10 CONTINUE
WRITE(NF,101) SIGMIN,SIGMAX
С ВИВЕДЕННЯ КОЕФІЦІЕНТІВ РОЗКЛАДУ
WRITE(NF,102)
WRITE(NF,103) (I,C(I),I=1,N)
STOP
100 FORMAT(3X,'*SISVD* НЕВ'ЯЗКА: ',E15.7)
101 FORMAT(9X,'МІНІМАЛЬНЕ СИНГУЛЯРНЕ ЧИСЛО:',
* E15.7/,9X,'МАКСИМАЛЬНЕ СИНГУЛЯРНЕ',
* 'ЧИСЛО:',E15.7)
102 FORMAT(3X,'*GALOR* КОЕФІЦІЕНТИ РОЗКЛАДУ')
103 FORMAT(11X,'C(' ,I2,')=' ,E15.7)
END
FUNCTION QA(X)
С ОБЧИСЛЕННЯ ФУНКЦІЇ K(X)
QA=1.
RETURN
END

```

```

FUNCTION QU(X)
С ОБЧИСЛЕННЯ ФУНКЦІЇ Q(X)
QU=1.
RETURN
END

```

```

FUNCTION P(X)
С ОБЧИСЛЕННЯ ФУНКЦІЇ P(X)
P=1.
RETURN
END

```

```

FUNCTION F0(X)
С ОБЧИСЛЕННЯ ФУНКЦІЇ F(X)
F0=-X*X-X+4.
RETURN
END

```

Обчислення за цією програмою дають такий результат:

GALOR КООРДИНАТНА СИСТЕМА, ЯКА ВІДПОВІДАЄ КРАЙОВИМ
УМОВАМ ТРЕТЬОГО РОДУ
SIMP УСІ ІНТЕГРАЛИ ОБЧИСЛЕНО З ЗАДАНОЮ ТОЧНІСТЮ
SISVD НЕВ'ЯЗКА: 0.3634211E-06

МІНІМАЛЬНЕ СИНГУЛЯРНЕ ЧИСЛО: 0.1078218E-02
МАКСИМАЛЬНЕ СИНГУЛЯРНЕ ЧИСЛО: 0.4873970E+00

GALOR КОЕФІЦІЄНТИ РОЗКЛАДУ
C(1)=-0.2999920E+01
C(2)= 0.2999822E+01
C(3)= 0.1632873E-02
C(4)=-0.1006007E-02

Точний розв'язок у вигляді його розкладу по координатних функціях $\varphi_i(x)$,
 $i = \overline{1, 4}$ можна подати так:

$$u^*(x) = -3\varphi_1(x) + 3\varphi_2(x) + 0\varphi_3(x) + 0\varphi_4(x),$$

тобто $c_1 = -3$, $c_2 = 3$, $c_3 = c_4 = 0$.

Абсолютні похибки визначення коефіцієнтів c_i , $i = \overline{1, 4}$ становлять: $\delta_1 = 0,000046$; $\delta_2 = 0,000185$; $\delta_3 = 0,001374$; $\delta_4 = 0,002058$.

Запитання для самоперевірки

1. За яких умов для оператора вихідної задачі можна застосовувати метод Рітца? 2. Як формулюється теорема, що лежить в основі методу Рітца? 3. Який функціонал мінімізується в методі Рітца? 4. За яким алгоритмом реалізується метод Рітца? 5. Як формулюється теорема про збіжність методу Рітца? 6. Який з методів — Рітца чи Бубнова — Гальоркіна застосовується більш широко? 7. За яким алгоритмом реалізується метод Бубнова — Гальоркіна? 8. Як формулюється теорема про збіжність методу Бубнова — Гальоркіна? 9. Який функціонал мінімізується в методі Рітца при розв'язуванні першої крайової задачі для ЗДР другого порядку? 10. Який функціонал мінімізується в методі Рітца при розв'язуванні третьої крайової задачі для ЗДР другого порядку?

Контрольні завдання

1. Методом Бубнова — Гальоркіна знайти розв'язок крайової задачі (3), (4) при $k(x) = p(x) = q(x) = 1$, $f(x) = mx^3 + (3m + l)x^2 + (2l + c - 6m)x + (c + d - 2l)$, $\beta = (3ma^2 + 2la + c)$, $\alpha = (ma^3 + la^2 + ca + d)$, $\gamma = mb^3 + lb^2 +$

$+cb + d$, $\delta = -(3mb^2 + 2lb + c)$, якщо параметри a, b, m, l, c, d ($a = 0; b = 1$) мають такі значення:

- 1) $d = 1, c = 2, m = 1, l = -3$; 2) $d = 2, c = 1, m = 1, l = -3$;
- 3) $d = 1, c = 2, m = 0, l = -2$; 4) $d = 2, c = 1, m = -2, l = 0$;
- 5) $d = 2, c = 1, m = -1, l = 0$; 6) $d = 1, c = 2, m = -1, l = 0$;
- 7) $d = 1, c = 2, m = -2, l = 0$; 8) $d = 2, c = 1, m = 0, l = -1$;
- 9) $d = 2, c = 1, m = 0, l = -2$; 10) $d = 0, c = 1, m = 0, l = -1$;
- 11) $d = 1, c = 0, m = 0, l = 0$.

Точним розв'язком задачі є функція $u_0(x) = mx^3 + lx^2 + cx + d$. Параметри a, b, m, l, c, d , кількість функцій $\varphi_n(x)$, а також вигляд їх визначаються викладачем.

2. Методом Рітца знайти розв'язок крайової задачі (1), (2) при $k(x) = 1, q(x) = x, f(x) = mx^4 + lx^3 + cx^2 + (d - 6m)x - 2l, \alpha = ma^3 + la^2 + ca + d, \beta = 3ma^2 + 2la + c, \gamma = mb^3 + lb^2 + cb + d, \delta = -(3mb^2 + 2lb + c)$, якщо параметри a, b, m, l, c, d ($a = 0; b = 1$) мають такі значення:

- 1) $d = 1, c = 2, m = 1, l = -3$; 2) $d = 2, c = 1, m = 1, l = -3$;
- 3) $d = 1, c = 2, m = 0, l = -2$; 4) $d = 2, c = 1, m = -2, l = 0$;
- 5) $d = 2, c = 1, m = -1, l = 0$; 6) $d = 1, c = 2, m = -1, l = 0$;
- 7) $d = 1, c = 2, m = -2, l = 0$; 8) $d = 2, c = 1, m = 0, l = -1$;
- 9) $d = 2, c = 1, m = 0, l = -2$; 10) $d = 0, c = 1, m = 0, l = -1$.

Точним розв'язком задачі є функція $u_0(x) = mx^3 + lx^2 + cx + d$. Параметри a, b, m, l, c, d , кількість функцій $\varphi_n(x)$, а також вигляд їх визначаються викладачем.

3. На прикладі розв'язання методом Рітца (використати поліномну координатну систему) крайової задачі

$$-u'' + u = -x^2 + x + \alpha;$$

$$u(0) = 0, \quad u(1) = 0,$$

точним розв'язком якої є $u_0(x) = x(1-x)$, тобто $u_0(x) = \varphi_1(x)$, з'ясувати вплив похибок, допущених при побудові СЛАР (16) і чисельному розв'язуванні її, на точність обчислення коефіцієнтів розкладу (7). Узяти $n = 2, 8$.

§ 14. Метод сіток розв'язування крайових задач для звичайного диференціального рівняння другого порядку

Постановка задачі. Знайти розв'язок диференціального рівняння другого порядку

$$\frac{1}{x^\gamma} \frac{d}{dx} \left(x^\gamma k(x) \frac{du}{dx} \right) - q(x)u = -f(x), \quad a < x < b, \quad (1)$$

де $q(x) \geq 0, 0 < c_1 \leq k(x) \leq c_2$ — відомі функції, а $k, q, f \in Q_2[a, b]$ з граничними (крайовими) умовами

$$\alpha_1 x^\gamma k u' = \delta_1 u - \mu_1 |_{x=a}; \quad (2)$$

$$-\alpha_2 x^\gamma k u' = \delta_2 u - \mu_2 |_{x=b}. \quad (3)$$

Тут $\alpha_i, \delta_i, \mu_i, i = 1, 2$ — задані коефіцієнти; γ — параметр задачі, причому $\gamma = 0$ відповідає плоскій симетрії (декартова система координат).

нат), $\gamma = 1$ — циліндричній симетрії (циліндрична система координат), $\gamma = 2$ — сферичній симетрії (сферична система координат).

Якщо $\gamma \neq 0$ і $a = 0$, то замість умови (2) задається гранична умова

$$u'(0) = 0.$$

Коефіцієнти α_1, α_2 дорівнюють 0 або 1 і визначають тип граничної умови.

Методичні вказівки. Для розв'язування граничної задачі (1) — (3) методом скінченних різниць уведемо на відрізку $[a, b]$ нерівномірну сітку $\hat{\omega}_h = \{x_{i+1} = x_i + h_{i+1}, i = 0, 1, 2, \dots, N-1, x_0 = a, h_i — \text{задані кроки; } \sum_{i=1}^N h_i = b - a\}$.

На цій сітці апроксимуємо рівняння (1) такою різницевою схемою, яку можна дістати інтегро-інтерполяційним методом [22, с. 156, 157]:

$$\Delta y_i = \frac{1}{h_i x_i \gamma} \left[x_{i+1/2}^\gamma a_{i+1} \frac{y_{i+1} - y_i}{h_{i+1}} - x_{i-1/2}^\gamma a_i \frac{y_i - y_{i-1}}{h_i} \right] - d_i y_i = -\varphi_i.$$

При цьому допускаємо, що точки можливих розривів коефіцієнтів $k(x), q(x), f(x)$ суміщуються з вузлами сітки $\hat{\omega}_h$. Таку сітку позначатимемо $\hat{\omega}_h(k, q, f)$.

Коефіцієнти різницевої схеми (4) визначаються виразами

$$a_i = k(x_{i-1/2}); \quad d_i = \frac{h_i q(x_{i-0}) + h_{i+1} q(x_{i+0})}{2h_i};$$

$$\varphi_i = \frac{h_i f(x_{i-0}) + h_{i+1} f(x_{i+0})}{2h_i},$$
(5)

де $h_i = \frac{h_i + h_{i+1}}{2}$.

Якщо k, q, f — неперервні функції, то

$$a_i = k(x_{i-1/2}); \quad d_i = d(x_i); \quad \varphi_i = f(x_i).$$
(5a)

Вибір коефіцієнтів у вигляді (5), (5a) забезпечує другий порядок апроксимації різницевої схеми (4).

▼ Скінченно-різницеve співвідношення (4) доповнимо такими різницеvими граничними умовами:

$$\alpha_1 a_1 x_{1/2}^\gamma y_{x,1} = \bar{\delta}_1 y_0 - \bar{\mu}_1; \tag{6}$$

$$-\alpha_2 a_N x_{N-1/2}^\gamma y_{x,N} = \bar{\delta}_2 y_N - \bar{\mu}_N, \tag{7}$$

де

$$\begin{aligned}\bar{\delta}_1 &= \delta_1 + \alpha_1 \frac{h_1}{2} q_0 x_0^\gamma; & \bar{\mu}_1 &= \mu_1 + \frac{\alpha_1 h_1}{2} f_0 x_0^\gamma; \\ \bar{\delta}_2 &= \delta_2 + \alpha_2 \frac{h_N}{2} q_N x_N^\gamma; & \bar{\mu}_2 &= \mu_2 + \frac{\alpha_2 h_N}{2} f_N x_N^\gamma.\end{aligned}\quad (8)$$

Якщо $a = 0$ і $\gamma \neq 0$, то додається гранична умова (2), яку можна апроксимувати різницеvim співвідношенням

$$a_1 y_{x_1} x_{1/2}^\gamma = \frac{h_1}{2(\gamma+1)} (q_0 y_0 - f_0). \quad (6a)$$

ТЕОРЕМА. Різницева схема (4) — (8) для $k, q, f \in Q^2 [a, b]$ на сітці $\hat{\omega}_h(k, q, f)$ має другий порядок точності, тобто

$$\|z\|_c = \|y - u\|_c \leq M \bar{h}^{-2}; \quad \bar{h} = \left(\prod_{i=1}^N h_i \right)^{1/N}.$$

Запишемо скінченно-різницеvu схему (4) — (8) у вигляді СЛАР з тридіагональною матрицею, а саме:

$$\begin{aligned}A_i y_{y-1} - C_i y_i + B_i y_{i+1} &= -F_i, \quad i = 1, 2, \dots, N-1; \\ y_0 &= x_1 y_1 + v_1; \\ y_N &= x_2 y_{N-1} + v_2,\end{aligned}\quad (9)$$

де

$$\begin{aligned}A_i &= \frac{x_{i-1/2}^\gamma a_i}{h_i}; & B_i &= \frac{x_{i+1/2}^\gamma a_{i+1}}{h_{i+1}}; \\ C_i &= A_i + B_i + \bar{h}_i x_i^\gamma d_i; & F_i &= \bar{h}_i x_i^\gamma \varphi_i; \\ x_1 &= \frac{\alpha_1 a_1 x_{1/2}^\gamma}{h_1 \bar{\delta}_1 + \alpha_1 a_1 x_{1/2}^\gamma}; & v_1 &= -\frac{h_N \bar{\mu}_1}{h_N \bar{\delta}_1 + \alpha_1 a_1 x_{1/2}^\gamma}; \\ x_2 &= \frac{\alpha_2 a_N x_{N-1/2}^\gamma}{h_N \bar{\delta}_2 + \alpha_2 a_N x_{N-1/2}^\gamma}; & v_2 &= \frac{h_N \bar{\mu}_2}{h_N \bar{\delta}_2 + \alpha_2 a_N x_{N-1/2}^\gamma}.\end{aligned}\quad (10)$$

Похибка розв'язку СЛАР (9), яка виникає через неточне задавання коефіцієнтів або прави частин її рівнянь, визначається числом обумовленості тридіагональної матриці A . Обумовленість таких матриць $\nu(A)$ дорівнює CN^2 , де C — стала порядку одиниці.

Таким чином, якщо похибка заокруглення $\epsilon < 10^{-6}$, то для СЛАР розмірністю $N \leq 100$ відповідна похибка розв'язку буде допустимою. Якщо ж порядок N становить кілька сотень, то обчислення на ЕОМ потрібно виконувати з подвійною точністю.

Розв'язок СЛАР (9) шукатимемо методом прогонки [22, с. 40, 41], користуючись формулами

$$\begin{aligned} \eta_{i+1} &= B_i / (C_i - \eta_i A_i), \quad \eta_1 = \kappa_1; \\ \beta_{i+1} &= (A_i \beta_i + F_i) / (C_i - \eta_i A_i), \quad \beta_1 = \gamma_1, \quad i = 1, 2, \dots, N-1; \\ y_N &= (v_2 + \kappa_2 \beta_N) / (1 + \alpha_N \kappa_2); \\ y_i &= \eta_{i+1} y_{i+1} + \beta_{i+1}, \quad i = N-1, N-2, \dots, 1, 0. \end{aligned} \quad (11)$$

▼ Для контролю точності наближеного розв'язку можна застосувати метод Рунге [22, с. 175—180]. Якщо y_h — наближений розв'язок СЛАР на сітці $\hat{\omega}_h(k, q, f)$, а $y_{0,5h}$ — на сітці $\hat{\omega}_{0,5h}(k, q, f)$, то

$$\|z_h\|_{C_h}^2 \approx \frac{4}{3} \|y_h - y_{0,5h}\|_{C_h}, \quad (12)$$

де

$$z_{h,i} = u(x_i) - y_{h,i}; \quad \|z_h\|_{C_h} = \max_{x_i \in \hat{\omega}_h(k, q, f)} |z_{h,i}|.$$

При цьому лінійна комбінація $\tilde{y} = \frac{4}{3} y_{0,5h} - \frac{1}{3} y_h$, визначена на сітці $\hat{\omega}_h(k, q, f)$, має четвертий порядок точності по \bar{h} .

Приклад використання підпрограм. Розглянемо чисельну реалізацію на ЕОМ скінченно-різницевого методу розв'язування рівняння (1) при $a = 0$, $b = 1$ та $\gamma = 0$, коли

$$\begin{aligned} k(x) &= \begin{cases} k_1 > 0, & 0 < x < x_0; \\ k_2 > 0, & x_0 < x < 1; \end{cases} \\ q(x) &= \begin{cases} q_1 = \frac{k_1 r_1 (r_1 - 1)}{x_0^2}, & 0 < x < x_0; \\ q_2 = \frac{k_2 r_2 (r_2 - 1)}{x_0^2}, & x_0 < x < 1; \end{cases} \end{aligned} \quad (13)$$

$$f(x) = \begin{cases} -(a_1 + x/x_0)^{r_1-2} q_1 (1 - (a_1 + x/x_0)^{r_1}), & 0 < x < x_0; \\ -(a_2 + x/x_0)^{r_2-2} q_2 (1 - (a_2 + x/x_0)^{r_2}), & x_0 < x < 1. \end{cases} \quad (14)$$

Нехай граничними умовами є такі:

$$u(0) = a_1^{r_1}; \quad (15)$$

$$-k_2 \frac{du(1)}{dx} = \frac{r_2 k_2}{x_0} \left(a_2 + \frac{1}{x_0} \right)^{r_2-1}.$$

де

$$\begin{aligned} r_1 &= 2; \quad r_2 = 4; \quad a_1 = 1,5; \quad x_0 = 0,5; \\ a_2 &= (1 + a_1)^{r_1/r_2} - 1; \quad k_2 = \frac{a_1 r_1 (1 + a_2)}{r_2 (1 + a_1)}. \end{aligned} \quad (16)$$

Точний розв'язок задачі (1), (13) — (16) має вигляд

$$u(x) = \begin{cases} (a_1 + x/x_0)^{r_1}, & 0 \leq x \leq x_0; \\ (a_2 + x/x_0)^{r_2}, & x_0 < x \leq 1. \end{cases} \quad (17)$$

▼ Для побудови сітки $\hat{\omega}_h(k, q, f)$ на відрізьку $[a, b]$ можна використати підпрограму GRID. Задаючись при цьому числом інтервалів N сітки, розмірністю k масиву, його елементами $RX(I)$, службовими параметрами $IR \neq 0$ та $ID \neq 0$, дістаємо сітку з $N + 1$ вузлом. Вона будується шляхом поділу кожного з відрізків $[RX(I), RX(I + 1)]$ на додаткову кількість вузлів, яка пропорційна його довжині. Якщо $RX(I)$ і $RX(K)$ є відповідно початком і кінцем відрізка інтегрування, а $RX(2), \dots, RX(K - 1)$ — можливими точками розриву коефіцієнтів, то за підпрограмою GRID будується сітка $\hat{\omega}(k, q, f)$.

Для згущення сітки $\hat{\omega}_h(k, q, f)$ шляхом збільшення вдвічі кількості її вузлів, тобто побудови сітки $\hat{\omega}_{0,5h}(k, q, f)$, треба повторно викликати підпрограму GRID у тому ж самому програмному модулі, задаючи $ID = 2$ і не змінюючи решту параметрів.

Задаючи $IR = 0$ і значення N, X , можна зафіксувати множину вузлів сітки. При цьому обчислюється масив кроків сітки. Решта параметрів можуть бути незначеними.

▼ Підпрограма ZKR1 дає змогу знайти наближений розв'язок граничної задачі (1) — (3) скінченно-різницеvim методом на сітці $\hat{\omega}_h(k, q, f)$. Вона ґрунтується на використанні другого порядку апроксимації (5) і методі прогонки за формулами (9) — (11). Перш ніж звернутися до підпрограми ZKR1, слід скласти підпрограми AK(x), Q(x), F(x) для визначення коефіцієнтів $k(x), q(x), f(x)$ рівняння (1) і включити їх у список формальних параметрів підпрограми ZKR1. Коефіцієнти граничних умов $\alpha_i, \delta_i, \mu_i$ ($i = 1, 2$), а також параметр типу симетрії γ передаються через COMMON-блоки /GR/ AL1, S1, M1, AL2, S2, M2 і /GG/G відповідно. Результатом роботи підпрограми є формування масиву X, в якому знаходиться наближений розв'язок задачі (1), (14) — (16) у вузлах сітки.

Головна програма, що реалізує розв'язок розглядуваної задачі, має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C ГОЛОВНА ПРОГРАМА РОЗВ'ЯЗУВАННЯ МЕТОДОМ                               C
C СІТОК ГРАНИЧНОЇ ЗАДАЧІ ДЛЯ САМОСПРЯЖЕНОГО ЗДР                     C
C ДРУГОГО СТУПЕНЯ                                                       C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    DIMENSION Y(401),Y1(201),X(401),H(401),RX(15)
    REAL M1,M2
    INTEGER G
    EXTERNAL F,Q,AK
    COMMON/GR/AL1,S1,M1,AL2,S2,M2,/GG/G
    COMMON/PARM/AK1,AK2,R1,R2,A1,A2,/PR/X0
C ЗАДАЙТЕ ПАРАМЕТРИ ГРАНИЧНИХ УМОВ І ПАРАМЕТРИ
C ВАРІАНТА РОЗРАХУНКУ (14)
    DATA AL1,S1,AL2,S2/0.,1.,1.,0./
    DATA R1,R2,A1,AK1/2.,4.,1.5,3./
    TYPE 100
C ВВЕДІТЬ НОМЕР ПРИСТРОЮ ДЛЯ ВИВЕДЕННЯ ІНФОРМАЦІЇ
    ACCEPТ 101,NV
    TYPE 102
C ВВЕДІТЬ ПАРАМЕТР ТИПУ СИМЕТРІЇ
    ACCEPТ 101,G
    TYPE 103
  
```

C ВВЕДИТЬ МАСИВ ФІКСОВАНИХ ВУЗЛІВ СІТКИ

```
АСЦЕРТ 104, RX(1)
TYPE 103
АСЦЕРТ 104, RX(2)
TYPE 103
АСЦЕРТ 104, RX(3)
TYPE 103
АСЦЕРТ 104, RX(4)
TYPE 103
АСЦЕРТ 104, RX(5)
X0=RX(2)
N=50
N2=2*N
N1=N+1
K=3
```

C ПОЗРАХУНОК УСІХ ПАРАМЕТРІВ ВАРІАНТА ЗАВДАННЯ

```
CALL SPR
M1=A1**R1
M2=R2*AK2/X0*(A2+1./X0)**(R2-1.)
IR=1
```

C ОБЧИСЛЕННЯ СІТКИ З КРОКОМ H

```
CALL GRID(N,K,X,H,RX,IR,1)
```

C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ГРАНИЧНОЇ ЗАДАЧІ НА ЦІЙ СІТЦІ

```
CALL ZKR1(F,Q,AK,Y1,H,X,N)
```

C ОБЧИСЛЕННЯ СІТКИ З КРОКОМ H/2.

```
CALL GRID(N2,K,X,H,RX,IR,2)
```

C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ГРАНИЧНОЇ ЗАДАЧІ НА БІЛЬШ

C ЩІЛЬНІЙ СІТЦІ

```
CALL ZKR1(F,Q,AK,Y,H,X,N2)
```

C МЕТОДОМ РУНГЕ ОБЧИСЛЮЄТЬСЯ ПОХИБКА РОЗВ'ЯЗКУ

```
EPS=0.
```

```
DO 10 J=1,N1
```

```
    S=ABS(Y1(J)-Y(2*J-1))
```

```
    IF (S.GT.EPS) EPS=S
```

```
10 CONTINUE
```

```
    EPS=EPS*4./3.
```

```
DO 20 J=1,N1
```

```
20 Y(J)=(4.*Y(2.*J-1)-Y1(J))/3.
```

```
WRITE(NV,105)(X(2*I-1),Y(I),U(X(2*I-1)),I=1,N1)
```

```
WRITE(NV,106)EPS
```

```
STOP
```

```
100 FORMAT(1H$, 'NV=')
```

```
101 FORMAT(12)
```

```
102 FORMAT(1H$, 'G=')
```

```
103 FORMAT(1H$, 'RX(I)=')
```

```
104 FORMAT(E12.5)
```

```
105 FORMAT(5X, 'НАБЛИЖЕНИЙ РОЗВ'ЯЗОК', 6X, 'ТОЧНИЙ',
```

```
    * 'РОЗВ'ЯЗОК'//'(3X, 'X=', E12.4, 3X, 'Y=', E13.5, 3X,
```

```
    * 'YT=', E13.5))
```

```
106 FORMAT(3X, '/', 'ПОХИБКА РОЗВ'ЯЗКУ=', E12.5)
```

```
END
```

```
FUNCTION F(X)
```

C ОБЧИСЛЮЄ ПРАВУ ЧАСТИНУ ДИФЕРЕНЦІАЛЬНОГО РІВНЯННЯ

```
COMMON/PARM/AK1,AK2,R1,R2,A1,A2,/PR/X0
```

```

IF(X.LT.X0) F=-(A1+X/X0)**(R1-2.)*
* AK1*R1*(R1-1.)/X0**2*(1.-(A1+X/X0)**2)
IF(X.GT.X0) F=-(A2+X/X0)**(R2-2.)*
* AK2*R2*(R2-1.)/X0**2*(1.-(A2+X/X0)**2)
RETURN
END

```

```

FUNCTION Q(X)
C ОБЧИСЛЮЄ КОЕФІЦІЄНТ РІВНЯННЯ
COMMON/PARM/AK1,AK2,R1,R2,A1,A2,/PR/X0
IF (X.LT.X0) Q=AK1*R1*(R1-1.)/X0**2
IF (X.GT.X0) Q=AK2*R2*(R2-1.)/X0**2
RETURN
END

```

```

FUNCTION AK(X)
C ОБЧИСЛЮЄ КОЕФІЦІЄНТ РІВНЯННЯ
COMMON/PARM/AK1,AK2,R1,R2,A1,A2,/PR/X0
IF(X.LT.X0) AK=AK1
IF(X.GT.X0) AK=AK2
RETURN
END

```

```

SUBROUTINE SPR
C ОБЧИСЛЮЄ НЕОБХІДНІ ПАРАМЕТРИ ВАРІАНТА ЗАВДАННЯ
COMMON/PARM/AK1,AK2,R1,R2,A1,A2
A2=(1.+A1)**(R1/R2)-1.
AK2=AK1*R1*(1.+A2)/R2/(1.+A1)
RETURN
END

```

```

FUNCTION U(X)
C ОБЧИСЛЮЄ ТОЧНИЙ РОЗВ'ЯЗОК ГРАНИЧНОЇ ЗАДАЧІ
COMMON/PARM/AK1,AK2,R1,R2,A1,A2,/PR/X0
IF(X.LE.X0) U=(A1+X/X0)**R1
IF(X.GT.X0) U=(A2+X/X0)**R2
RETURN
END

```

Обчислення на ЕОМ дають такий результат:

НАБЛИЖЕНИЙ РОЗВ'ЯЗОК		ТОЧНИЙ РОЗВ'ЯЗОК	
X = 0.0000E + 00	Y = 0.22500E + 01	YT = 0.22500E + 01	
X = 0.2000E - 01	Y = 0.23716E + 01	YT = 0.23716E + 01	
X = 0.4000E - 01	Y = 0.24964E + 01	YT = 0.24964E + 01	
X = 0.6000E - 01	Y = 0.26244E + 01	YT = 0.26244E + 01	
X = 0.8000E - 01	Y = 0.27556E + 01	YT = 0.27556E + 01	
X = 0.1000E + 00	Y = 0.28900E + 01	YT = 0.28900E + 01	
X = 0.1200E + 00	Y = 0.30276E + 01	YT = 0.30276E + 01	
X = 0.1400E + 00	Y = 0.31684E + 01	YT = 0.31684E + 01	
X = 0.1600E + 00	Y = 0.33124E + 01	YT = 0.33124E + 01	
X = 0.1800E + 00	Y = 0.34596E + 01	YT = 0.34596E + 01	
X = 0.2000E + 00	Y = 0.36100E + 01	YT = 0.36100E + 01	
X = 0.2200E + 00	Y = 0.37635E + 01	YT = 0.37636E + 01	
X = 0.2400E + 00	Y = 0.39203E + 01	YT = 0.39204E + 01	
X = 0.2600E + 00	Y = 0.40803E + 01	YT = 0.40804E + 01	

X = 0.2800E + 00	Y = 0.42435E + 01	YT = 0.42436E + 01
X = 0.3000E + 00	Y = 0.44099E + 01	YT = 0.44100E + 01
X = 0.3200E + 00	Y = 0.45795E + 01	YT = 0.45796E + 01
X = 0.3400E + 00	Y = 0.47523E + 01	YT = 0.47524E + 01
X = 0.3600E + 00	Y = 0.49283E + 01	YT = 0.49284E + 01
X = 0.3800E + 00	Y = 0.51075E + 01	YT = 0.51076E + 01
X = 0.4000E + 00	Y = 0.52899E + 01	YT = 0.52900E + 01
X = 0.4200E + 00	Y = 0.54755E + 01	YT = 0.54756E + 01
X = 0.4400E + 00	Y = 0.56643E + 01	YT = 0.56644E + 01
X = 0.4600E + 00	Y = 0.58563E + 01	YT = 0.58564E + 01
X = 0.4800E + 00	Y = 0.60515E + 01	YT = 0.60516E + 01
X = 0.5000E + 00	Y = 0.62499E + 01	YT = 0.62500E + 01
X = 0.5200E + 00	Y = 0.69068E + 01	YT = 0.69069E + 01
X = 0.5400E + 00	Y = 0.76141E + 01	YT = 0.76142E + 01
X = 0.5600E + 00	Y = 0.83744E + 01	YT = 0.83745E + 01
X = 0.5800E + 00	Y = 0.91903E + 01	YT = 0.91904E + 01
X = 0.6000E + 00	Y = 0.10064E + 02	YT = 0.10064E + 02
X = 0.6200E + 00	Y = 0.10999E + 02	YT = 0.10999E + 02
X = 0.6400E + 00	Y = 0.11998E + 02	YT = 0.11998E + 02
X = 0.6600E + 00	Y = 0.13063E + 02	YT = 0.13063E + 02
X = 0.6800E + 00	Y = 0.14198E + 02	YT = 0.14198E + 02
X = 0.7000E + 00	Y = 0.15405E + 02	YT = 0.15405E + 02
X = 0.7200E + 00	Y = 0.16687E + 02	YT = 0.16687E + 02
X = 0.7400E + 00	Y = 0.18048E + 02	YT = 0.18048E + 02
X = 0.7600E + 00	Y = 0.19490E + 02	YT = 0.19490E + 02
X = 0.7800E + 00	Y = 0.21017E + 02	YT = 0.21017E + 02
X = 0.8000E + 00	Y = 0.22632E + 02	YT = 0.22633E + 02
X = 0.8200E + 00	Y = 0.24339E + 02	YT = 0.24339E + 02
X = 0.8400E + 00	Y = 0.26140E + 02	YT = 0.26140E + 02
X = 0.8600E + 00	Y = 0.28039E + 02	YT = 0.28040E + 02
X = 0.8800E + 00	Y = 0.30040E + 02	YT = 0.30041E + 02
X = 0.9000E + 00	Y = 0.32147E + 02	YT = 0.32147E + 02
X = 0.9200E + 00	Y = 0.34362E + 02	YT = 0.34362E + 02
X = 0.9400E + 00	Y = 0.36690E + 02	YT = 0.36690E + 02
X = 0.9600E + 00	Y = 0.39134E + 02	YT = 0.39134E + 02
X = 0.9800E + 00	Y = 0.41698E + 02	YT = 0.41698E + 02
X = 0.1000E + 01	Y = 0.44386E + 02	YT = 0.44386E + 02

ПОХИБКА РОЗВ'ЯЗКУ= 0.43386E-02

Запитання для самоперевірки

1. Який вигляд має різницева схема, побудована інтегро-інтерполяційним методом і як записуються більш загальні формули для визначення її коефіцієнтів?
2. Як можна записати похибку апроксимації різницевої схеми (4) досить гладких розв'язків вихідної задачі?
3. Як визначається похибка апроксимації граничних умов (6), (7)?
4. Як можна оцінити кількість арифметичних дій при використанні методу прогонки залежно від числа вузлів сітки? Як відбивається ця кількість від кількості арифметичних дій при застосуванні методу Гауса? 5. Що таке стійкість методу прогонки? Чи буде цей метод стійким для СЛАР (9)?
6. Як обчислюється апостеріорна оцінка похибки за допомогою методу Рунге? Як можна уточнити розв'язок задачі, знайдений на послідовності сіток?

Контрольні завдання

1. Знайти розв'язок граничної задачі (1), (2) на відрізку $[0, 1]$ з розривними коефіцієнтами $k(x)$, $q(x)$, $f(x)$, $\gamma = 0$, де $k(x) = k_i = \text{const} > 0$, $q(x) = q_i = \text{const} \geq 0$, $f(x) = f_i(x)$, $d_{i-1} < x < d_i$ ($i = \overline{1, 3}$), $d_0 = 0$, $d_3 = 1$, а

$$a_{i+1} = (1 + a_i) - \frac{d_i}{d_{i+1}}; \quad (18)$$

$$k_{i+1} = \left(\frac{d_{i+1}}{d_i}\right)^2 \left(\frac{a_i d_i + d_i}{a_{i+1} d_{i+1} + d_i}\right) k_i, \quad i = 1, 2; \quad (19)$$

$$q_i = \frac{2k_i}{d_i^2}; \quad (20)$$

$$f_i(x) = \frac{2k_i}{d_i^2} \left[1 - \left(a_i + \frac{x}{d_i} \right)^2 \right], \quad i = \overline{1, 3}, \quad (21)$$

за таких граничних умов:

$$u|_{x=0} = a_1^2, \quad u'|_{x=1} = \frac{2}{a_3 d_3 + 1} u|_{x=1}. \quad (22)$$

Точний розв'язок задачі має вигляд

$$u(x) = (a_i + x/d_i)^2, \quad d_{i-1} \leq x \leq d_i, \quad i = \overline{1, 3}.$$

Конкретний варіант її визначається нерівностями

$$0 < d_1 < d_2 < 1, \quad a_1 > 1, \quad k_1 > 0.$$

Для фіксованої кількості вузлів сітки знайти наближений розв'язок граничної задачі та оцінити його точність методом Рунге.

2. Знайти розв'язок граничної задачі (1), (2) на відрізку $[0, 1]$ при таких значеннях коефіцієнтів:

$$k(x) = \frac{\ln c(x)}{c(x)}; \quad q(x) = \frac{2((1+\gamma)x^\gamma)^2}{c(x)};$$

$$f(x) = -4a\gamma(1+\gamma)x^{\gamma-1} \ln c(x) - \frac{(a^2+1) \cdot 2((1+\gamma)x^\gamma)^2}{c(x)},$$

де $c(x) = ax^{1+\gamma} + b$, і за таких граничних умов:

$$u'|_{x=0} = \begin{cases} 2ab, & \gamma = 0, \\ 0, & \gamma = 1, 2; \end{cases} \quad u|_{x=1} = (a+b)^2. \quad (23)$$

Точний розв'язок задачі має вигляд

$$u(x) = c^2(x). \quad (24)$$

Наближений розв'язок її визначити з абсолютною похибкою ϵ .

Варіант завдання визначається викладачем відповідно до табл. 13.

3. Знайти розв'язок граничної задачі (1), (2) на відрізку $[0, 1]$ при таких значеннях коефіцієнтів:

$$k(x) = c(x); \quad q(x) = c(x); \quad f(x) = ax^2 c(x) - 2alx \sin 2(lx+d); \\ c(x) = \sin^2(lx+d) \quad (25)$$

Таблиця 13

Номер варіанта	a	b	c	γ
1	0,5	1,5	10^{-4}	0; 1; 2
2	1	2	$5 \cdot 10^{-4}$	0; 1; 2
3	1,5	2,5	10^{-3}	1; 2
4	2	3	10^{-3}	0; 1; 2
5	-1	4	10^{-4}	0
6	-4	5	$5 \cdot 10^{-5}$	0; 1; 2

Таблиця 14

Номер варіанта	a	δ_2	d	l	h
1	1	0	$\pi/6$	$\pi/3$	0,02
2	2	1	$\pi/3$	$\pi/6$	0,025
3	3	-2/3	$\pi/4$	$\pi/4$	0,02
4	0,5	1	$\pi/5$	$3\pi/10$	0,05
5	1	2	$3\pi/10$	$\pi/5$	0,04
6	2	0,05	$\pi/7$	$5\pi/14$	0,025

і за таких граничних умов:

$$u|_{x=0} = 2a; \quad -ku'|_{x=1} = \delta_2 u - 2ac(x) - 3a\delta_2|_{x=1}.$$

Точний розв'язок задачі має вигляд

$$u(x) = a(x^2 + 2). \quad (26)$$

Оцінити точність наближеного розв'язку й уточнити його, використовуючи наближення на двох послідовних сітках.

Варіант завдання визначається викладачем згідно з табл. 14.

§ 15. Метод скінченних елементів для розв'язування рівняння Пуассона

Постановка задачі. Методом скінченних елементів (МСЕ) розв'язати задачу Діріхле для рівняння Пуассона в опуклому багатокутнику Ω :

$$-\Delta u = f(x, y), \quad (x, y) \in \Omega; \quad (1)$$

$$u(x, y) = 0, \quad (x, y) \in \Gamma, \quad (2)$$

де Γ — границя області Ω .

Припустимо, що функція $f(x, y) \in C^1(\bar{\Omega})$. Вигляд її та області Ω визначається варіантом завдання.

Методичні вказівки. МСЕ є деякою модифікацією відомих методів (наприклад, методів Рітца, Бубнова—Гальоркіна), коли як координата системи застосовуються функції зі скінченними носіями (фінітні функції), які відмінні від нуля тільки на невеликій частині тієї області, в якій визначено розв'язок вихідної задачі. Такий підхід дає змогу поєднувати достоїнства відомих варіаційних і проєкційних методів, з одного боку, а з другого, формувати систему рівнянь з розрядженими матрицями (матриця буде розрядженою, якщо в ній тільки незначна кількість елементів відмінна від нуля).

▼ Побудуємо схему МСЕ для задачі (1), (2) на основі кусково-лінійних функцій. Розіб'ємо область Ω на трикутники E_m , тобто здійснимо триангуляцію області Ω так, щоб:

1) кожна пара трикутників мала або одну спільну вершину, або одну спільну сторону, або не перетиналася;

2) об'єднання трикутників складало область Ω .

Вершини трикутників триангуляції утворюють в області Ω нерегулярну сітку вузлів.

Вузли, що належать границі Γ , назвемо *граничними*, решту вузлів — *внутрішніми*. Вважатимемо, що вузли перенумеровано в деякому порядку.

Неперервні кусково-лінійні координатні функції на нерегулярній сітці будують так: кожному вузлу сітки $P_k = (x_k, y_k)$ ставимо у відповідність неперервну кусково-лінійну функцію $\varphi_k(x, y)$, яка дорівнює одиниці у вузлі (x_k, y_k) і нулю в решті вузлів. Функція $\varphi_k(x, y)$ відмінна від нуля тільки в полі трикутників, які примикають до вузла (x_k, y_k) .

Розглянемо трикутник з вершинами (P_i, P_j, P_k) . В полі цього трикутника функція $\varphi_i(x, y)$ має вигляд

$$\varphi_i(x, y) = \frac{1}{2A} (\alpha_i x + \beta_i y + \gamma_i), \quad (3)$$

де $\alpha_i = y_j - y_k$; $\beta_i = x_k - x_j$; $\gamma_i = x_j y_k - x_k y_j$;
 A — площа трикутника, що обчислюється так:

$$A = \frac{1}{2} \det \begin{pmatrix} 1 & x_i & y_i \\ 1 & y_j & y_j \\ 1 & x_k & y_k \end{pmatrix}.$$

Відзначимо, що в цьому трикутнику функції $\varphi_j(x, y)$ і $\varphi_k(x, y)$ впливають з $\varphi_i(x, y)$ циклічним переставленням індексів $i \rightarrow j \rightarrow k \rightarrow i$ при α, β, γ, x і y . Наприклад, $\alpha_j = y_k - y_i$, $\beta_j = x_i - x_k$, $\gamma_j = x_k y_i - x_i y_k$ і т. д.

Використовуючи (3), можна визначити функцію $\varphi_i(x, y)$ в полі будь-якого іншого трикутника, де вона буде відмінною від нуля. Внаслідок довільності індекса i всі координатні функції вважаємо заданими аналітично.

Позначимо через \dot{H}^1 підпростір простору $\dot{W}_2^1(\Omega)$ неперервних кусково-лінійних функцій, які дорівнюють нулю в граничних вузлах. Простір $\dot{W}_2^1(\Omega)$ є замиканням множини функцій $u \in C^\infty(\Omega)$ за нормою

$$\|u\|_{\dot{W}_2^1(\Omega)} = \left\{ \int_{\Omega} \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 + u^2 \right] dx dy \right\}^{1/2},$$

де $C^\infty(\Omega)$ — сукупність нескінченно диференційовних функцій з компактним носієм в Ω (носієм неперервної функції $f(x, y)$ називається замиканням множини тих точок, де $f(x, y) \neq 0$).

Наближений розв'язок задачі (1), (2) відшукуємо у вигляді

$$v(x, y) = \sum_{k=1}^{N_0} v_k \varphi_k(x, y),$$

де $v_k = v(x_k, y_k)$, а N_0 — кількість внутрішніх вузлів.

Схему МСЕ для задачі (1), (2) будуватимемо на основі методу Бубнова — Гальборкіна. Для цього в інтегральну тотожність, яка використовується для визначення узагальненого розв'язку розглядуваної задачі,

$$a(u, \eta) = \int_{\Omega} f \eta dx dy, \quad \forall \eta \in \bar{W}_2^1(\Omega);$$

$$a(u, \eta) = \int_{\Omega} \left[\frac{\partial u}{\partial x} \frac{\partial \eta}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial \eta}{\partial y} \right] dx dy$$

замість $u(x, y)$ підставимо функцію $v(x, y)$, а замість функції $\eta(x, y)$ — набір функцій $\varphi_i(x, y)$, $i = \overline{1, N_0}$. Тоді шукана схема МСЕ для визначення невідомих v_k матиме вигляд

$$a(v, \varphi_i) = \int_{\Omega} f \varphi_i dx dy, \quad i = \overline{1, N_0}. \quad (4)$$

ТЕОРЕМА [13, с. 186, 189]. Наближений розв'язок $v(x, y)$ при $h \rightarrow 0$ збігається до розв'язку $u(x, y)$ задачі (1), (2), причому справедливою є оцінка швидкості збіжності

$$\|u - v\|_{W_2^1(\Omega)} \leq \frac{Mh}{\sin \theta_0} \|u\|_{W_2^2(\Omega)},$$

де M — стала, яка не залежить від h , θ_0 , u . Тут h — найбільша зі сторін трикутників E_i ; $\theta_0 > 0$ — найменший із кутів у всіх трикутниках E_i , на які розбито область Ω ;

$$\|u\|_{W_2^2(\Omega)} = \left\{ \|u\|_{W_2^1(\Omega)}^2 + \int_{\Omega} \left[\left(\frac{\partial^2 u}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 u}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 u}{\partial y^2} \right)^2 \right] dx dy \right\}^{1/2},$$

а $W_2^2(\Omega)$ — замикання множини функцій $u \in C^2(\bar{\Omega})$ за нормою $\|u\|_{W_2^2(\Omega)}$.

Для обчислення інтегралів у правій частині (4) функцію $u \in C^2(\bar{\Omega})$ замінимо інтерполюючим поліномом

$$f_I(x, y) = \sum_{i=0}^N f(P_i) \varphi_i(x, y),$$

де N — кількість вузлів нерегулярної сітки, а потім точно проінтегруємо поліноми.

Оскільки $f(x, y) \in C^1(\bar{\Omega})$, маємо $\|f - f_I\|_{L_2(\Omega)} \leq Mh \|f\|_{C^1(\bar{\Omega})}$ і для схеми МСЕ

$$a(\tilde{v}, \varphi_k) \int_{\Omega} f_I \varphi_k dx dy, \quad k = \overline{1, N_0}, \quad \tilde{v} = \sum_{j=1}^{N_0} \tilde{v}(P_j) \varphi_j(x, y) \quad (5)$$

справджується сформульована вище теорема.

Зазначимо, що якщо значення функції $f(x, y)$ у вершинах кожного трикутника E_m замінити її значенням $f(\bar{x}^{(m)}, \bar{y}^{(m)})$ у центрі його ваги, де

$$\bar{x}^{(m)} = \frac{1}{3}(x_i + x_j + x_k); \quad \bar{y}^{(m)} = \frac{1}{3}(y_i + y_j + y_k), \quad (6)$$

а точки $P_i = (x_i, \varphi_i)$, $P_j = (x_j, \varphi_j)$, $P_k = (x_k, \varphi_k)$ є вершинами трикутника E_m , то точність схеми МСЕ (5) не зміниться.

▼ Зупинимося на інженерному підході в МСЕ, який є корисним для розуміння як суті цього методу, так і його алгоритмічних можливостей. Знову розглянемо задачу (1), (2). Схему МСЕ будуватимемо тепер на основі методу Рітца. Задача (1), (2) при цьому буде еквівалентною задачі відшукування елемента $u(x, y) \in \overset{\circ}{W}_2^1(\Omega)$, який мінімізує функціонал енергії

$$F(u) = \int_{\Omega} \left[\left(\frac{\partial u}{\partial x} \right)^2 + \left(\frac{\partial u}{\partial y} \right)^2 \right] dx dy - 2 \int_{\Omega} f u dx dy. \quad (7)$$

Здійснимо триангуляцію області Ω так, як описано вище, і вважатимемо, що в полі кожного трикутника E_m наближений розв'язок задачі є лінійною функцією. Тим самим для визначення цього розв'язку досить знати його значення у вершинах трикутника E_m , які позначимо через P_i, P_j, P_k .

Нехай наближеним розв'язком розглядуваної задачі є $v(x, y)$, а його значенням у вершинах трикутника E_m — значення $v_i = v(P_i)$. Очевидно, в цьому трикутнику

$$v(x, y) = v_i \varphi_i^{(m)}(x, y) + v_j \varphi_j^{(m)}(x, y) + v_k \varphi_k^{(m)}(x, y), \quad (x, y) \in E_m. \quad (8)$$

де верхній індекс указує номер трикутника, функція $\varphi_i^{(m)}(x, y)$ обчислюється за формулою (3), а функції $\varphi_j^{(m)}(x, y)$, $\varphi_k^{(m)}(x, y)$ впливають з $\varphi_i^{(m)}(x, y)$ циклічним переставленням індексів $i \rightarrow j \rightarrow k \rightarrow i$ при α, β, γ, x і y . Функції $\varphi_i^{(m)}, \varphi_j^{(m)}, \varphi_k^{(m)}$ — це функції форми скінченного елемента (в даному разі — трикутника).

Нехай

$$\vec{W}^{(m)} = [v_i \quad v_j \quad v_k]^T; \quad \vec{\Phi}^{(m)} = [\varphi_i^{(m)} \quad \varphi_j^{(m)} \quad \varphi_k^{(m)}].$$

Тоді співвідношення (8) матиме вигляд

$$v(x, y) = \vec{\Phi}^{(m)} \vec{W}^{(m)}, \quad (x, y) \in E_m. \quad (9)$$

У полі кожного трикутника E_m функцію $f(x, y)$ запишемо так:

$$f(x, y) = f_m \varphi_i^{(m)} + f_m \varphi_j^{(m)} + f_m \varphi_k^{(m)}, \quad (10)$$

де f_m — значення цієї функції в центрі ваги трикутника E_m , координати якого визначаються виразами (6).

Співвідношення (10) перепишемо у вигляді

$$f(x, y) = \vec{\Phi}^{(m)} f_m \vec{e}, \quad (x, y) \in E_m, \quad (11)$$

де $\vec{e} = [1 \ 1 \ 1]^T$.

Підставимо наближений розв'язок $v(x, y)$ у функціонал (7). Для цього спочатку подамо його як суму функціоналів, кожний з яких на своєму елементі визначається так:

$$F(v) = \sum_{m=1}^M F^{(m)}(v), \quad (12)$$

де M — кількість елементів; $F^{(m)}(v) = F_1^{(m)}(v) + F_2^{(m)}(v) = \int_{E_m} \left[\left(\frac{\partial v}{\partial x} \right)^2 + \left(\frac{\partial v}{\partial y} \right)^2 \right] dx dy - 2 \int_{E_m} f v dx dy$.

Підставивши (9) у $F_1^{(m)}(v)$ і врахувавши те, що $\vec{\Phi}^{(m)} \vec{W}^{(m)} = \vec{W}^{(m)T} \vec{\Phi}^{(m)T}$, а

$$\vec{B}^{(m)} = \frac{\partial}{\partial x} \vec{\Phi}^{(m)} = \begin{bmatrix} \alpha_i & \alpha_j & \alpha_k \\ 2A & 2A & 2A \end{bmatrix};$$

$$\vec{D}^{(m)} = \frac{\partial}{\partial y} \vec{\Phi}^{(m)} = \begin{bmatrix} \beta_i & \beta_j & \beta_k \\ 2A & 2A & 2A \end{bmatrix},$$

дістанемо

$$F_1^{(m)}(v) = \int_{E_m} [\vec{W}^{(m)T} \vec{B}^{(m)T} \vec{B}^{(m)} \vec{W}^{(m)} + \vec{W}^{(m)T} \vec{D}^{(m)T} \vec{D}^{(m)} \vec{W}^{(m)}] dx dy = \vec{W}^{(m)T} S^{(m)} \vec{W}^{(m)}, \quad (13)$$

де

$$S^{(m)} = \int_{E_m} [\vec{B}^{(m)T} \vec{B}^{(m)} + \vec{D}^{(m)T} \vec{D}^{(m)}] dx dy = \int_{E_m} \left[\frac{1}{4A^2} \begin{pmatrix} \alpha_i^2 & \alpha_i \alpha_j & \alpha_i \alpha_k \\ \alpha_j \alpha_i & \alpha_j^2 & \alpha_j \alpha_k \\ \alpha_k & \alpha_k \alpha_j & \alpha_k^2 \end{pmatrix} + \frac{1}{4A^2} \begin{pmatrix} \beta_i^2 & \beta_i \beta_j & \beta_i \beta_k \\ \beta_j \beta_i & \beta_j^2 & \beta_j \beta_k \\ \beta_k \beta_j & \beta_k \beta_j & \beta_k^2 \end{pmatrix} \right] dx dy = \frac{1}{4A} \begin{pmatrix} \alpha_i^2 + \beta_i^2 & \alpha_i \alpha_j + \beta_i \beta_j & \alpha_i \alpha_k + \beta_i \beta_k \\ \alpha_j \alpha_i + \beta_j \beta_i & \alpha_j^2 + \beta_j^2 & \alpha_j \alpha_k + \beta_j \beta_k \\ \alpha_k \alpha_i + \beta_k \beta_i & \alpha_k \alpha_j + \beta_k \beta_j & \alpha_k^2 + \beta_k^2 \end{pmatrix} \quad (14)$$

— матриця жорсткості елемента.

З урахуванням (9) і (11) матимемо

$$F_2^{(m)}(v) = -2 \int_{E_m} \vec{\Phi}^{(m)} f_m \vec{e} \vec{\Phi}^{(m)T} \vec{W}^{(m)} dx dy =$$

$$= -2 \int_{E_m} \vec{W}^{(m)T} \vec{\Phi}^{(m)T} \vec{\Phi}^{(m)} f_m \vec{e} dx dy = -2 \vec{W}^{(m)T} T^{(m)} f_m \vec{e}, \quad (15)$$

де

$$T^{(m)} = \int_{E_m} \vec{\Phi}^{(m)T} \vec{\Phi}^{(m)} dx dy =$$

$$= \begin{bmatrix} \int_{E_m} (\varphi_i^{(m)})^2 dx dy & \int_{E_m} \varphi_i^{(m)} \varphi_j^{(m)} dx dy & \int_{E_m} \varphi_i^{(m)} \varphi_k^{(m)} dx dy \\ \int_{E_m} \varphi_j^{(m)} \varphi_i^{(m)} dx dy & \int_{E_m} (\varphi_j^{(m)})^2 dx dy & \int_{E_m} \varphi_j^{(m)} \varphi_k^{(m)} dx dy \\ \int_{E_m} \varphi_k^{(m)} \varphi_i^{(m)} dx dy & \int_{E_m} \varphi_k^{(m)} \varphi_j^{(m)} dx dy & \int_{E_m} (\varphi_k^{(m)})^2 dx dy \end{bmatrix} \quad (16)$$

— матриця маси елемента.

Вектор $\vec{f}^{(m)} = T^{(m)} f_m \vec{e}$ називається *вектором вузлових сил елемента*, чи *вектором навантаження*.

Отже, враховуючи (13) і (15), функціонал

$$F^{(m)}(v) \vec{W}^{(m)T} S^{(m)} \vec{W}^{(m)} - 2 \vec{W}^{(m)T} \vec{f}^{(m)} \quad (17)$$

і є квадратичною формою вузлових значень наближеного розв'язку розглядуваної задачі.

Підставимо (17) у (12). Але спочатку встановимо зв'язок між повним вектором невідомих $V = [v_0 v_1 \dots v_N]^T$ і вектором невідомих $\vec{W}^{(m)}$ на елементі E_m . Маємо

$$\vec{W}^{(m)} = C^{(m)} \vec{V}, \quad (18)$$

де $C^{(m)}$ — прямокутна матриця розмірністю $3 \times (N + 1)$, яка визначається так:

$$C^{(m)} = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 1 & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 1 & \dots & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & \dots & 1 & \dots & 0 \end{bmatrix}$$

і називається *матрицею кінематичних зв'язків*.

З урахуванням (18) співвідношення (17) набуває вигляду

$$F^{(m)}(v) = \vec{V}^T C^{(m)T} S^{(m)} C^{(m)} \vec{V} - 2 \vec{V}^T C^{(m)T} \vec{f}^{(m)},$$

а (12) — вигляду

$$F(v) = \vec{V}^T S \vec{V} - 2\vec{V}^T \vec{F}, \quad (19)$$

де

$$S = \sum_{m=1}^M C^{(m)T} S^{(m)} C^{(m)} \quad (20)$$

— глобальна матриця жорсткості, а

$$\vec{F} = \sum_{m=1}^M C^{(m)T} \vec{f}^{(m)} \quad (21)$$

— глобальний вектор навантаження.

Із (19) випливає, що значення функціонала F наближеного розв'язку $v(x, y)$ є функцією $N + 1$ змінних v_0, \dots, v_N , умовою мінімуму якої є обернення в нуль її перших похідних.

Виконавши диференціювання і прирівнявши знайдені вирази до нуля, з урахуванням симетрії матриці S (яку легко перевірити) і після скорочення на множник 2, дістанемо систему рівнянь

$$S\vec{V} = \vec{F}. \quad (22)$$

Зазначимо, що (22) — це ще не та система, розв'язок якої є наближеним розв'язком розглядуваної задачі, оскільки не враховано умову Діріхле. Розглянемо метод, який дає змогу зберегти симетрію матриці жорсткості при врахуванні граничних умов. Його можна сформулювати у вигляді такого правила [15, с. 90]: якщо у вузлі P_i задано умову Діріхле $v(P_i) = 0$, то f_i (i -та компонента вектора \vec{F}) замінюється нулем, S_{ii} — одиницею, а решта елементів i -го рядка та i -го стовпця матриці S — нулями.

Слід відзначити, що стосовно крайової задачі з умовою Діріхле сформульоване вище правило може бути використане для зменшення порядку системи рівнянь (22). Наприклад, якщо у вузлі i задано умову Діріхле, то в матриці S викреслюються i -й рядок і i -й стовець, а у векторі \vec{F} — i -та компонента. У вузлі P_i функція $v(x, y)$ задається рівністю $v(P_i) = 0$. В результаті кількість невідомих зменшується на одиницю.

Таким чином, якщо число граничних вузлів дорівнює N_1 , то порядок системи рівнянь (22) може бути зменшений до $N + 1 - N_1$.

Зауважимо, що при реальних обчисленнях глобальних матриць жорсткості S і вектора навантаження \vec{F} зовсім не обов'язково виконувати перемноження матриць і множення матриць на вектори за формулами (20) і (21). Це пов'язано з простою структурою матриць $C^{(m)}$. Проілюструємо процес обчислень S і \vec{F} конкретним прикладом.

Для простоти припустимо, що існуючий ансамбль скінченних елементів складається з двох трикутників E_1 і E_2 , які мають вершини (1, 2, 3), (2, 3, 4). Тоді

$$\begin{aligned}
 C^{(1)T} S^{(1)} C^{(1)} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} S_{12}^{(1)} & S_{12}^{(1)} & S_{13}^{(1)} \\ S_{21}^{(1)} & S_{22}^{(1)} & S_{23}^{(1)} \\ S_{31}^{(1)} & S_{32}^{(1)} & S_{33}^{(1)} \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} = \\
 &= \begin{pmatrix} S_{11}^{(1)} & S_{12}^{(1)} & S_{13}^{(1)} & 0 \\ S_{21}^{(1)} & S_{22}^{(1)} & S_{23}^{(1)} & 0 \\ S_{31}^{(1)} & S_{32}^{(1)} & S_{33}^{(1)} & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}; \\
 C^{(2)T} S^{(2)} C^{(2)} &= \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} S_{22}^{(2)} & S_{23}^{(2)} & S_{24}^{(2)} \\ S_{32}^{(2)} & S_{33}^{(2)} & S_{34}^{(2)} \\ S_{42}^{(2)} & S_{43}^{(2)} & S_{44}^{(2)} \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \\
 &= \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & S_{22}^{(2)} & S_{23}^{(2)} & S_{24}^{(2)} \\ 0 & S_{32}^{(2)} & S_{33}^{(2)} & S_{34}^{(2)} \\ 0 & S_{42}^{(2)} & S_{43}^{(2)} & S_{44}^{(2)} \end{pmatrix}; \\
 \sum_{m=1}^2 C^{(m)T} S^{(m)} C^{(m)} &= \begin{pmatrix} S_{11}^{(1)} & S_{12}^{(1)} & S_{13}^{(1)} & 0 \\ S_{21}^{(1)} & S_{22}^{(1)} + S_{22}^{(2)} & S_{23}^{(1)} + S_{23}^{(2)} & S_{24}^{(2)} \\ S_{31}^{(1)} & S_{32}^{(1)} + S_{32}^{(2)} & S_{33}^{(1)} + S_{33}^{(2)} & S_{34}^{(2)} \\ 0 & S_{42}^{(2)} & S_{43}^{(2)} & S_{44}^{(2)} \end{pmatrix}. \quad (23)
 \end{aligned}$$

Із (23) випливає, що в загальному випадку глобальна матриця жорсткості S може бути знайдена обчисленням матриці жорсткості окремого елемента з наступним додаванням дев'яти ненульових матричних внесків, обумовлених цим елементом, до відповідних елементів матриці S . Цей підхід, у свою чергу, передбачає дуже компактний і зручний метод збереження топологічної інформації, що міститься в матриці $C^{(m)}$, за допомогою масива, який визначає вершини кожного трикутника через вузлові номери їх.

Розглянемо тепер, як обчислюється вектор навантаження \vec{F} . Маємо

$$C^{(1)T} \vec{f}^{(1)} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} f_1^{(1)} \\ f_2^{(1)} \\ f_3^{(1)} \end{pmatrix} = \begin{pmatrix} f_1^{(1)} \\ f_2^{(1)} \\ f_3^{(1)} \\ 0 \end{pmatrix},$$


```

C  НОМЕР ПРИСТРОЮ ВВЕДЕННЯ ВИХІДНОЇ ІНФОРМАЦІЇ
   INPUT=3
C  НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ РЕЗУЛЬТАТІВ
   KONSOL=2
C  ПІДПРОГРАМА МФЕ ОБЧИСЛЮЄ НАБЛИЖЕНИЙ РОЗВ'ЯЗОК МСЕ
C  ЗАДАЧІ ДІРІХЛЕ ДЛЯ РІВНЯННЯ ПУАССОНА В ОПУКЛОМУ
C  МНОГОКУТНИКУ
   CALL MFE(F)
   STOP
   END

      FUNCTION F(X,Y)
C  ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯННЯ ПУАССОНА
   F=2.*(X+Y)
   RETURN
   END

```

```

1 0.0 0.0
2 0.2 0.0
3 0.4 0.0
4 0.6 0.0
5 0.8 0.0
6 1.0 0.0
7 0.0 0.2
8 0.2 0.2
9 0.4 0.2
10 0.6 0.2
11 0.8 0.2
12 0.0 0.4
13 0.2 0.4
14 0.4 0.4
15 0.6 0.4
16 0.0 0.6
17 0.2 0.6
18 0.4 0.6
19 0.0 0.8
20 0.2 0.8
21 0.0 1.0
0 0.0 0.0
1 2 7
8 7 2
2 3 8
9 8 3
3 4 9
10 9 4
4 5 10
11 10 5
5 6 11
7 8 12
13 12 8
8 9 13
14 13 9
9 10 14
15 14 10
10 11 15
12 13 16

```


17 16 13
13 14 17
18 17 14
14 15 18
16 17 19
20 19 17
17 18 20
19 20 21
0 0 0

1
2
3
4
5
6
7
11
12
15
16
18
19
20
21
0

Вихідна інформація записується у файл вихідних даних з номером INPUT.
Обчислення на ЕОМ за цією програмою дають такий результат:

РОЗВ'ЯЗОК ЗАДАЧІ

ВУЗОЛ	X	Y	U
1	0.00000	0.00000	0.00000
2	0.20000	0.00000	0.00000
3	0.40000	0.00000	0.00000
4	0.60000	0.00000	0.00000
5	0.80000	0.00000	0.00000
6	1.00000	0.00000	0.00000
7	0.00000	0.20000	0.00000
8	0.20000	0.20000	0.02400
9	0.40000	0.20000	0.03200
10	0.60000	0.20000	0.02400
11	0.80000	0.20000	0.00000
12	0.00000	0.40000	0.00000
13	0.20000	0.40000	0.03200
14	0.40000	0.40000	0.03200
15	0.60000	0.40000	0.00000
16	0.00000	0.60000	0.00000
17	0.20000	0.60000	0.02400
18	0.40000	0.60000	0.00000
19	0.00000	0.80000	0.00000
20	0.20000	0.80000	0.00000
21	0.00000	1.00000	0.00000

Запитання для самоперевірки

1. Які функції називаються функціями зі скінченними носіями? 2. Яка матриця називається розрідженою? 3. Як розбивається на трикутники многокутна область? 4. Як будуються кусково-лінійні координатні функції на нерегулярній сітці? 5. Як формулюється теорема про оцінку швидкості збіжності схеми МСЕ, побудованої на основі кусково-лінійних координатних функцій, стосовно задачі Діріхле для рівняння Пуассона в опуклому многокутнику? 6. Як будується глобальна матриця жорсткості з матриці жорсткості скінченних елементів? 7. Як будується глобальний вектор навантаження з векторів навантаження скінченних елементів?

Контрольні завдання

1. За допомогою МСЕ знайти розв'язок задачі (1), (2), якщо $f(x, y) = \alpha x + \beta y$, $\alpha, \beta = \overline{1, 5}$, а областю $\bar{\Omega}$ є одиничний квадрат, причому

$$\bar{\Omega} = \{0 \leq x \leq 1, 0 \leq y \leq 1\}, \quad f(x, y) = 2[y(1-y) + x(1-x)].$$

Параметри α, β і кількість трикутників задаються викладачем.

2. Розв'язати попередню задачу, якщо областю $\bar{\Omega}$ є прямокутник, причому

$$\bar{\Omega} = \{0 \leq x \leq 1/2, 0 \leq y \leq 1\}, \quad f(x, y) = 2.$$

3. Розв'язати задачу 1, якщо областю $\bar{\Omega}$ є трикутник, причому

$$\bar{\Omega} = \{x \geq 0, y \geq 0, x + y \leq 1\}.$$

§ 16. Розв'язування одновимірного рівняння теплопровідності

Постановка задачі. В області $\bar{Q}_T = \{a \leq x \leq b, 0 \leq t \leq T\}$ знайти розв'язок одновимірного нестационарного рівняння теплопровідності

$$\frac{du}{dt} = \frac{1}{x^m} \frac{\partial}{\partial x} \left(x^m k(x, t) \frac{\partial u}{\partial x} \right) - q(x, t)u + f(x, t), \quad x \in (a, b), t > 0, \quad (1)$$

яке задовольняє початкові умови

$$u(x, 0) = u_0(x), \quad x \in [a, b] \quad (2)$$

і крайові умови

$$\begin{aligned} \alpha_1 k(a, t) \frac{\partial u(a, t)}{\partial x} &= \beta_1 u(a, t) - \mu_1(t); \\ -\alpha_2 k(b, t) \frac{\partial u(b, t)}{\partial x} &= \beta_2 u(b, t) - \mu_2(t), \end{aligned} \quad (3)$$

де $k(x, t), q(x, t), f(x, t), u_0(x), \mu_1(t), \mu_2(t)$ задані функції; α_k, β_k ($k = 1, 2$) — задані невід'ємні сталі, причому виконуються нерівності $0 < k_0 \leq k(x, t) \leq k_1, q(x, t) \geq 0, \alpha_k^2 + \beta_k^2 \neq 0, k = 1, 2$ (k_0, k_1 — деякі сталі).

Методичні вказівки. Розглянемо різницеві методи розв'язування задачі (1) — (3). В області \bar{Q}_T введемо сітку $\bar{\omega}_{h,\tau} = \bar{\omega}_h \times \bar{\omega}_\tau$, де $\bar{\omega}_h = \{x_i = a + ih, h = (b - a)/N, i = 0, 1, \dots, N\}$; $\bar{\omega}_\tau = \{t_j = j\tau, \tau = T/M, j = 0, 1, \dots, M\}$. Позначимо $y_{ij} = y(x_i, t_j)$.

За допомогою інтегро-інтерполяційного методу апроксимуємо задачу (1) — (3) різницевою схемою з ваговими коефіцієнтами [22, с. 378—381, с. 398—405]:

$$\begin{aligned} \tilde{x}_i^m y_{i,j}^l &= \sigma (\tilde{p} y_x^{l+1})_{x,i} - \tilde{\sigma} \tilde{x}_i^m \tilde{q}_i y_i^{l+1} + (1 - \sigma) (\tilde{r} y_x^l)_{x,i} - \\ &\quad - (1 - \sigma) \tilde{x}_i^m \tilde{q}_i y_i^l + \tilde{x}_i^m \tilde{r}_i, \end{aligned} \quad (4)$$

$$i = 1, 2, \dots, N - 1, j = 1, 2, \dots, M, y_i^0 = u_0(x_i),$$

$$i = 0, 1, \dots, N;$$

$$\begin{aligned} \sigma \alpha_1 \tilde{p}_1 y_{x,1}^{l+1} + (1 - \sigma) \alpha_1 \tilde{p}_1 y_{x,i}^l &= x_0^m \beta_1 \sigma y_0^{l+1} + (1 - \sigma) \beta_1 x_0^m y_0^l - x_0^m \bar{\mu}_1 + \\ &\quad + \frac{h}{2} \alpha_1 \tilde{x}_0^m y_{t,0}^l - \frac{h}{2} \alpha_1 \tilde{x}_0^m (\tilde{f}_0 - \sigma \bar{q}_0 y_0^{l+1} + (1 - \sigma) \bar{q}_0 y_0^l); \end{aligned} \quad (5)$$

$$\begin{aligned} -\sigma \alpha_2 \tilde{p}_N y_{x,N}^{l+1} - (1 - \sigma) \alpha_2 \tilde{p}_N y_{x,N}^l &= \sigma x_N^m \beta_2 y_N^{l+1} + (1 - \sigma) \beta_2 x_N^m y_N^l - \\ - x_N^m \bar{\mu}_2 + \frac{h}{2} \alpha_2 \tilde{x}_N^m y_{t,N}^l - \frac{h}{2} \alpha_2 \tilde{x}_N^m (\tilde{f}_N - \sigma \bar{q}_N y_N^{l+1} - (1 - \sigma) \bar{q}_N y_N^l), \end{aligned} \quad (6)$$

де

$$\tilde{x}_0^m = h^{-1} \int_0^{x_1} x^m dx; \quad \tilde{x}_N^m = h^{-1} \int_{x_{N-1}}^{x_N} x^m dx;$$

$$\tilde{x}_i^m = (2h)^{-1} \int_{x_{i-1}}^{x_{i+1}} x^m dx, \quad i = 2, 3, \dots, N - 1;$$

$$\tilde{p}_i = x_{i-1/2}^m \bar{k}_{i-1/2}, \quad i = 1, 2, \dots, N; \quad \bar{S}_i = s_i^{l+\sigma} = s(x_i, t_j + \sigma\tau),$$

$$i = 0, 1, \dots, N; \quad j = 0, 1, \dots, M - 1.$$

Покладаючи в (4) — (6) $\sigma = 0$, дістаємо явну схему; при $\sigma = 1$ — схему з випередженням (повністю неявну схему); при $\sigma = 0,5$ — симетричну схему Кранка — Ніколсона, яка записується на шаблоні з шести вузлів.

У разі досить гладких вихідних даних різницева схема (4) — (6) стійка при $\sigma \geq 0,5$ і має місце рівномірна збіжність її зі швидкістю $O(h^2 + \tau^{m\sigma})$, де

$$m_\sigma = \begin{cases} 2 & \text{при } \sigma = 0,5; \\ 1 & \text{при } \sigma \neq 0,5. \end{cases}$$

Різницева схема (4) — (6) при $\sigma \neq 0$ буде неявною, тому y_i^{j+1} знаходиться як розв'язок СЛАР з тридіагональною матрицею, а саме:

$$\begin{aligned} c_1 v_1 + b_1 v_2 &= \varphi_1; \\ d_i v_{i-1} + c_i v_i + b_i v_{i+1} &= \varphi_i, \quad i = 2, 3, \dots, N; \\ d_{N+1} v_N + c_{N+1} v_{N+1} &= \varphi_{N+1}, \end{aligned} \quad (7)$$

де

$$\begin{aligned} b_1 &= \sigma \frac{\tau}{h^2} \alpha_1 \tilde{p}_1; \quad c_1 = -\sigma \frac{\tau}{h} \beta_1 x_0^m - \frac{\alpha_1}{2} \tilde{x}_0^m - \\ &\quad - \sigma \frac{\tau}{2} \alpha_1 \tilde{x}_0^m \tilde{q}_0 - b_1; \\ \varphi_1 &= (1 - \sigma) \frac{\tau}{h} \beta_1 x_0^m y_0^j - \frac{\tau}{h} x_0^m \mu_2 - \frac{\alpha_1}{2} \tilde{x}_0^m y_0^j - \\ &\quad - \frac{\tau}{2} \alpha_1 \tilde{x}_0^m \tilde{f}_0 + (1 - \sigma) \frac{\tau}{2} \alpha_1 \tilde{x}_0^m \tilde{q}_0 y_0^j - (1 - \sigma) \frac{\tau}{h^2} \alpha_1 \tilde{p}_1 (y_1^j - y_0^j); \\ d_i &= \frac{\sigma \tau}{h^2} \tilde{p}_i; \quad b_i = \frac{\sigma \tau}{h^2} \tilde{p}_{i+1}; \quad c_i = -\tilde{x}_i^m - \tau \sigma \tilde{x}_i^m \tilde{q}_i - (d_i + b_i); \\ \varphi_i &= -\tilde{x}_i^m y_i^j - \frac{\tau(1 - \sigma)}{h^2} (\tilde{p}_{i+1} (y_{i+1}^j - y_i^j) - \tilde{p}_i (y_i^j - y_{i-1}^j)) + \\ &\quad + \tau(1 - \sigma) \tilde{x}_i^m \tilde{q}_i y_i^j - \tau \tilde{f}_i \tilde{x}_i^m, \quad i = 2, 3, \dots, N; \\ d_{N+1} &= \sigma \frac{\tau}{h^2} \alpha_2 \tilde{p}_N; \quad c_{N+1} = -\sigma \frac{\tau}{h} \beta_2 x_N^m - \frac{\alpha_2}{2} x_N^m - \sigma \frac{\tau}{2} \alpha_2 \tilde{x}_N^m \tilde{q}_N - d_{N+1}; \\ \varphi_{N+1} &= (1 - \sigma) \frac{\tau}{h} \beta_2 x_N y_N^j - \frac{\tau}{h} x_N^m \mu_2 - \frac{\alpha_2}{2} \tilde{x}_N^m y_N^j - \frac{\tau}{2} \tilde{x}_N^m \tilde{f}_N + \\ &\quad + (1 - \sigma) \frac{\tau}{2} \alpha_2 \tilde{x}_N^m \tilde{q}_N y_N^j + (1 - \sigma) \frac{\tau}{h^2} \alpha_2 \tilde{p}_N (y_N^j - y_{N-1}^j). \end{aligned} \quad (8)$$

Таким чином, розв'язавши СЛАР (7), знайдемо значення $y_i^{j+1} = v_{i+1}$ ($i = 0, 1, \dots, N$), якщо відомо розв'язок y_i^j на j -му ярусі (на нульовому ярусі розв'язок задається виразом (5)).

Система лінійних алгебраїчних рівнянь (7) розв'язується методом прогонки. Обчислювальна схема цього методу зводиться до виконання таких дій:

а) визначення коефіцієнтів m_i , w_i за формулами

$$\begin{aligned} m_2 &= -b_1/c_1, \quad w_2 = \varphi_1/c_1, \quad c_1 \neq 0; \\ m_{i+1} &= -b_i/(c_i + d_i m_i), \quad w_{i+1} = (\varphi_i - d_i w_i)/(c_i + d_i w_i), \\ &\quad i = 2, 3, \dots, N; \end{aligned}$$

б) обчислення v_{N+1} за формулою

$$v_{N+1} = (\varphi_{N+1} - d_{N+1}w_{N+1}) / (c_{N+1} + d_{N+1}m_{N+1});$$

в) визначення v_i за формулою

$$v_{i-1} = m_i v_i + w_i, \quad i = N + 1, N, \dots, 2.$$

З (8) випливає, що умова стійкості методу прогонки $|c_i| \geq |b_i| + |d_i|$ виконується.

Задача (1) — (3) є математичною моделлю різних нестационарних процесів, наприклад теплопровідності, дифузії та ін. Так, процес поширення тепла в тілі може бути описаний диференціальним рівнянням

$$c\rho \frac{\partial u}{\partial t} = \frac{1}{x^m} \frac{\partial}{\partial x} \left(x^m \lambda \frac{\partial u}{\partial x} \right) - d(u - u_{cp}) + F, \quad x \in (a, b), t > 0, \quad (9)$$

яке задовольняє такі початкові та крайові умови:

$$u(x, 0) = u_0(x), \quad x \in [a, b]; \quad (10)$$

$$\alpha_1 \lambda \frac{\partial u}{\partial x} = \gamma_1 (u - u_{cp}) \quad \text{при } x = a;$$

$$-\alpha_2 \lambda \frac{\partial u}{\partial x} = \gamma_2 (u - u_{cp}) \quad \text{при } x = b. \quad (11)$$

Тут c — питома теплоємність; ρ — щільність; λ — коефіцієнт теплопровідності; d — коефіцієнт теплообміну на поверхні тіла; F — щільність джерел тепла; u_{cp} — температура навколишнього середовища; u_0 — початковий розподіл температури; γ_k ($k = 1, 2$) — коефіцієнт тепловіддачі на границі; α_k ($k = 1, 2$) — деякі сталі величини, які дорівнюють нулю чи одиниці.

Показник m може дорівнювати 0, 1 або 2, що відповідає запису рівняння в декартових, циліндричних або сферичних координатах. Якщо величини c та ρ сталі, то задачу (9) — (11) можна записати у вигляді (1) — (3), де $k = \lambda / (c\rho)$, $q = d / (c\rho)$, $f = qu_{cp} + F / (c\rho)$, $\beta_k = \gamma_k / (c\rho)$, $\mu_k = \gamma_k u_{cp} / (c\rho)$.

Співвідношення (3) залежно від значень параметрів α_k , β_k ($k = 1, 2$) визначають різні фізичні умови на границі: а) випадок $\alpha_k = 0$, $\beta_k > 0$ означає, що на границі задано температуру тіла (*крайові умови першого роду*); б) випадок $\alpha_k > 0$, $\beta_k = 0$ свідчить про те, що на границі задано тепловий потік (*крайові умови другого роду*); в) випадок $\alpha_k > 0$, $\beta_k > 0$ означає, що на границі задано теплообмін з навколишнім середовищем (*крайові умови третього роду*).

Зауважимо, що коли рівняння (1) розглядається в циліндричних або сферичних координатах ($m = 1$ або $m = 2$) і $a = 0$, то в точці $x = 0$ має виконуватись умова обмеженості розв'язку, тобто

$$\lim_{x \rightarrow 0} x^m k \frac{\partial u}{\partial x} = 0, \quad m = 1, 2 \quad [22, \text{с. 403}].$$

Приклади використання підпрограм. У наведених нижче прикладах використовується підпрограма STEP1, що реалізує один крок (по змінній часу) різницевої схеми (4) — (6). Зазначимо, що фактично цю реалізацію виконує підпрограма STEP, яка викликається з підпрограми STEP1. Головне ж призначення підпрограми STEP1 полягає в розподілі робочого масиву WORK, завдяки чому значно скорочується список її параметрів і вона стає більш зручною для користування.

Приклад 1. Знайдемо розв'язок в декартових, циліндричних і сферичних координатах ($m = 0; 1; 2$) рівняння теплопровідності

$$\begin{aligned} \frac{\partial u}{\partial t} = & \frac{1}{x^m} \frac{\partial}{\partial x} \left(x^m (e^{-0,5t} (2-x) + 1) \frac{\partial u}{\partial x} \right) - e^{-0,5t} (x-1) u + \\ & + e^{-0,5t} (-2,5x^{m+1} (2-x) - 5 (2x^m (e^{-0,5t} (2-x) + 1) ((m+1)(2-x) - x) - \\ & - e^{-0,5t} ((m+1)x^m (2-x) - x^{m+1}) - (e^{-0,5t} (2-x) + 1) x^m (m+2)) + \\ & + (x-1) (5e^{-0,5t} x^{m+1} (2-x) + 2), \quad x \in (1, 2), \quad 0 < t \leq 1, \end{aligned}$$

якщо початковими та крайовими умовами є такі:

$$\begin{aligned} u(x, 0) &= 5x^{m+1} (2-x) + 2, \quad x \in (1, 2); \\ (e^{-0,5t} + 1) \frac{\partial u(1, t)}{\partial x} &= 2u(1, t) - 2(5e^{-0,5t} + 2) + (e^{-0,5t} + 1) \cdot 5me^{-0,5t}; \\ - \frac{\partial u(2, t)}{\partial x} &= 2,5u(2, t) - 5 + 5e^{-0,5t} \cdot 2^{m+1}, \\ &t > 0. \end{aligned}$$

Головна програма з використанням підпрограм STEP1, а також допоміжних підпрограм-функцій для обчислення вихідних даних задачі має вигляд:

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
С ГОЛОВНА ПРОГРАМА, ЩО ІЛЮСТРУЄ РОЗВ'ЯЗУВАННЯ
С ОДНОВИМІРНОГО РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ В
С ДЕКАРТОВИХ, ЦІЛІНДРИЧНИХ І СФЕРИЧНИХ КООРДИНАТАХ
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

```

REAL Y(11),WORK(54)
EXTERNAL FUNP,Q,P,K,PMU1
COMMON /TEST/M
DATA A,B/1.,2./,N/11/,TFINAL/1./
TYPE 102
ACCEPT *,NW
TYPE 103
ACCEPT *,TAU
TYPE 104
ACCEPT *,SIGMA
TYPE 105
ACCEPT *,M
WRITE(NW,*)'NW=' ,NW, 'M=' ,M
WRITE(NW,*)'TAU=' ,TAU, 'SIGMA=' ,SIGMA

```

C

```

T=0.
WORK(1)=A
WORK(2)=B
WORK(3)=SIGMA
WORK(4)=1.
WORK(5)=1.
WORK(6)=2.
WORK(7)=2.5
WORK(8)=M

```

```

C  ОБЧИСЛЕННЯ ПОЧАТКОВОГО РОЗПОДІЛУ ТЕМПЕРАТУРИ
    H=(B-A)/FLOAT(N-1)
    X=A
    DO 10 I=1,N.
        Y(I)=5.*X**(M+1)*(2.-X)+2.
    C
    X=X+H
20  CALL STEP1(T,TAU,Y,N,FUNP,Q,PK,PMU1,IERR,WORK)
    IF(IERR,NE.0) GO TO 30
    IF(T+0.5*TAU.LT.TFINAL) GO TO 20
C  ВИВЕДЕННЯ РЕЗУЛЬТАТІВ ОБЧИСЛЕНЬ
    WRITE(NW,100)T,Y
    STOP
C
30  WRITE(NW,101)IERR
    STOP
100  FORMAT(' НАБЛИЖЕНИЙ РОЗВ'ЯЗОК ПРИ T=',
    *      IPE11.4/5(X,IPE9.2))
101  FORMAT(' IERR=',I5)
102  FORMAT('$ВВЕДІТЬ НОМЕР КАНАЛУ ВИВЕД. NW=')
103  FORMAT('$ВВЕДІТЬ РОЗМІР КРОКУ ПО ЗМІННИЙ',
    *      'ЧАСУ TAU=')
104  FORMAT('$ВВЕДІТЬ ЗНАЧЕННЯ ВАГОВОГО КОЕФІЦІ',
    *      'ЄНТА РІЗНИЦЕВОЇ СХЕМИ SIGMA=')
105  FORMAT('$ВВЕДІТЬ НОМЕР СИСТЕМИ КООРДИНАТ M=')
    END

    FUNCTION FUNP(X,T)
C  ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯННЯ
    COMMON/TEST/M
    M1=M+1
    VX=2.-X
    ET=EXP(-0.5*T)
    XM=X**M
    XM1=X**M1
    IF(M.EQ.0) XMM=0.
    IF(M.GT.0) XMM=M*X**(M-1)
    FUNP=ET*(-2.5*XM1*VX-5.*(-(ET*VX+1.)*(M+2)*XM-
    *      ET*(M1*XM*VX-XM1)+2*XMM*(ET*VX+1))*
    *      (M1*VX-X))+(X-1.)*(5.*ET*VX*XM1+2.))
    RETURN
    END

    FUNCTION Q(X,T)
C  ОБЧИСЛЕННЯ КОЕФІЦІЄНТА ТЕПЛОВІДДАЧІ
    Q=EXP(-0.5*T)*(X-1.)
    RETURN
    END

    FUNCTION PK(X,T)
C  ОБЧИСЛЕННЯ КОЕФІЦІЄНТА ТЕПЛОПРОВІДНОСТІ
    PK=EXP(-0.5*T)*(2.-X)+1.
    RETURN
    END

```

```

FUNCTION PMU1(T,PMU2)
С ОБЧИСЛЕННЯ ПРАВИХ ЧАСТИН У КРАЙОВИХ УМОВАХ
COMMON /TEST/M
ET=EXP(-0.5*T)
PMU1=2.*(5.*ET+2.)-(5.*ET*M*(ET+1.))
PMU2=5.-5.*ET*2.**(M+1)
RETURN
END

```

У цій програмі крок по змінній x дорівнює 0,1. Значення цього кроку та вагового коефіцієнта вона дозволяє ввести в діалоговому режимі з терміналу ЕОМ. Обчислення за головною програмою дають такий результат:

```

NW=      2 M=      0
TAU= 1.9999999E-03 SIGMA= 0.0000000
НАБЛИЖЕНИЙ РОЗВ'ЯЗОК ПРИ T=9.9999E-01
5.03E+00 5.00E+00 4.91E+00 4.76E+00 4.55E+00
4.27E+00 3.94E+00 3.55E+00 3.09E+00 2.57E+00
2.00E+00

```

```

NW=      3 M=      1
TAU= 4.9999997E-02 SIGMA= 0.5000000
НАБЛИЖЕНИЙ РОЗВ'ЯЗОК ПРИ T= 1.0000E+00
5.01E+00 5.29E+00 5.48E+00 5.57E+00 5.55E+00
5.40E+00 5.09E+00 4.62E+00 3.96E+00 3.09E+00
1.99E+00

```

```

NW=      4 M=      2
TAU= 4.9999997E-02 SIGMA= 1.0000000
НАБЛИЖЕНИЙ РОЗВ'ЯЗОК ПРИ T= 1.0000E+00
4.96E+00 5.56E+00 6.13E+00 6.60E+00 6.94E+00
7.07E+00 6.93E+00 6.43E+00 5.51E+00 4.06E+00
1.98E+00

```

Приклад 2. Грудочку вугілля сферичної форми діаметром 20 мм, що має початкову температуру 0°C , вміщено в піч, температура якої дорівнює 300°C . Визначити, через який час температура в середині цієї грудочки дорівнюватиме 30°C . Фізичні характеристики вугілля мають такі значення: $\lambda = 0,175 \text{ Вт}/(\text{м} \cdot \text{K})$; $c = 1,3 \text{ кДж}/(\text{кг} \cdot \text{K})$; $\rho = 1400 \text{ кг}/\text{м}^3$; $\gamma = 58,2 \text{ Вт}/(\text{м}^2 \cdot \text{K})$.

Процес нагрівання грудочки вугілля може бути описаний таким диференціальним рівнянням:

$$c\rho \frac{\partial u}{\partial t} = \frac{\lambda}{x^2} \frac{\partial}{\partial x} \left(x^2 \frac{\partial u}{\partial x} \right), \quad x \in (0, R), \quad t > 0; \quad (12)$$

$$u(x, 0) = 0, \quad x \in [0, R]; \quad x^2 \frac{\partial u}{\partial x} \Big|_{x=0} = 0; \quad (13)$$

$$\left(\lambda \frac{\partial u}{\partial x} + \gamma(u - u_{\text{ср}}) \right) \Big|_{x=R} = 0, \quad t > 0,$$

де $R = 10^{-2}$ м; $u_{\text{ср}} = 300^\circ\text{C}$.

Введемо безрозмірні змінні

$$v(x, t) = (u(x, t) - u_{\text{кр}})/u_{\text{кр}}; \quad t_1 = a^2 t/R^2; \quad x_1 = x/R, \quad (14)$$

де $a^2 = \lambda/(c\rho)$; $u_{\text{кр}}$ — деяка стала величина, якій відповідає, наприклад, $u_{\text{ср}}$.

Переходячи в задачі (12), (13) до безрозмірних змінних за формулами (14), дістаємо таку задачу для функції $v(x_1, t_1)$:

$$\frac{\partial v}{\partial t_1} = \frac{1}{x_1^2} \frac{x}{\partial x_1} \left(x_1^2 \frac{\partial v}{\partial x_1} \right), \quad x_1 \in (0, 1), \quad t_1 > 0; \quad (15)$$

$$v(x_1, 0) = v_0(x_1), \quad x_1 \in [0, 1];$$

$$x_1^2 \frac{\partial v}{\partial x_1} \Big|_{x_1=0} = 0, \quad \left(\frac{\partial v}{\partial x_1} + \gamma_1 v \right)_{x_1=1} = 0, \quad (16)$$

де $v_0(x_1) = (u_0(x) - u_{cp})/u_{cp}$; $\gamma_1 = \gamma R/\lambda$.

Головна програма з використанням підпрограми STEP1 для розв'язування рівнянь (15), (16) має вигляд:

```

С ГОЛОВНА ПРОГРАМА, ЩО ІЛЮСТРУЄ РОЗВ'ЯЗУВАННЯ
С ОДНОВИМІРНОГО РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ В
С СФЕРИЧНИХ КООРДИНАТАХ

```

```

REAL Y(20),V(220),WORK(99)
EXTERNAL FUNP,Q,PK,PMU1
DATA NW,IEND,K,MM/2,0,1,6/,N/20/,
* A,B/0.,10.E-3/,T,TAU/0.,0.002/,
* PK0,C,P0/0.175,1300.,1400./,
* YKP/300./,BT2/58.2/,TPRIN/20./

```

```

NM=N*MM
H=(B-A)/FLOAT(N-1)
X=A

```

```

10 DO 10 I=1,N
    V(I)=X
    X=X+H
    AR=B**2*C*R0/PK0
    TPRIN=TPRIN/AR
    TOUT1=TPRINI
    M1=0
    WORK(1)=0.
    WORK(2)=1.
    WORK(3)=0.5
    WORK(4)=1.
    WORK(5)=1.
    WORK(6)=0.
    WORK(7)=BT2*B/PK0
    WORK(8)=2

```

С ОБЧИСЛЕННЯ ПОЧАТКОВОГО РОЗПОДІЛУ ТЕМПЕРАТУРИ

```

20 DO 20 I=1,N
    Y(I)=-1.

```

```

30 CALL STEP1(T,TAU,Y,N,FUNP,Q,PK,PMU1,IERR,WORK)
IF(IERR.NE.0) GO TO 80
IF(YKP*(1.+Y(1)).LT.30.) GO TO 40
IEND=1
MM=M1+2
GO TO 50

```

```

40 IF(T+10.5*TAU.LT.TOUTI) GO TO 30
50 M1=M1+1
   DO 60 I=1,N
60   V(N*M1+I)=YKR*(1.+Y(I))
   TOUTI=TOUTI+TPRIN1
   IF(M1.LT.MM-1) GO TO 30
C  ВИВЕДЕННЯ ГРАФІКА
   CALL GRAPH(NW,K,N,MM,NM,V)
   IF(IEND.EQ.1) GO TO 70
   K=K+1
   M1=0
   GO TO 30
C  ВИВЕДЕННЯ РЕЗУЛЬТАТІВ ОБЧИСЛЕНЬ
70  WRITE(NW,100) T*AR,(Y(1)+1.)*YKR
   STOP
80  WRITE(NW,101)IERR
   STOP
100 FORMAT(' ЧАС ЗАКІНЧЕННЯ НАГРІВУ=',E12.5,
*         'СЕКУНД/' ТЕМПЕРАТУРА В СЕРЕДИНІ',
*         'ТІЛА=',E12.5)
101 FORMAT('IERR=',I5)
   END

   FUNCTION FUNP(X,T)
C  ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯННЯ
   FUNP=0.
   RETURN
   END

   FUNCTION Q(X,T)
C  ОБЧИСЛЕННЯ КОЕФІЦІЕНТА ТЕПЛОВІДДАЧІ
   Q=0.
   RETURN
   END

   FUNCTION PK(X,T)
C  ОБЧИСЛЕННЯ КОЕФІЦІЕНТА ТЕПЛОПРОВІДНОСТІ
   PK=1.
   RETURN
   END

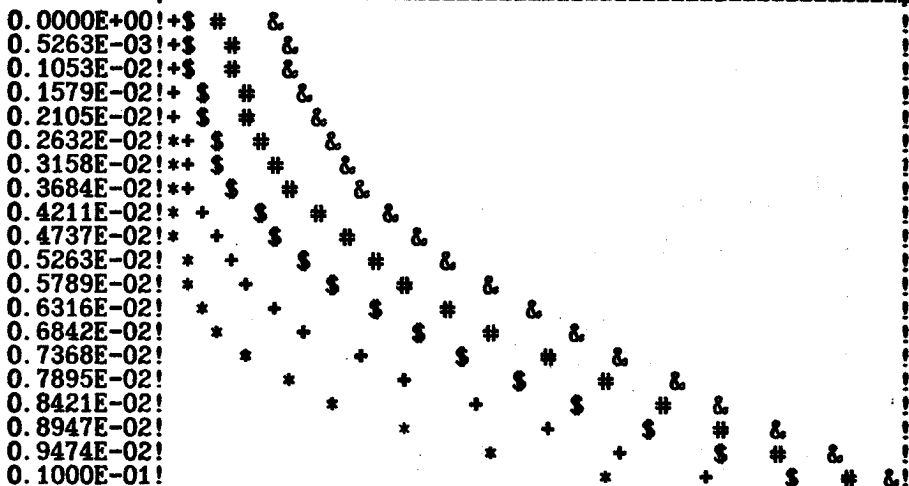
   FUNCTION PMU1(T,PMU2)
C  ОБЧИСЛЕННЯ ПРАВИХ ЧАСТИН У КРАЙОВИХ УМОВАХ
   PMU1=0.
   PMU2=0.
   RETURN
   END

```

У цій програмі значення вагового коефіцієнта σ дорівнює 0,5 (тобто маємо схему Кранка — Ніколсона), причому графік розподілу температури будується через кожні 10 с.

Обчислення на ЕОМ за наведеною програмою дають такий результат:

ФУНКЦІЇ: 1 - * 2 - + 3 - \$ 4 - # 5 - &
 АБСЦИСИ ГРАФІК 1



ШКАЛА ОРДИНАТ

!-----!-----!-----!-----!-----!-----!-----!-----!-----!
 1 2 3 4 5 6 7 8 9 10 11

STEPY= 0.20700E+02

VMIN= 0.27001E-02 VMAX= 0.20700E+03

ЧАС ЗАКІНЧЕННЯ НАГРІВУ = 0.99840E+02 СЕКУНД
 ТЕМПЕРАТУРА В СЕРЕДИНІ ТІЛА = 0.30748E+02

Запитання для самоперевірки

1. Який вигляд має різницева схема з ваговими коефіцієнтами для розв'язування одновимірного рівняння теплопровідності в одній з систем координат (декартових, циліндричних або сферичних)? 2. Що таке явні та неявні різницеві схеми розв'язування рівняння теплопровідності? Які переваги та недоліки їх? 3. Які є основні методи побудови різницевих схем для розв'язування одновимірного нестационарного рівняння теплопровідності?

Контрольні завдання

1. Знайти розв'язок рівняння теплопровідності (1) — (3) при таких вихідних даних:

$$k(x, t) = e^{-0,5t} (b - x) + 1; \quad q(x, t) = e^{-0,5t} (x - a);$$

$$f(x, t) = e^{-0,5t} (-2x^{m+1} (b - x) - 5(2mx^{m-1} (e^{-0,5t} (b - x) + 1)) ((m + 1) \times$$

$$\times (b - x) - x) - (e^{-0,5t} (b - x) + 1) (m + 2) x^m - e^{-0,5t} ((m + 1) x^m (b - x) -$$

$$- x^{m+1})) + (x - a) (5e^{-0,5t} (b - x) x^{m+1} + a + 1));$$

$$\mu_1(t) = \beta_1 (5e^{-0,5t} a^{m+1} (b-a) + a + 1) - \alpha_1 (5e^{-0,5t} ((m+1) a^m (b-a) - a^{m+1})) (e^{-0,5t} (b-a) + 1));$$

$$\mu_2(t) = \beta_2 (a + 1) - \alpha_2 \cdot 5e^{-0,5t} b^{m+1},$$

де $m, a, b, \alpha_k, \beta_k (k = 1, 2)$ — задані величини, причому $m = 0, 2, 0 \leq a < b, \alpha_k = 0, 1, \beta_k \geq 0 (k = 1, 2)$.

Знайдений розв'язок дослідити через однакові проміжки часу, які дорівнюють $0,2T$. Варіанти завдання, що визначаються величинами $m, a, b, T, \alpha_k, \beta_k (k = 1, 2)$, задаються викладачем.

Зауваження. Для контролю знайдених результатів можна скористатися точним розв'язком задачі $u = 5e^{-0,5t} x^{m+1} (b-x) + a + 1$.

2. Довгий сталевий вал діаметром 12 см, що має температуру 20°C , вміщено в піч, температура якої становить 820°C . Визначити час, необхідний для нагрівання вала, якщо воно вважається закінченим, коли температура в середині вала дорівнюватиме 800°C . Знайти температуру на поверхні вала в кінці його нагрівання. Фізичні характеристики сталю вала мають такі значення: $\lambda = 45,5 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,46 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 790 \text{ кг/м}^3$; $\gamma = 140 \text{ Вт/(м}^2 \cdot \text{К)}$.

3. Довгий сталевий стержень діаметром 10 см був нагрітий до температури 500°C . Визначити час, після закінчення якого температура стержня, вміщеного в середовище з температурою 20°C , зменшиться в п'ять разів. Фізичні характеристики сталі ті самі, що і в попередньому завданні.

4. Визначити температуру в середині і на поверхні бетонної колони діаметром $0,5 \text{ м}$ через одну годину, якщо раптово температура навколишнього середовища знизилась з $+20$ до -20°C . Фізичні параметри бетону мають такі значення: $\lambda = 0,885 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,882 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 2100 \text{ кг/м}^3$; $\gamma = 8,15 \text{ Вт/(м}^2 \cdot \text{К)}$.

5. Однорідний сталевий циліндр радіусом $R = 2,5 \text{ см}$ має початкову температуру 5°C . В середині циліндра існують теплові джерела, щільність розподілу яких описується функцією $F = 2 (x/R - 1) \cos 3t$, а на поверхні відбувається теплообмін з середовищем, яке має нульову температуру. Визначити температуру в середині та на поверхні циліндра через 10 хв. Фізичні характеристики сталю вала ті самі, що і в завданні 2.

6. Гумову кулю діаметром 10 см, нагріту до температури 140°C , вміщено в середовище з температурою 15°C . Визначити температуру в середині і на поверхні кулі через 30 хв. Фізичні характеристики гумової кулі мають такі значення: $\lambda = 0,7 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,8 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 1500 \text{ кг/м}^3$; $\gamma = 65 \text{ Вт/(м}^2 \cdot \text{К)}$.

7. Грудочку вугілля сферичної форми, що має температуру 0°C , вміщено в середовище з температурою 300°C . Визначити час, який потрібен для підвищення температури в середині цієї грудочки до 100°C . Фізичні характеристики вугілля ті самі, що і в розглянутому вище прикладі 2.

8. Визначити час, після закінчення якого температура в середині мідної кулі діаметром 10 см дорівнюватиме 10°C , якщо поверхня кулі підтримується при нульовій температурі, а початкова температура кулі дорівнює 50°C . Фізичні характеристики мідної кулі мають такі значення: $\lambda = 398 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,38 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 8900 \text{ кг/м}^3$.

9. Алюмінієву кулю діаметром $D = 30 \text{ см}$ вміщено в середовище з температурою 20°C . Визначити температуру в середині кулі через 20 хв, якщо початковий розподіл температури в ній задається функцією $u_0(x) = 100 + 20 \sin(x(D/2 - x))$. Фізичні характеристики алюмінієвої кулі мають такі значення: $\lambda = 220 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,89 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 2700 \text{ кг/м}^3$; $\gamma = 300 \text{ Вт/(м}^2 \cdot \text{К)}$.

10. Однорідна сталеві куля діаметром 10 см знаходиться в середовищі з температурою 0°C . Починаючи з моменту часу $t = 0$ температура навколишнього середовища змінюється за законом $u_{\text{ср}}(t) = 10 \sin(0,01t)$. Визначити температуру в середині кулі через 15 хв. Фізичні характеристики сталю вала мають такі значення: $\lambda = 42,1 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,445 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 7800 \text{ кг/м}^3$; $\gamma = 140 \text{ Вт/(м}^2 \cdot \text{К)}$.

11. Визначити, через який час температура в середині алюмінієвого дроту завдовжки 20 см з нетеплопровідною зовнішньою поверхнею досягає 10°C , якщо один кінець дроту теплоізолюваний, а другий підтримується при температурі 0°C . Початкова температура дроту дорівнює 20°C . Фізичні характеристики алюмінієвого дроту ті самі, що і в завданні 9.

12. Цегляна плита завдовжки 0,5 м має температуру 18°C . Визначити, як зміниться температура на поверхні та в середині плити через 1 год, якщо температура навколишнього середовища раптово знизиться до 5°C . Фізичні характеристики цегляної плити мають такі значення: $\lambda = 0,77 \text{ Вт/(м} \cdot \text{К)}$; $c = 0,83 \text{ кДж/(кг} \cdot \text{К)}$; $\rho = 1600 \text{ кг/м}^3$; $\gamma = 7 \text{ Вт/(м}^2 \cdot \text{К)}$.

13. Крізь мідний провідник, який має форму плоскої пластинки завдовжки 2 см, проходить струм, внаслідок чого виділяється тепло зі сталюю щільністю 3500 кВт/м . Визначити розподіл температури в пластині через 10 с, якщо на її границі відбувається теплообмін з навколишнім середовищем за законом Ньютона. Початкова температура дорівнює 0°C , температура навколишнього середовища становить 20°C . Фізичні характеристики мідного провідника мають ті самі значення, що і в завданні 8, а $\gamma = 365 \text{ Вт/(м}^2 \cdot \text{К)}$.

14. Визначити розподіл температури в мідному стержні завдовжки $l = 10 \text{ см}$ з нетеплопровідною зовнішньою поверхнею через 1 хв, якщо на його кінцях підтримується температура 0°C , а початкова температура розподілена за законом $u_0(x) = 10 \sin(x(x-l)/l)$. Фізичні характеристики мідного стержня мають ті самі значення, що і в завданні 8.

§ 17. Економічні методи розв'язування двовимірної рівняння теплопровідності

Постановка задачі. Розв'язати мішану задачу для двовимірної рівняння теплопровідності в прямокутнику $\Omega = \{x = (x_1, x_2) : 0 < x_\alpha < l_\alpha, \alpha = 1, 2\}$:

$$\frac{\partial u}{\partial t} = Lu + f(x, t), \quad x \in \Omega, t \in (0, T]; \quad (1)$$

$$u(0, x_2, t) = g_1(x_2, t), \quad u(l_1, x_2, t) = g_3(x_2, t); \quad (2)$$

$$u(x_1, 0, t) = g_2(x_1, t), \quad u(x_1, l_2, t) = g_4(x_1, t),$$

$$u(x_1, x_2, 0) = u_0(x_1, x_2), \quad (3)$$

де $Lu = \Delta u = L_1u + L_2u$; $L_\alpha u = \frac{\partial^2 u}{\partial x_\alpha^2}$, $\alpha = 1, 2$.

Функції $f(x_1, x_2, t)$, $g_h(x_\alpha, t)$, $h = \overline{1, 4}$, $\alpha = \overline{1, 2}$, $u_0(x_1, x_2)$ і метод розв'язування задачі (1) — (3) визначаються варіантом завдання.

Методичні вказівки. В прямокутнику $\bar{\Omega}$ побудуємо рівномірну по кожному напрямку x_α сітку $\bar{\omega}_h = \{x = (x_1, x_2) : x_\alpha \in \omega_{h,\alpha}, \alpha = 1, 2\}$; $\bar{\omega}_{h,\alpha} = \{x_\alpha = i_\alpha h_\alpha, i_\alpha = 0, 1, \dots, N_\alpha, h_\alpha = l_\alpha/N_\alpha\}$, а на відрізку $[0, T]$ — сітку $\bar{\omega}_\tau = \{t = t_n = n\tau, n = 0, 1, \dots, M, \tau = T/M\}$.

Апроксимуємо рівняння (1) різницевою схемою з ваговими коефіцієнтами

$$y_t = \sigma \Delta \hat{y} + (1 - \sigma) \Delta y + \varphi, \quad x \in \omega_h, \quad t \in \omega_\tau, \quad (4)$$

де

$$y_t = (\hat{y} - y)/\tau; \quad y = y^n = y(x, t_n), \quad x \in \bar{\omega}_h;$$
$$\hat{y} = y^{n+1} = y(x, t_{n+1}); \quad \varphi = f(x, t_n + 0,5\tau);$$
$$\Lambda = \Lambda_1 + \Lambda_2; \quad \Lambda_\alpha y = y_{x_\alpha x_\alpha}, \quad \alpha = 1, 2.$$

Покладаючи $\sigma = 0$, дістаємо явну різницеву схему, яка легко реалізується, для чого при переході від ярусу t_n до ярусу t_{n+1} потрібно виконати $O(1/h^2)$ арифметичних операцій, тобто їх кількість пропорційна числу внутрішніх вузлів сітки ω_h . Однак ця схема умовно стійка. Умова стійкості

$$\tau \leq 0,5 (h_1^{-2} + h_2^{-2})^{-1}$$

є досить жорсткою.

Неявні різницеві схеми ($\sigma > 0$) із множини схем з ваговими коефіцієнтами абсолютно (безумовно) стійкі при $\sigma \geq 0,5$, але для їх реалізації необхідно розв'язати СЛАР

$$\hat{y} - \tau\sigma\Lambda\hat{y} = y + \tau(1 - \sigma)\Lambda y + \tau\varphi \quad (5)$$

відносно значень функції \hat{y} на верхньому ярусі. При цьому кількість потрібних арифметичних операцій значно більша, ніж при реалізації явної різницевої схеми.

Поняття економічної різницевої схеми для багатовимірних рівнянь математичної фізики, зокрема рівняння (1), пов'язане з виконанням двох вимог:

1) схема має бути абсолютно стійкою (як, наприклад, неявна схема при $\sigma \geq 0,5$);

2) знаходження розв'язку на верхньому ярусі при цій схемі потребує такої кількості арифметичних операцій, яка пропорційна числу вузлів просторової сітки (як для явної різницевої схеми).

Існують такі три класи економічних різницевих схем:

1) схеми змінних напрямків, або поздовжньо-поперечні схеми [22, с. 445];

2) схеми з факторизованим оператором [22, с. 458];

3) локально-одновимірні схеми, які належать до адитивних схем [22, с. 495].

▼ Однією з перших економічних різницевих схем для задачі (1) — (3) була схема змінних напрямків, або поздовжньо-поперечна схема (ППС). В літературі вона називається ще *схемою Пісмена — Рекфорда*.

Поряд з основними ярусами t_n, t_{n+1} в ній розглядаються проміжні яруси $\bar{t} = t_{n+1/2} = t_n + 0,5\tau$, до яких віднесемо значення розв'язку

ку $\bar{y} = y^{n+1/2}$. Перехід від n - до $(n + 1)$ -го ярусу тут здійснюється в два етапи:

$$\frac{\bar{y} - y}{0,5\tau} = \Lambda_1 \bar{y} + \Lambda_2 y + \varphi; \quad (6)$$

$$\frac{\hat{y} - \bar{y}}{0,5\tau} = \Lambda_1 \bar{y} + \Lambda_2 \hat{y} + \varphi. \quad (7)$$

Перше різницеве рівняння неявне по x_1 і явне по x_2 , друге — явне по x_1 , але неявне по x_2 . До цих рівнянь додаються такі крайові умови:

$$\bar{y} = \bar{g}_k = \frac{1}{2} (\hat{g}_k + g_k) - \frac{\tau}{4} \Lambda_2 (\hat{g}_k - g_k) \quad (8)$$

при $x_1 = 0$ ($i_1 = 0$, $k = 2$) і $x_1 = l_1$ ($i_1 = N_1$, $k = 3$); (9)

$$\hat{y} = \check{g}_k$$

при $x_2 = 0$ ($i_2 = 0$, $k = 2$) і $x_2 = l_2$ ($i_2 = N_2$, $k = 4$).

Тут $g_k = g_k(x_\alpha, t)$, $\hat{g}_k = g_k(x_\alpha, t + \tau)$ — значення функцій, які входять в крайові умови (2) у відповідній точці просторових змінних x_α в момент часу t або $t + \tau$.

Початкові умови для (6), (7) мають вигляд

$$y(x_1, x_2, 0) = u_0(x_1, x_2), \quad x \in \bar{\omega}_h. \quad (10)$$

Схема (6) — (10) абсолютно стійка [22, с. 449] і має другий порядок точності. Алгоритм реалізації цієї схеми зводиться до розв'язування різницевих рівнянь другого порядку методом прогонки [22, с. 35].

Запишемо рівняння (6), (8) в індексній формі

$$\begin{aligned} \frac{\tau}{h_1^2} \bar{y}_{i_1+i_2} - 2 \left(1 + \frac{\tau}{h_1^2} \right) \bar{y}_{i_1 i_2} + \frac{\tau}{h_1^2} \bar{y}_{i_1-i_2} &= -F_{i_1 i_2}, \\ i_\alpha &= 1, 2, \dots, N_{\alpha-1}, \quad \alpha = 1, 2; \\ \bar{y}_{0 i_2} &= \bar{g}_1, \quad \bar{y}_{N_1 i_2} = \bar{g}_3. \end{aligned} \quad (11)$$

Рівнянням (7), (9) відповідають такі різницеві рівняння:

$$\frac{\tau}{h_2^2} \hat{y}_{i_1 i_2+1} - 2 \left(1 + \frac{\tau}{h_2^2} \right) \hat{y}_{i_1 i_2} + \frac{\tau}{h_2^2} \hat{y}_{i_1 i_2-1} = -\bar{F}_{i_1 i_2}; \quad (12)$$

$$\hat{y}_{i_1 0} = \hat{g}_2, \quad \hat{y}_{i_1 N_2} = \hat{g}_4, \quad i_\alpha = 1, 2, \dots, N_\alpha - 1, \quad \alpha = 1, 2,$$

де $F = 2y + \tau(\Lambda_2 y + \varphi)$; $\bar{F} = 2\bar{y} + \tau(\Lambda_1 \bar{y} + \varphi)$.

Різницеве рівняння (11) розв'язується методом прогонки по i_1 при фіксованому i_2 (прогонка в поздовжньому напрямку), а рівняння (12) — прогонкою по i_2 при фіксованому i_1 (прогонка в поперечному

напрямку). Завдяки застосуванню методу змінних напрямків загальна кількість арифметичних операцій для визначення розв'язку \hat{y} пропорційна числу вузлів просторової сітки, тобто $0((N_1 - 1)(N_2 - 1))$. Отже, різницева схема (6) — (10) є економічною.

▼ До економічних різницевих схем розв'язування багатовимірних задач належать також локально-одновимірні схеми (ЛОС), які можуть використовуватись для областей більш загального вигляду, ніж прямокутні.

Розглянемо застосування ЛОС для задачі (1) — (3). Як і при використанні ППС, розв'язок \hat{y} на ярусі $t + \tau$ тут відшукується в два етапи:

$$\frac{\bar{y} - y}{\tau} = \Lambda_1 \bar{y} + \varphi_1; \quad (13)$$

$$\frac{\hat{y} - \bar{y}}{\tau} = \Lambda_2 \hat{y} + \varphi_2, \quad (14)$$

де $\varphi_1 + \varphi_2 = \varphi$ (наприклад, $\varphi_1 = \varphi_2 = 0,5\varphi$), а $\varphi = f(x, t)$.

Крайові та початкові умови при цьому мають вигляд

$$\bar{y} = g_k(x_2, \bar{t}) \text{ при } x_1 = 0 \quad (i_1 = 0, k = 1), \quad x_1 = l_1 \quad (i_1 = N_1, k = 3); \quad (15)$$

$$\hat{y} = g_k(x, \hat{t}) \text{ при } x_2 = 0 \quad (i_2 = 0, k = 2), \quad x_2 = l_2 \quad (i_2 = N_2, k = 4);$$

$$y(x_1, x_2, 0) = u_0(x_1, x_2). \quad (16)$$

Кожне з рівнянь (13), (14) на відміну від (6), (7) не апроксимує диференціальне рівняння (1). Проте сумарна похибка апроксимації прямуватиме до нуля. Отже, схема (13) — (16) матиме сумарну апроксимацію [22, с. 478]. Крім того, вона абсолютно стійка і має другий порядок точності по h_1, h_2 і перший по τ . Зауважимо, що форма запису крайових умов для ЛОС значно простіша, ніж для ППС.

Реалізація ЛОС виконується за допомогою методу прогонки по поздовжніх і поперечних лініях сітки. Різницева схема (13) — (16) в індексній формі має вигляд

$$\frac{\tau}{h_1^2} \bar{y}_{i_1, t_1 + 1} - \left(1 + \frac{2\tau}{h_1^2}\right) \bar{y}_{i_1, t_1} + \frac{\tau}{h_1^2} \bar{y}_{i_1 - 1, t_1} = -F_{i_1, t_1}; \quad (17)$$

$$\bar{y}_{0, t_1} = g_1(x_2, \bar{t}), \quad \bar{y}_{N_1, t_1} = g_3(x_2, \bar{t}), \quad i_\alpha = 1, 2, \dots, \\ N_\alpha - 1, \quad \alpha = 1, 2;$$

$$\frac{\tau}{h_2^2} \hat{y}_{i_2, t_2 + 1} - \left(1 + \frac{2\tau}{h_2^2}\right) \hat{y}_{i_2, t_2} + \frac{\tau}{h_2^2} \hat{y}_{i_2, t_2 - 1} = -\bar{F}_{i_2, t_2}; \quad (18)$$

$$\hat{y}_{i_2, 0} = g_2(x_1, \hat{t}), \quad \hat{y}_{i_2, N_2} = g_4(x_1, \hat{t}), \quad i_\alpha = 1, 2, \dots, \\ N_\alpha - 1, \quad \alpha = 1, 2,$$

де $F = y + \tau\varphi_1$; $\bar{F} = \bar{y} + \tau\varphi_2$.

Очевидно, кількість арифметичних операцій, необхідних для знаходження розв'язку \hat{u} у всіх вузлах сітки, пропорційна числу вузлів сітки ω_n , тобто схема є економічною.

Приклад використання підпрограм. Підпрограми PPS, LOC дають змогу реалізувати один крок у напрямку змінної часу ППС і ЛОС відповідно до формул (11), (12) і (17), (18). Застосування цих підпрограм проілюструємо на прикладі знаходження розв'язку такої задачі:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} - 3, \quad x \in \Omega = \{0 < x_\alpha < 1, \alpha = 1, 2\};$$

$$u|_{x_1=0} = x_2^2 + t, \quad u|_{x_2=0} = x_1^2 + t;$$

$$u|_{x_1=1} = 1 + t + x_2^2, \quad u|_{x_2=1} = 1 + t + x_1^2;$$

$$u|_{t=0} = x_1^2 + x_2^2.$$

Якщо покласти $N_1 = N_2 = 5$ і $\tau = 0,01$, то головна програма з використанням підпрограм PPS, LOC для розв'язування цієї задачі, матиме вигляд:

```

С ГОЛОВНА ПРОГРАМА, ЩО ІЛЮСТРУЄ РОЗВ'ЯЗУВАННЯ
С ДВОВИМІРНОГО РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ

```

```

REAL Y(6,6),Y1(6,6),YR(6,6),X1(6),X2(6),

```

```

* AR(5),BR(5)

```

```

DATA N1,N2,NW/6,6,2/,TAU,TFINAL/0.01,0.1/

```

```

H1=1./FLOAT(N1-1)

```

```

H2=1./FLOAT(N2-1)

```

```

T=0.

```

```

С ОБЧИСЛЕННЯ ПОЧАТКОВОГО РОЗПОДІЛУ ТЕМПЕРАТУРИ

```

```

X1(1)=0.

```

```

X2(1)=0.

```

```

DO 10 I=2,N1

```

```

10 X1(I)=X1(I-1)+H1

```

```

DO 20 J=2,N2

```

```

20 X2(J)=X2(J-1)+H2

```

```

DO 30 I=1,N1

```

```

DO 30 J=1,N2

```

```

Y(I,J)=X1(I)*X1(I)+X2(J)*X2(J)

```

```

30 Y1(I,J)=Y(I,J)

```

```

С ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ЗА ДОПОМОГОЮ ППС

```

```

40 CALL PPS(T,TAU,N1,N2,H1,H2,Y,YR,AR,BR)

```

```

IF(T+.5*TAU.LT.TFINAL) GO TO 40

```

```

WRITE(NW,100) T,((Y(I,J),J=1,N2),I=1,N1)

```

```

CALL LINUR(NW,45,30,5,0.4,X1,X2,N1,N2,N1,Y)

```

```

С ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ЗА ДОПОМОГОЮ ЛОС

```

```

T=0.

```

```

50 CALL LOC(T,TAU,N1,N2,H1,H2,Y1,YR,AR,BR)

```

```

IF(T+.5*TAU.LT.TFINAL) GOTO 50

```

```

WRITE(NW,100) T,((Y1(I,J),J=1,N2),I=1,N1)

```

```

CALL LINUR(NW,45,30,5,0.4,X1,X2,N1,N2,N1,Y1)

```

```

STOP

```

```

100 FORMAT(5X,'T=',E12.5,6/(X,1P6E9.2)))

```

```

END

```

Підпрограми-функції тут застосовуються такі:

- ```
FUNCTION G1(X2,T)
C ОБЧИСЛЕННЯ ГРАНИЧНОЇ ФУНКЦІЇ ПРИ X1=0
G1=X2*X2+T
RETURN
END

FUNCTION G2(X1,T)
C ОБЧИСЛЕННЯ ГРАНИЧНОЇ ФУНКЦІЇ ПРИ X2=0
G2=X1*X1+T
RETURN
END

FUNCTION G3(X2,T)
C ОБЧИСЛЕННЯ ГРАНИЧНОЇ ФУНКЦІЇ ПРИ X1=1
G3=1.+X2*X2+T
RETURN
END

FUNCTION G4(X1,T)
C ОБЧИСЛЕННЯ ГРАНИЧНОЇ ФУНКЦІЇ ПРИ X2=1
G4=1.+X1*X1+T
RETURN
END

FUNCTION F(X1,X2,T)
C ОБЧИСЛЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯННЯ
F=-3.
RETURN
END
```

Обчислення за допомогою підпрограм PPS і LOC дають однаковий результат, який подано на с. 186.

### Запитання для самоперевірки

1. Які різницеві схеми називаються економічними? 2. Який вигляд мають економічні різницеві схеми розв'язування багатовимірних параболічних рівнянь? 3. Який вигляд ППС для розв'язування двовимірного рівняння теплопровідності? Яку точність розв'язку забезпечує ця схема? 4. Який вигляд має ЛОС для розв'язування двовимірного рівняння теплопровідності? Яку точність розв'язку забезпечує ця схема? 5. Що таке стійкість різницевої схеми? 6. Які переваги й недоліки мають явні та неявні різницеві схеми для розв'язування рівняння теплопровідності?

### Контрольні завдання

Усі варіанти завдань стосуються розв'язування задачі (1) — (3) в області  $\bar{\Omega} = \{0 \leq x_\alpha \leq 1, \alpha = 1, 2\}$  і відрізняються один від одного крайовими та початковими умовами або правою частиною рівнянь. Метод розв'язування їх і проміжок інтегрування задаються викладачем. Для більш наочного відображення результатів обчислень можна скористатися підпрограмою виведення ліній рівня.

Варіант 1:

$$f = \begin{cases} 1, & |x_1 - a| < 0,25, \quad |x_2 - b| < 0,25; \\ 0, & |x_1 - a| \geq 0,25, \quad |x_2 - b| \geq 0,25, \end{cases}$$

T= 0.10000E+00

|          |          |          |          |          |          |
|----------|----------|----------|----------|----------|----------|
| 1.00E-01 | 1.40E-01 | 2.60E-01 | 4.60E-01 | 7.40E-01 | 1.10E+00 |
| 1.40E-01 | 1.80E-01 | 3.00E-01 | 5.00E-01 | 7.80E-01 | 1.14E+00 |
| 2.60E-01 | 3.00E-01 | 4.20E-01 | 6.20E-01 | 9.00E-01 | 1.26E+00 |
| 4.60E-01 | 5.00E-01 | 6.20E-01 | 8.20E-01 | 1.10E+00 | 1.46E+00 |
| 7.40E-01 | 7.80E-01 | 9.00E-01 | 1.10E+00 | 1.38E+00 | 1.74E+00 |
| 1.10E+00 | 1.14E+00 | 1.26E+00 | 1.46E+00 | 1.74E+00 | 2.10E+00 |

СИМВОЛ ДІАПАЗОН ЗМІНИ ФУНКЦІЇ

|   |             |             |
|---|-------------|-------------|
| 0 | 0.10000E+00 | 0.30000E+00 |
| 1 | 0.40000E+00 | 0.80000E+00 |
| 2 | 0.90000E+00 | 0.13000E+01 |
| 3 | 0.14000E+01 | 0.18000E+01 |
| 4 | 0.19000E+01 | 0.21000E+01 |

+-----+  
!22222222222222222222...333333333333...44444!  
!22222222222222222222...333333333333...4444!  
!22222222222222222222...333333333333...44!  
!22222222222222222222...333333333333...4!  
!.....222222222222222222...3333333333...!  
!111.....2222222222222222...3333333333...!  
!1111111111.....222222222222...3333333333!  
!111111111111.....222222222222...33333333!  
!11111111111111.....222222222222...333333!  
!1111111111111111.....222222222222...33333!  
!111111111111111111.....222222222222...3333!  
!11111111111111111111.....222222222222...3333!  
!1111111111111111111111.....222222222222...333!  
!111111111111111111111111.....222222222222...33!  
!11111111111111111111111111.....222222222222...3!  
!.....1111111111111111111111.....222222222222...!  
!.....1111111111111111111111.....222222222222...!  
!.....1111111111111111111111.....222222222222...!  
!000000.....1111111111111111.....2222222222!  
!0000000000.....1111111111111111.....222222222!  
!000000000000.....1111111111111111.....22222222!  
!00000000000000.....1111111111111111.....2222222!  
!0000000000000000.....1111111111111111.....2222222!  
!000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
!00000000000000000000.....1111111111111111.....222222!  
+-----+

NX= 45 NY= 30 NI= 6 NJ= 6 EPS=0.40  
XINI= 0.00E-01 XEND= 1.00E+00  
YINI= 0.00E-01 YEND= 1.00E+00

де  $a, b$  — числа проміжку  $[0,25; 0,75]$ ;  $g_k = 0, k = \overline{1, 4}; u_0 = 0$ .

Варіант 2:

$f = 0, g_1 = 0,5 - |x_2 - 0,5|, g_k = 0, k = \overline{2, 4}, u_0 = 0$ .

Варіант 3:

$$f = 0, \quad g_1 = x_2(1 - x_2), \quad g_k = 0, \quad k = \overline{2, 4}, \quad u_0 = 0.$$

Варіант 4:

$$f = \begin{cases} 1, & x_1 = a_1, \quad x_2 = a_2; \\ 0, & x_1 \neq a_1, \quad x_2 \neq a_2, \end{cases}$$

де  $a_1, a_2$  — числа проміжку  $(0, 1)$ , які вибираються так, щоб точка  $(a_1, a_2)$  була точкою сітки (наприклад,  $a_1 = a_2 = 0,5$ );  $N_1 = N_2 = 2p + 1$ , де  $p$  — довільне ціле.

Варіант 5:

$$f = \delta(x_1 - a_1)(x_2 - a_2), \quad g_k = 0, \quad k = \overline{1, 4}, \quad u_0 = 0,$$

де числа  $a_1$  і  $a_2$  задаються так, як і в попередньому варіанті,  $\delta(x)$  — дельта-функція Дірака, для апроксимації якої на сітці можна скористатися виразом

$$\delta_{h\alpha}(x_\alpha - a_\alpha) = \begin{cases} h_\alpha^{-1}, & x_\alpha = a_\alpha, \\ 0, & x_\alpha \neq a_\alpha, \end{cases} \quad \alpha = 1, 2.$$

## § 18. Різницева схема для розв'язування одновимірних рівнянь гіперболічного типу

**Постановка задачі.** За допомогою різницевої схеми розв'язати таку початково-крайову задачу:

$$\frac{\partial^2 u}{\partial t^2} = \frac{1}{x^m} \frac{\partial}{\partial x} \left( x^m k \frac{\partial u}{\partial x} \right) - qu + f, \quad a < x < b, \quad t > 0; \quad (1)$$

$$-\alpha_1 k \frac{\partial u}{\partial x} + \beta_1 u = g_1(t), \quad x = a, \quad t > 0; \quad (2)$$

$$\alpha_2 k \frac{\partial u}{\partial x} + \beta_2 u = g_2(t), \quad x = b, \quad t > 0; \quad (3)$$

$$u = u_0(x), \quad \frac{\partial u}{\partial t} = u_1(x), \quad t = 0, \quad a < x < b. \quad (4)$$

Сталі  $\alpha_1, \beta_1, \alpha_2, \beta_2, m = 0, 1, 2$  і неперервні функції  $k(x) \geq c_0 > 0, q(x) \geq 0, f(x, t), g_1(t), g_2(t), u_0(x), u_1(x)$  визначаються варіантом завдання.

**Методичні вказівки.** Ефективним методом розв'язування задачі (1) — (4) є застосування явної різницевої схеми. Вона має другий порядок апроксимації, а умова її стійкості здебільшого необтяжливо обмежує крок сітки по часу  $t$ .

Якщо виходити з сіток по просторовій змінній  $\bar{\omega}_h = \{x_i = ih, i = a + ih, i = 0, 1, \dots, N, h = \frac{b-a}{N}\}$  та по часу  $\bar{\omega}_\tau = \{t_n = n\tau, n = 0, 1, \dots, \tau > 0\}$ , то, коли  $\alpha_1 \neq 0, \alpha_2 \neq 0$  (крайові умови третього

роду), явна різницева схема, побудована за допомогою інтегро-інтерполяційного методу [22, с. 156], матиме вигляд

$$y_{it} = \frac{1}{c} (x^m \bar{k} y_x)_x - qy + f, \quad x \in \bar{\omega}_h, \quad t \in \bar{\omega}_\tau; \quad (5)$$

$$y_{it} = \frac{2\bar{x}_1^m k_1}{hc_0} y_x - \frac{2x_0^m}{hc_0 \alpha_1} (\beta_1 y - g_1(t)) - qy + f, \quad x = a, \quad t \in \bar{\omega}_\tau; \quad (6)$$

$$y_{it} = \frac{2\bar{x}_N^m \bar{k}_N}{hc_N} y_x - \frac{2x_N^m}{hc_N \alpha_2} (\beta_2 y - g_2(t)) - qy + f, \quad x = b, \quad t \in \bar{\omega}_\tau; \quad (7)$$

$$y = u_0(x); \quad y_i = u_1(x); \quad t = 0, \quad x \in \bar{\omega}_h, \quad (8)$$

де  $\bar{x} = \bar{x}_i = x_i - h/2$ ,  $\bar{k} = \bar{k}_i = k(\bar{x}_i)$ ,  $i = 1, 2, \dots, N$ ;

$$c = c_i = \frac{\bar{x}_{i+1}^{m+1} - \bar{x}_i^{m+1}}{(m+1)(\bar{x}_{i+1} - \bar{x}_i)}, \quad \bar{x}_0 = x_0 = a, \quad \bar{x}_{N+1} = x_N = b,$$

$$y = y_i^n = y(x_i, t_n);$$

$$y_{it} = \frac{y_i^{n+1} - 2y_i^n + y_i^{n-1}}{\tau^2}, \quad y_i = \frac{y_i^{n+1} - y_i^{n-1}}{2\tau},$$

$$y_{\bar{x}} = \frac{y_i^n - y_{i-1}^n}{h}, \quad y_x = \frac{y_{i+1}^n - y_i^n}{h}.$$

При  $\alpha_1 = 0$  (крайові умови першого роду) рівняння (6) замінюється таким рівнянням:

$$y_0 = y(a, t) = g_1(t)/b_1. \quad (6a)$$

При  $\alpha_2 = 0$  замість (7) маємо рівняння

$$y_N = y(b, t) = g_2(t)/\beta_2. \quad (7a)$$

Зауважимо, що при апроксимації другої початкової умови застосовується центральна різницева похідна, яка дає розв'язок на фіктивному  $(-1)$ -му шарі  $y(x, -\tau)$ . Він може бути виключений використанням рівняння (5) при  $t = 0$  та за умови (8).

Реалізація явної різницевої схеми (5) – (8) проста: для знаходження розв'язку на  $(n+1)$ -му шарі ( $n = 1, 2, \dots$ ) маємо формулу

$$y_i^{n+1} = 2y_i^n - y_i^{n-1} + \tau^2 v_i, \quad i = 0, 1, \dots, N, \quad (9)$$

де  $v_i$  – значення правої частини рівняння (5), коли  $i = 1, 2, \dots, N - 1$ , рівняння (6), коли  $i = 0$  та рівняння (7), коли  $i = N$ .

Розв'язок на першому шарі дає формула

$$y_i^1 = y_i^0 + \tau u_1(x_i) + \frac{\tau^2}{2} v_i, \quad i = 0, 1, \dots, N, \quad (10)$$

де  $v_i$  має той самий зміст, що і в (9).

Умова стійкості побудованої різницевої схеми записується у вигляді

$$\tau \leq \frac{h}{\left(KA + \frac{h^2}{4} Q\right)^{1/2}}, \quad (11)$$

де  $K = \max k(x)$ ;  $Q = \max q(x)$ ;  $A = \frac{x_0 + h^2/4}{x_0 + h^2/6}$ .

**Приклад використання підпрограм.** Побудовану вище явну різницеву схему реалізовано в підпрограмі GIP1, яка дає змогу визначити розв'язок задачі на новому,  $(n + 1)$ -му шарі ( $n = 1, 2, \dots$ ) за відомими його значеннями на  $(n - 1)$ - та  $n$ -му шарах або на першому шарі за відомим значенням розв'язку на нульовому шарі. Тому в головній програмі необхідно передбачити звернення до цієї підпрограми для всіх шарів, включаючи кінцевий момент часу.

Опис параметрів підпрограми наведено в коментарях до неї. Зупинимось лише на особливостях застосування деяких з них.

Формальні параметри T, Y0, Y1 змінюються в підпрограмі GIP1. Вони характеризують відповідно старий момент часу  $t_n$ , значення розв'язку на  $(n - 1)$ - та  $n$ -му шарах при зверненні до неї, а також новий момент часу  $t_{n+1}$ , значення розв'язку на  $n$ - та  $(n + 1)$ -му шарах при виході з підпрограми. У початковий момент часу, тобто при  $T = 0$ , початкове значення розв'язку повинно розміщатися в масиві Y1, а масив Y0 може бути довільним (наприклад, йому можуть бути надані нульові значення).

Параметр M дає змогу застосувати одну підпрограму GIP1 для розв'язання задач з коливань у суцільному середовищі чи інших задач, які мають плоску ( $M = 0$ ), циліндричну ( $M = 1$ ) або сферичну ( $M = 2$ ) симетрію.

Користувач повинен передбачити також обчислення значень функцій  $u_1(x)$ ,  $g_1(t)$ ,  $g_2(t)$ ,  $k(x)$ ,  $q(x)$ ,  $f(x, t)$  за допомогою підпрограм-функцій U1(X), G1(T), G2(T), AK(X), Q(X), F(X, T).

▼ Використання підпрограми GIP1 проілюструємо на прикладі знаходження розв'язку такої задачі:

$$\frac{\partial^2 u}{\partial t^2} = \frac{1}{r} \frac{\partial}{\partial r} \left( r \frac{\partial u}{\partial r} \right) - \frac{u}{r^2}, \quad 0,25 < r < 1, \quad 0 < t \leq 1; \quad (12)$$

$$u = 0, \quad r = 0,25; \quad (13)$$

$$\frac{\partial u}{\partial r} = \sin(2\pi t), \quad r = 1; \quad (14)$$

$$u = \frac{\partial u}{\partial t} = 0, \quad t = 0. \quad (15)$$

Головна програма для визначення розв'язку у вузлах рівномірної сітки та виведення знайдених результатів через інтервал часу  $TP = 0,5$  має вигляд:

```

CC
C ГОЛОВНА ПРОГРАМА ДО ПІДПРОГРАМИ GIP1:
C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ЗАДАЧІ (12) — (15)
CC
 DIMENSION Y0(101),Y1(101),U(1100),V(110)
 COMMON /C1/N,TAU,H
 COMMON /C2/M,A,B,AL1,AL2,BE1,BE2
C ЗАДАВАННЯ ПАРАМЕТРІВ ЗАДАЧІ

```

```

DATA M/1,A,B/.25,1./
DATA AL1,AL2,BE1,BE2/0.,1.,1.,0./
DATA T,TEND/0.,1./
C ЗАДАВАННЯ ПАРАМЕТРІВ СХЕМИ
DATA TAU,TP/.01,.5/
C ВВЕДЕННЯ ІНШИХ ПАРАМЕТРІВ З ТЕРМІНАЛА
TYPE 100
АССЕРТ *,NW
TYPE 101
АССЕРТ *,N
H=(B-A)/FLOAT(N)
NP1=N+1
X=A
DO 10 I=1,NP1
 Y1(I)=U0(X)
 Y0(I)=0.
10 X=X+H
 TC=0.
C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ НА ВСІХ ШАРАХ
20 CALL GIP1(T,Y0,Y1)
 TYPE 102,T
 TC=TC+TAU
 K=K+1
C ПЕРЕВІРКА НЕОБХІДНОСТІ ВИВЕДЕННЯ РОЗВ'ЯЗКУ
30 IF(TC-TP+.1*TAU) 50,50,30
 TC=0.
 WRITE(NW,103) T,(Y1(I),I=1,NP1)
C ВИВЕДЕННЯ ГРАФІКА РОЗВ'ЯЗКУ
X=A
DO 40 I=1,NP1
 V(I)=X
 V(NP1+I)=Y0(I)
40 X=X+H
 CALL GRAPH(NW,K,NP1,2,2*NP1,V)
50 CONTINUE
C ПЕРЕВІРКА ЗАКІНЧЕННЯ ОБЧИСЛЕНЬ
IF(T-TEND+.1*TAU) 20,20,60
60 STOP
100 FORMAT(1H$, 'НОМЕР ПРИСТРОЮ ВИВЕД. NW=')
101 FORMAT(1H$, 'КІЛЬКІСТЬ ІНТЕРВАЛІВ ПО X',
 * '(N <=100)N=')
102 FORMAT(1X, 'T=', 1PE12.4)
103 FORMAT(10X, 'РОЗВ'ЯЗОК У МОМЕНТ ЧАСУ T=',
 * E12.5, 21/(2X, 5E12.4))
 END
C ПІДПРОГРАМИ-ФУНКЦІЇ
FUNCTION U0(X)
U0=0.
RETURN
END

FUNCTION U1(X)
U1=0.
RETURN
END

```

FUNCTION G1(T)

G1=0.  
RETURN  
END

FUNCTION G2(T)  
DATA P12/6.28306/  
G2=SIN(P12\*T)  
RETURN  
END

FUNCTION AK(X)  
AK=1.  
RETURN  
END

FUNCTION Q(X)  
Q=1./(X\*X)  
RETURN  
END

FUNCTION F(X,T)  
F=0.  
RETURN  
END

Після виконання цієї програми, якщо  $NW = 6$  і  $N = 25$ , принтер роздруковує два масиви розв'язку з двома графіками. Як приклад нижче наведено роздруковку другого розв'язку.

РОЗВ'ЯЗОК У МОМЕНТ ЧАСУ  $T = 0.10000E+01$

|            |            |            |            |
|------------|------------|------------|------------|
| 0.0000E+00 | 0.1053E+00 | 0.1984E+00 | 0.2784E+00 |
| 0.3445E+00 | 0.3950E+00 | 0.4271E+00 | 0.4425E+00 |
| 0.4460E+00 | 0.4394E+00 | 0.4244E+00 | 0.4040E+00 |
| 0.3780E+00 | 0.3479E+00 | 0.3155E+00 | 0.2804E+00 |
| 0.2452E+00 | 0.2100E+00 | 0.1762E+00 | 0.1452E+00 |
| 0.1170E+00 | 0.9345E-01 | 0.7406E-01 | 0.6047E-01 |
| 0.5180E-01 | 0.4928E-01 |            |            |

Відповідний графік зображено на с. 192.

### Запитання для самоперевірки

1. Який шаблон має явна різницева схема (5) — (8)? 2. Який порядок апроксимації має явна різницева схема (5) — (8)? 3. Як записується умова стійкості явної різницевої схеми для рівняння коливання струни  $\frac{\partial^2 u}{\partial t^2} = a^2 \frac{\partial^2 u}{\partial x^2}$ ? 4. Чому явні різницеві схеми широко застосовуються при розв'язуванні початково-крайових задач для гіперболічних рівнянь?

### Контрольні завдання

1. Розв'язати таку задачу про коливання однорідної струни під дією періодичної сили на лівому її кінці, якщо правий кінець струни закріплено:

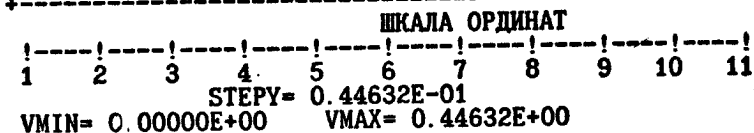
$$\frac{\partial^2 u}{\partial t^2} = \frac{\partial^2 u}{\partial x^2}, \quad 0 < x < 1, \quad 0 < t < T;$$



ФУНКЦІЇ: 1 - \*  
АБСЦИСИ

ГРАФІК 100

0.2500E+00!\*  
0.2800E+00!:  
0.3100E+00!:  
0.3400E+00!:  
0.3700E+00!:  
0.4000E+00!:  
0.4300E+00!:  
0.4600E+00!:  
0.4900E+00!:  
0.5200E+00!:  
0.5500E+00!:  
0.5800E+00!:  
0.6100E+00!:  
0.6400E+00!:  
0.6700E+00!:  
0.7000E+00!:  
0.7300E+00!:  
0.7600E+00!:  
0.7900E+00!:  
0.8200E+00!:  
0.8500E+00!:  
0.8800E+00!:  
0.9100E+00!:  
0.9400E+00!:  
0.9700E+00!:  
0.1000E+01!:



$$-\frac{\partial u}{\partial x} = \sin(\pi \omega t), \quad x=0, \quad u=0, \quad x=1;$$

$$u = \frac{\partial u}{\partial t} = 0, \quad 0 \leq x \leq 1.$$

Покласти  $\omega = 1 \div 3$ ,  $T = 1 \div 3$ .

Змінюючи крайові умови (лівий кінець струни закріплено, а збуджуюча сила діє при  $x=1$ ; на кінцях струни відбувається її зміщення, наприклад за законом  $u = \sin(\omega t)$ ), викладач може задати той чи інший варіант завдання.

2. Коливання порожнистого п'єзокерамічного циліндра під дією електричного навантаження можна описати такою початково-крайовою задачею:

$$\frac{\partial^2 u}{\partial t^2} = \frac{1}{r} \frac{\partial}{\partial r} (r \sigma_r) - \frac{\sigma_\varphi}{r}; \quad r_0 < r < 1, \quad 0 < t \leq T;$$

$$\sigma_r = 0, \quad r = r_0, 1;$$

$$u = \frac{\partial u}{\partial t} = 0, \quad t = 0.$$

де  $\sigma_r = c_{11}\varepsilon_r + c_{12}\varepsilon_\varphi - e_1 E_\varphi$ ;  $\sigma_\varphi = c_{12}\varepsilon_r + c_{22}\varepsilon_\varphi - e_2 E_\varphi$ ;  $\varepsilon_r = \frac{\partial u}{\partial r}$ ;  $\varepsilon_\varphi = \frac{u}{r}$ ;  $E_\varphi = -V(t)$ . Тут  $u$  — зміщення циліндра вздовж його радіуса  $r$ ;  $\sigma_r$ ,  $\sigma_\varphi$ ,  $\varepsilon_r$  — компоненти тензорів напруження та деформації;  $E_\varphi$  — компонента вектора напруженості електричного поля;  $V(t)$  — електрична напруга.

При  $c_{11} = 1$ ;  $c_{22} = 0,8$ ;  $c_{12} = 0,4$ ;  $e_1 = 0,6$ ;  $e_2 = 0,3$  задача має один числовий параметр  $r_0$  і функцію  $V(t)$ .

Знайти розв'язок задачі, якщо  $r = 0,1$  (0,5; 0,9);  $V(t) = 1$  і  $V(t) = \sin(\omega t)$ ,  $1 \leq \omega \leq 3$ .

Побудувати графік зміщень зовнішньої поверхні циліндра як функції часу  $t$ .

ДОДАТОК

```

SUBROUTINE GRAPH(NW, K, N, M, MN, V)
CC
С ПІДПРОГРАМА ПОБУДОВИ ГРАФІКА КІЛЬКОХ ФУНКЦІЙ С
С ОДНІЄЇ ЗМІННОЇ С
С ВХІДНІ ПАРАМЕТРИ: С
С ----- С
С NW - НОМЕР ПРИСТРОЮ, НА ЯКИЙ ВИВОДИТЬСЯ С
С ГРАФІК; С
С K - НОМЕР ГРАФІКА; С
С N - КІЛЬКІСТЬ ТОЧОК СІТКИ; С
С M - КІЛЬКІСТЬ ФУНКЦІЙ, ЩО ВИВОДЯТЬСЯ, С
С ПЛЮС 1; С
С MN - M*N; С
С V - ОДНОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ MN, ПЕРШІ С
С N ЙОГО ЕЛЕМЕНТІВ - АБСЦИСИ ТОЧОК СІТКИ, С
С ІНШІ ЕЛЕМЕНТИ - ЗНАЧЕННЯ ФУНКЦІЙ В ЦИХ С
С ТОЧКАХ. С
С ПРИМІТКА: КІЛЬКІСТЬ ФУНКЦІЙ <= 9. С
CC
DIMENSION V(MN)
LOGICAL*1 OUT(52), SYM(9), BLANK, ZERO
DATA SYM /'*, '+', '$', '#', '&', '@', '^', '%', '<' /
DATA BLANK /' ', /, ZERO /':', /
M1=M-1
WRITE(NW, 100) (J, SYM(J), J=1, M1)
WRITE(NW, 101) K
WRITE(NW, 102)
VMIN=V(N+1)
VMAX=VMIN
N2=N+2
DO 10 I=N2, MN
VAR=V(I)
IF (VAR. LT. VMIN) VMIN=VAR
IF (VAR. GT. VMAX) VMAX=VAR
IF (VMAX-VMIN. GT. (1.E-3)*VMAX) GO TO 20
VMAX=VMAX+1.
VMIN=VMIN-1.
20 Z=VMAX-VMIN
STEPY=Z/50.
IF (VMIN. GT. 0. OR. VMAX. LT. 0.) GO TO 30
IZERO=-VMIN/STEPY+1.5
GO TO 40
30 IZERO=52
40 DO 70 I=1, N
X=V(I)
DO 50 J=1, 51

```

```

50 OUT (J) = BLANK
 OUT (IZERO) = ZERO
 DO 60 J=1, M1
 VAR = V(N*J+I)
 NJ = N*J+I
 JVAR = (VAR - VMIN) / STEPY + 1.5
60 OUT (JVAR) = SYM (J)
70 WRITE (NW, 103) X, (OUT (J), J=1, 51)
 WRITE (NW, 102)
 WRITE (NW, 104)
 WRITE (NW, 105)
 WRITE (NW, 106) (I, I=1, 11)
 STEPY = Z/10.
 Y = VMIN + STEPY * (J-1)
 WRITE (NW, 107) STEPY, VMIN, VMAX
 RETURN
100 FORMAT (2X, 'ФУНКЦІЇ:', 9(2X, I1, ' - ', A1))
101 FORMAT (4X, 'АБСЦИСИ', 31X, 'ГРАФІК', X, I5)
102 FORMAT (12X, '+', 51('-', '+'))
103 FORMAT (X, E11.4, '!', 51A1, '!')
104 FORMAT (38X, 'ШКАЛА ОРДИНАТ')
105 FORMAT (13X, '!', 10(4('-', '!')))
106 FORMAT (12X, 11(I2, 3X))
107 FORMAT (26X, 'STEPY=', E12.5/13X, 'VMIN=', E12.5,
* 4X, 'VMAX=', E12.5/)
 END

```

SUBROUTINE GRAFIK(NW, M, N, NM, X, Y, NX, NY)

```

CC
С ПІДПРОГРАМА ПОВБУДОВИ ГРАФІКА ФУНКЦІЙ ОДНІЄЇ ЗМІН-С
С НОЇ, ЗАДАНИХ КООРДИНАТАМИ ТОЧОК НА ПЛОЩИНІ С
С ВХІДНІ ПАРАМЕТРИ: С
С ----- С
С NW - НОМЕР ПРИСТРОЮ, НА ЯКИЙ ВИВОДИТЬСЯ ГРАФІК; С
С M - КІЛЬКІСТЬ ФУНКЦІЙ; С
С N - МАСИВ РОЗМІРНІСТЮ M, K-Й ЕЛЕМЕНТ ЯКОГО Є С
С КІЛЬКІСТЬ ТОЧОК ДЛЯ K-Ї ФУНКЦІЇ; С
С NM - КІЛЬКІСТЬ ТОЧОК ДЛЯ ВСІХ ФУНКЦІЙ; С
С X - МАСИВ РОЗМІРНІСТЮ NM, В ЯКОМУ МІСТЯТЬСЯ С
С ПОСЛІДОВНО АБСЦИСИ ТОЧОК УСІХ ФУНКЦІЙ; С
С Y - МАСИВ РОЗМІРНІСТЮ NM, В ЯКОМУ МІСТЯТЬСЯ С
С ПОСЛІДОВНО ОРДИНАТИ ТОЧОК УСІХ ФУНКЦІЙ; С
С NX - КІЛЬКІСТЬ ПОЗИЦІЙ, ЩО ВИДІЛЯЮТЬСЯ ПІД С
С ГРАФІК НА ПРИСТРОЇ ВИВЕДЕННЯ ПО С
С ГОРИЗОНТАЛІ; С
С NY - КІЛЬКІСТЬ ПОЗИЦІЙ, ЩО ВИДІЛЯЮТЬСЯ ПІД С
С ГРАФІК НА ПРИСТРОЇ ВИВЕДЕННЯ ПО ВЕРТИКАЛІ. С
С ПРИМІТКА: NX<=70. ДЛЯ ДИСПЛЕЯ МОЖНА РЕКОМЕНДУВАТИС С
С NX=34, NY=16; ДЛЯ ПРИНТЕРА -- NX=40, NY=25. С
CC
DIMENSION N(M), X(NM), Y(NM)
LOGICAL*1 SYM(9), BLANK, LIN, SYMX, OUT(71)
DATA SYM/'*', '+', '$', '#', '&', '@', '?', '%', '<'/

```

```

DATA BLANK/' ', LIN/'-' / SYMX/'X' /
NX1=NX+1
XMAX=X(1)
XMIN=X(1)
YMAX=Y(1)
YMIN=Y(1)
DO 10 I=1, NM
 IF(X(I).GT.XMAX) XMAX=X(I)
 IF(X(I).LT.XMIN) XMIN=X(I)
 IF(Y(I).GT.YMAX) YMAX=Y(I)
 IF(Y(I).LT.YMIN) YMIN=Y(I)
10 CONTINUE
 HX=(XMAX-XMIN)/FLOAT(NX-1)
 HY=(YMAX-YMIN)/FLOAT(NY-1)
 RH=HX/HY
 IF(HX.GT.1E-5.OR.HY.GT.1E-5) GO TO 20
 TYPE 206
 RETURN
20 IF(HX.GT.1E-5) GO TO 30
 XMAX=XMAX+1.
 XMIN=XMIN-1.
30 IF(HY.GT.1E-5) GO TO 40
 YMAX=YMAX+1.
 YMIN=YMIN-1.
40 MINY=NINT(YMIN/HY+.5)
 MAXY=NINT(YMAX/HY+.5)
 WRITE(NW,207) (J,SYM(J),J=1,M)
 WRITE(NW,201)
 DO 90 L1=MINY,MAXY
 L=MAXY+MINY-L1
 DO 50 J=1,NX1
50 OUT(J)=BLANK
 K1=0
 DO 80 K=1,M
 N1=N(K)
 DO 70 I=1,N1
 IK=I+K1
 IY=NINT(Y(IK)/HY+.5)
 IF(L-IY) 70,60,70
60 IX=NINT((X(IK)-XMIN)/HX+.5)+1
 OUT(IX)=SYM(K)
70 CONTINUE
80 K1=K1+N(K)
 WRITE(NW,200) (OUT(J),J=1,NX1)
90 CONTINUE
 DO 100 J=1,NX1
100 OUT(J)=LIN
 WRITE(NW,202) (OUT(J),J=1,NX1)
 DO 110 J=1,NX
110 OUT(J)=BLANK
 OUT(NX1)=SYMX
 WRITE(NW,205) (OUT(J),J=1,NX1)
 WRITE(NW,203) XMAX,YMAX,XMIN,YMIN,HX,HY,RH

```

```

RETURN
200 FORMAT (5X, '!', 71A1)
201 FORMAT (3X, 'Y !')
202 FORMAT (5X, '+', 71A1)
203 FORMAT (5X, 'XMAX=', E11.4, 2X, 'YMAX=', E11.4/
* 5X, 'XMIN=', E11.4, 2X, 'YMIN=', E11.4/
* 7X, 'HX=', E11.4, 4X, 'HY=', E11.4/
* 12X, 'HX/HY=', E9.2)
204 FORMAT (3X, 'Y ' '!')
205 FORMAT (6X, 71A1)
206 FORMAT (5X, 'ПОМИЛКА ЗАДАВАННЯ МАСИВІВ X I Y')
207 FORMAT (5X, 'ФУНКЦІІ:', 9(2X, I1, ' - ', A1)/)
END

```

```

FUNCTION NINT(X)
NINT=INT(X)
IF(X.LT.0.) NINT=NINT-1
RETURN
END

```

```

SUBROUTINE LINUR(NW, NX, NY, N, EPS, X, Y,
* NI, NJ, NDIM, U)

```

```

CC

```

```

C ПІДПРОГРАМА ПОБУДОВИ ЛІНІЙ РІВНЯ ФУНКЦІЇ ДВОХ C
C ЗМІННИХ, ЗАДАНОЇ ЗНАЧЕННЯМИ У ВУЗЛАХ C
C НЕРІВНОМІРНОЇ ПРЯМОКУТНОЇ СІТКИ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C

```

```

C NW - НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ ЛІНІЙ РІВНЯ; C
C NX - КІЛЬКІСТЬ СИМВОЛІВ ДЛЯ ВІДОБРАЖЕННЯ C
C РЕЗУЛЬТАТУ НА ПРИСТРОЇ ВИВЕДЕННЯ ПО C
C ДОВЖИНІ (1<NX<=70); C
C NY - КІЛЬКІСТЬ СИМВОЛІВ ДЛЯ РЕЗУЛЬТАТУ ПО C
C ВИСОТІ (NY>1); C
C N - КІЛЬКІСТЬ ЛІНІЙ РІВНЯ (0<N<=10); C
C EPS - ШИРИНА ЛІНІЙ (СМУГ) РІВНЯ (0<EPS<=.5); C
C X - ОДНОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ NI АБС- C
C ЦИС ВУЗЛІВ СІТКИ; C
C Y - ОДНОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ NJ ОРДИ- C
C НАТ СІТКИ; C
C NI - КІЛЬКІСТЬ ТОЧОК СІТКИ ПО ОСІ АБСЦИС; C
C NJ - КІЛЬКІСТЬ ТОЧОК СІТКИ ПО ОСІ ОРДИНАТ; C
C NDIM- ЗАЯВЛЕНА В ГОЛОВНІЙ ПРОГРАМІ РЯДКОВА C
C РОЗМІРНІСТЬ МАСИВУ U (NDIM>=NI); C
C U - ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ (NDIM, NJ), C
C ЩО МІСТИТЬ ЗНАЧЕННЯ ФУНКЦІЇ У ВУЗЛАХ C
C СІТКИ. C

```

```

CC

```

```

DIMENSION X(NI), Y(NJ), U(NDIM, NJ)
LOGICAL*1 SYM(10), BLANK, OUT(72), LIN, PLUS, PRAV
DATA SYM/'0', '1', '2', '3', '4', '5', '6', '7', '8', '9',
DATA BLANK/' ', LIN/'-', PLUS/'+', PRAV/'!'/
HX=(X(NI)-X(1))/FLOAT(NX-1)

```

```

HY=(Y(NJ)-Y(1))/FLOAT(NY-1)
UMIN=U(1,1)
UMAX=UMIN
DO 10 J=1,NJ
DO 10 I=1,NI
 IF(U(I,J).LT.UMIN) UMIN=U(I,J)
 IF(U(I,J).GT.UMAX) UMAX=U(I,J)
CONTINUE
P=(UMAX-UMIN)/FLOAT(N-1)
EPS1=EPS*P
WRITE(NW,200)
V=UMIN
DO 20 K=1,N
 V1=V-EPS1
 V2=V+EPS1
 IF(K.EQ.1) V1=V
 IF(K.EQ.N) V2=V
 K1=K-1
 WRITE(NW,210) K1,V1,V2
 V=V+P
NX1=NX+1
NX2=NX+2
DO 30 I=1,NX
 OUT(I+1)=LIN
OUT(1)=PLUS
OUT(NX2)=PLUS
WRITE(NW,220) (OUT(I),I=1,NX)
JO=NJ
YY=Y(NJ)
DO 40 M=1,NY
 J=NY+1-M
 IF(YY-1.E-3*HY-Y(JO-1))50,70,70
50 JO=JO-1
 IF(JO.EQ.1) GO TO 60
 GO TO 40
60 JO=2
70 YH=Y(JO-1)-Y(JO)
 RY=(YY-Y(JO))/YH
 IO=1
 XX=X(1)
DO 80 I=1,NX
 OUT(I)=BLANK
OUT(NX1)=PRAW
DO 140 I=1,NX
 IF(XX+1.E-3*HX-X(IO+1))120,100,100
100 IO=IO+1
 IF(IO.EQ.NI) GO TO 110
 GO TO 90
110 IO=NI-1
120 XH=X(IO+1)-X(IO)
 RX=(XX-X(IO))/XH
 V=U(IO,JO)*(1.-RX)*(1.-RY)+U(IO+1,JO)*
 RX*(1.-RY)+U(IO,JO-1)*(1.-RX)*RY+

```

\*

```

 U(IO+1, JO-1)*RX*RY
 DO 130 K=1, N
 R=ABS(V-UMIN-P*FLOAT(K-1))
 IF (R. LT. EPS1) OUT(I)=SYM(K)
 CONTINUE
130
140 XX=XX+HX
 WRITE (NW, 230) (OUT(I), I=1, NX1)
 YY=YY-HY
150 DO 160 I=1, NX
 OUT(I+1)=LIN
160 OUT(1)=PLUS
 OUT(NX2)=PLUS
 WRITE (NW, 220) (OUT(I), I=1, NX2)
 WRITE (NW, 240) NX, NY, NI, NJ, EPS, X(1), X(NI),
 * Y(1), Y(NJ)
200 FORMAT(4X, 'СИМВОЛ', 4X, 'ДІАПАЗОН ЗМІНИ',
 * 'ФУНКЦІЇ' /)
210 FORMAT(6X, I1, 2(5X, E12.5))
220 FORMAT(4X, 73A1)
230 FORMAT(4X, '!', 71A1)
240 FORMAT(6X, 'NX=', I3, ' NY=', I3, ' NI=', I3,
 * ' NJ=', I3, ' EPS=', F4.2/6X, 'XINI=',
 * 1PE9.2, ' XEND=', 1PE9.2/
 * 6X, 'YINI=', 1PE9.2, ' YEND=', 1PE9.2)
 RETURN
 END

```

```

 SUBROUTINE BISECT(A, B, EPS, F, X, K)
CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ НЕЛІНІЙНОГО РІВНЯННЯ C
C МЕТОДОМ ДІЛЕННЯ НАВПІЛ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C A, B-ЛІВИЙ І ПРАВИЙ КІНЦІ ВІДРІЗКА, НА ЯКОМУ C
C ЗНАХОДИТЬСЯ ШУКАНИЙ КОРІНЬ РІВНЯННЯ; C
C EPS-ПОХИБКА ВИЗНАЧЕННЯ КОРЕНЯ; C
C F -ІМ'Я ПІДПРОГРАМИ-ФУНКЦІЇ, ЩО ОБЧИСЛЮЄ C
C ЗНАЧЕННЯ ЛІВОЇ ЧАСТИНИ РІВНЯННЯ. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C X -НАБЛИЖЕНЕ ЗНАЧЕННЯ КОРЕНЯ; C
C K -КІЛЬКІСТЬ ІТЕРАЦІЙ. C
CC
 K=0
 S=1.
 R=F(A)
10 X=(A+B)*0.5
 Y=F(X)
 IF((Y.EQ.0.).OR.((B-A).LE.(2.*EPS))) GO TO 40
 K=K+1
 S=SIGN(S, Y)*SIGN(S, R)
 IF(S)20,40,30

```



```
20 B=X
GO TO 10
30 A=X
R=Y
GO TO 10
40 RETURN
END
```

```
SUBROUTINE SITER(X0, EPS, K, F, Q)
CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ НЕЛІНІЙНОГО РІВНЯННЯ C
C МЕТОДОМ ПРОСТОЇ ІТЕРАЦІЇ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C X0 -ПОЧАТКОВЕ НАБЛИЖЕННЯ КОРЕНЯ; C
C EPS-ПОХИБКА ВИЗНАЧЕННЯ КОРЕНЯ; C
C Q -ДОПОМІЖНА ВЕЛИЧИНА, ЩО ВИЗНАЧАЄ C
C ЗБІЖНІСТЬ МЕТОДУ; C
C F -ІМ'Я ПІДПРОГРАМИ-ФУНКЦІЇ, ЩО ОБЧИСЛЮЄ C
C ЗНАЧЕННЯ ПРАВОЇ ЧАСТИНИ ПЕРЕТВОРеноГО C
C РІВНЯННЯ. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C X0 -НАБЛИЖЕНЕ ЗНАЧЕННЯ КОРЕНЯ; C
C K -КІЛЬКІСТЬ ВИКОНАНИХ ІТЕРАЦІЙ. C
CC
K=0
X=X0
EPS1=EPS
IF(Q.GT.0)EPS1=(1.-Q)/Q*EPS
10 Y=F(X)
K=K+1
IF(ABS(Y-X).LE.EPS1) GO TO 20
X=Y
GO TO 10
20 X0=Y
RETURN
END
```

```
SUBROUTINE NEWTON(X0, N, EPS, IH, K, F, F1)
CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ НЕЛІНІЙНОГО РІВНЯННЯ C
C МЕТОДОМ НЬЮТОНА C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C X0 -ПОЧАТКОВЕ НАБЛИЖЕННЯ КОРЕНЯ; C
C N -МАКСИМАЛЬНО ДОПУСТИМА КІЛЬКІСТЬ ІТЕРАЦІЙ; C
C EPS-ЧИСЛО, ЩО ВИЗНАЧАЄ ЗАКІНЧЕННЯ ІТЕРА- C
C ЦІЙНОГО ПРОЦЕСУ; C
C F, F1-ІМЕНА ПІДПРОГРАМ-ФУНКЦІЙ, ЩО ОБЧИС- C
C ЛЮТЬ ЗНАЧЕННЯ ЛІВОЇ ЧАСТИНИ РІВНЯН- C
C НЯ ТА ЇЇ ПОХІДНИХ. C
```

```

C ВИХІДНІ ПАРАМЕТРИ:
C -----
C XO -НАБЛИЖЕНЕ ЗНАЧЕННЯ КОРЕНЯ;
C K -КІЛЬКІСТЬ ІТЕРАЦІЙ;
C IH -ПОКАЗНИК ПРИЧИНИ ЗАКІНЧЕННЯ ОБЧИСЛЕНЬ:
C ЯКЩО IH=0, ТО КОРІНЬ РІВНЯННЯ ЗНАЙДЕНО
C З ЗАДАНОЮ ТОЧНІСТЮ, ЯКЩО IH=1, ТО ІТЕ-
C РАЦІЇ НЕ ЗБІГАЮТЬСЯ ДО КОРЕНЯ РІВНЯННЯ.
C ЯКЩО IH=2, ТО КІЛЬКІСТЬ ВИКОНАНИХ ІТЕРАЦІЙ
C ПЕРЕВЕРШИЛА МАКСИМАЛЬНО ДОПУСТИМЕ ЧИСЛО N.
C СС

```

```

DELO= 1. E12
XS=XO
K=1
10 Y=F1(XS)
 XN=XS-F(XS)/Y
 DEL=ABS(XN-XS)
 IF(DEL.LT.EPS) GO TO 40
 IF(DEL.LT.DELO) GO TO 20
 IH=1
 GO TO 50
20 IF(K.GT.N) GO TO 30
 K=K+1
 XS=XN
 DELO=DEL
 GO TO 10
30 IH=2
 GO TO 50
40 XO=XN
 IH=0
50 RETURN
 END

```

SUBROUTINE ZEROIN(AX, BX, F, TOL, Z)

```

C СС
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ НЕЛІНІЙНОГО РІВНЯННЯ,
C МЕТОДОМ, ЩО ПОЄДНУЄ ДІЛЕННЯ НАВПІЛ, ОБЕРНЕНУ
C КВАДРАТИЧНУ ІНТЕРПОЛЯЦІЮ ТА ЛІНІЙНУ ІНТЕРПОЛЯ-
C ЦІЮ (МЕТОД СІЧНИХ)
C ВИХІДНІ ПАРАМЕТРИ:
C -----
C AX -ЛІВИЙ КІНЕЦЬ ВХІДНОГО ІНТЕРВАЛУ;
C BX -ПРАВІЙ КІНЕЦЬ ВХІДНОГО ІНТЕРВАЛУ;
C F -ПІДПРОГРАМА-ФУНКЦІЯ, ЩО ОБЧИСЛЮЄ
C ЗНАЧЕННЯ ЛІВОЇ ЧАСТИНИ РІВНЯННЯ;
C TOL-БАЖАНА ДОВЖИНА ІНТЕРВАЛУ НЕВИЗНАЧЕН-
C НОСТІ КІНЦЕВОГО РЕЗУЛЬТАТУ.
C ВИХІДНІ ПАРАМЕТРИ:
C -----
C Z -НАБЛИЖЕНЕ ЗНАЧЕННЯ КОРЕНЯ.
C СС
C REAL AX, BX, F, TOL
C REAL A, B, C, D, E, EPS, FA, FB, FC, TOL1, XM, P, Q, R, S

```

```

C ОБЧИСЛИТИ EPS - ВІДНОСНУ МАШИННУ ТОЧНІСТЬ
 EPS=1.0
10 EPS=EPS/2.0
 TOL1=1.0+EPS
 IF(TOL1.GT.1.0) GO TO 10
C ПРИСВОЄННЯ ПОЧАТКОВИХ ЗНАЧЕНЬ
 A=AX
 B=BX
 FA=F(A)
 FB=F(B)
C ПОЧАТИ ОБЧИСЛЕННЯ
20 C=A
 FC=FA
 D=B-A
 E=D
30 IF(ABS(FC).GE.ABS(FB)) GO TO 40
 A=B
 B=C
 C=A
 FA=FB
 FB=FC
 FC=FA
C ПЕРЕВІРКА ЗБІЖНОСТІ
40 TOL1=2.0*EPS*ABS(B)+0.5*TOL
 XM=.5*(C-B)
 IF(ABS(XM).LE.TOL1) GO TO 90
 IF(FB.EQ.0.0) GO TO 90
C ЧИ ПОТРІБЕН МЕТОД БІСЕКЦІЇ?
 IF(ABS(E).LT.TOL1) GO TO 70
 IF(ABS(FA).LE.ABS(FB)) GO TO 70
C ЧИ МОЖЛИВА КВАДРАТИЧНА ІНТЕРПОЛЯЦІЯ?
 IF(A.NE.C) GO TO 50
C ЛІНІЙНА ІНТЕРПОЛЯЦІЯ
 S=FB/FA
 P=2.0*XM*S
 Q=1.0-S
 GO TO 60
C ОБЕРНЕНА КВАДРАТИЧНА ІНТЕРПОЛЯЦІЯ
50 Q=FA/FC
 R=FB/FC
 S=FB/FA
 P=S*(2.0*XM*Q*(Q-R)-(B-A)*(R-1.0))
 Q=(Q-1.0)*(R-1.0)*(S-1.0)
C ВИБРАТИ ЗНАКИ
60 IF(P.GT.0.0) Q=-Q
 P=ABS(P)
C ЧИ ЗАДОВІЛЬНА ІНТЕРПОЛЯЦІЯ?
 IF((2.0*P).GE.(3.0*XM*Q-ABS(TOL1*Q))) GO TO 70
 IF(P.GE.ABS(0.5*E*Q)) GO TO 70
 E=D
 D=P/Q
 GO TO 80.
C МЕТОД БІСЕКЦІЇ

```

```

70 D=XM
 E=D
C ЗАВЕРШИТИ ОБЧИСЛЕННЯ
80 A=B
 FA=FB
 IF (ABS (D) .GT. TOL1) B=B+D
 IF (ABS (D) .LE. TOL1) B=B+SIGN (TOL1, XM)
 FB=F (B)
 IF ((FB*(FC/ABS (FC))).GT. 0. 0) GO TO 20
 GO TO 30
90 Z=B
 RETURN
 END

```

```

SUBROUTINE DECOMP (NDIM, N, A, COND, IPVT, WORK)
CC
C ПІДПРОГРАМА ТРИКУТНОГО РОЗКЛАДАННЯ ДОВІЛЬНОЇ C
C МАТРИЦІ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C NDIM- РЯДКОВА РОЗМІРНІСТЬ МАТРИЦІ, C
C ОГОЛОШЕНА В ГОЛОВНІЙ ПРОГРАМІ; C
C N - ПОРЯДОК МАТРИЦІ; C
C A - МАТРИЦЯ, ЩО ПОТРЕБУЄ РОЗКЛАДУ. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C A - ТРИКУТНИЙ РОЗКЛАД МАТРИЦІ U=M*A: C
C У ВЕРХНЬОМУ ТРИКУТНИКУ ЗБЕРІГАЄТЬСЯ C
C МАТРИЦЯ U, В НИЖНЬОМУ - МАТРИЦЯ M; C
C IPVT- МАСИВ НОМЕРІВ ВЕДУЧИХ ЕЛЕМЕНТІВ НА C
C К-МУ КРОЦІ; IPVT(N)=(-1)**L, ДЕ L - C
C ЧИСЛО ПЕРЕСТАНОВОК РЯДКІВ; C
C WORK- РОБОЧИЙ МАСИВ РОЗМІРНІСТЮ N, ЙОГО C
C ТРЕБА ОПИСАТИ В ГОЛОВНІЙ ПРОГРАМІ; C
C COND- ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ. C
C ПІДПРОГРАМИ ТА ФУНКЦІЇ, ЩО ВИКЛИКАЮТЬСЯ: C
C SOLVE - П/П РОЗВ'ЯЗУВАННЯ СЛАР A*X=B. C
C ПРИМІТКИ: C
C 1. ЦЯ П/П ЗАПОЗИЧЕНА З КНИГИ ФОРСАЙТ ДЖ., C
C МАЛЬКОЛЬМ М., МОУЛЕР К., "МАШИННЫЕ МЕТОДЫ C
C МАТЕМАТИЧЕСКИХ ВЫЧИСЛЕНИЙ" - М., МИР, 1980. C
C С. 65-70. C
C 2. НА ВИХОДІ З П/П DECOMP МАТРИЦЯ А НЕ C
C ЗБЕРІГАЄТЬСЯ. C
CC
INTEGER N, NDIM
REAL A (NDIM, N), COND, WORK (N)
INTEGER IPVT (N)
REAL EK, T, ANORM, YNORM, ZNORM
INTEGER NM1, I, J, K, KP1, KB, KM1, M
IPVT (N) = 1
IF (N. EQ. 1) GO TO 150
NM1=N-1

```

```

ANORM=0.
DO 20 J=1, N
 T=0.
 DO 10 I=1, N
 T=T+ABS(A(I, J))
 CONTINUE
 IF (T. GT. ANORM) ANORM=T
 CONTINUE
DO 70 K=1, NM1
 KP1=K+1
 M=K
 DO 30 I=KP1, N
 IF (ABS(A(I, K)). GT. ABS(A(M, K))) M=I
 CONTINUE
 IPVT(K)=M
 IF (M. NE. K) IPVT(N)=-IPVT(N)
 T=A(M, K)
 A(M, K)=A(K, K)
 A(K, K)=T
 IF (T. EQ. 0.) GO TO 70
 DO 40 I=KP1, N
 A(I, K)=-A(I, K)/T
 CONTINUE
 DO 60 J=KP1, N
 T=A(M, J)
 A(M, J)=A(K, J)
 A(K, J)=T
 IF (T. EQ. 0.) GO TO 60
 DO 50 I=KP1, N
 A(I, J)=A(I, J)+A(I, K)*T
 CONTINUE
 CONTINUE
CONTINUE
DO 100 K=1, N
 T=0.
 IF (K. EQ. 1) GO TO 90
 KM1=K-1
 DO 80 I=1, KM1
 T=T+A(I, K)*WORK(I)
 CONTINUE
 EK=1.
 IF (T. LT. 0.) EK=-1.
 IF (A(K, K). EQ. 0.) GO TO 160
 WORK(K)=- (EK+T)/A(K, K)
 CONTINUE
DO 120 KB=1, NM1
 K=N-KB
 T=WORK(K)
 KP1=K+1
 DO 110 I=KP1, N
 T=T+A(I, K)*WORK(I)
 CONTINUE
 WORK(K)=T

```

```

M=IPVT(K)
IF (M.EQ.K) GO TO 120
T=WORK(M)
WORK(M)=WORK(K)
WORK(K)=T
120 CONTINUE
 YNORM=0.
 DO 130 I=1,N
 YNORM=YNORM+ABS(WORK(I))
130 CONTINUE
 CALL SOLVE (NDIM,N,A,WORK,IPVT)
 ZNORM=0.
 DO 140 I=1,N
 ZNORM=ZNORM+ABS(WORK(I))
140 CONTINUE
 COND=ANORM*ZNORM/YNORM
 IF (COND.LT.1.) COND=1.
 RETURN
150 COND=1.
 IF (A(1,1).NE.0.) RETURN
160 COND=0.
 RETURN
 END

```

```

SUBROUTINE SOLVE (NDIM,N,A,B,IPVT)
CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ СЛАР $A \cdot X = B$, ДЕ А - C
C МАТРИЦЯ, ПОПЕРЕДНЬО РОЗКЛАДЕНА ЗА П/П DECOMP НА C
C ТРИКУТНІ МНОЖНИКИ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C NDIM- РЯДКОВА РОЗМІРНІСТЬ МАТРИЦІ, ОГОЛОШЕНА C
C В ГОЛОВНІЙ ПРОГРАМІ; C
C N - ПОРЯДОК МАТРИЦІ; C
C A - МАТРИЦЯ А, РОЗКЛАДЕНА ЗА П/П DECOMP НА C
C МНОЖНИКИ; C
C B - МАСИВ ПРАВИХ ЧАСТИН СИСТЕМИ; C
C IPVT- МАСИВ ВЕДУЧИХ ЕЛЕМЕНТІВ, ЗНАЙДЕНИЙ ЗА C
C П/П DECOMP. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C B - МАСИВ РОЗВ'ЯЗКІВ СИСТЕМИ. C
C ПРИМІТКА: C
C ПІДПРОГРАМУ SOLVE ЗАСТОСОВУВАТИ НЕ ТРЕБА, ЯКЩО C
C ЗА П/П DECOMP УСТАНОВЛЕНО, ЩО COND=0. C
CC

```

```

INTEGER N,NDIM,IPVT(N)
REAL A(NDIM,N),B(N)
INTEGER KB,KM1,NM1,KP1,I,K,M
REAL T
IF (N.EQ.1) GO TO 50
NM1=N-1
DO 20 K=1,NM1

```

```

 KP1=K+1
 M=IPVT(K)
 T=B(M)
 B(M)=B(K)
 B(K)=T
 DO 10 I=KP1,N
 B(I)=B(I)+A(I,K)*T
 CONTINUE
10 CONTINUE
20 DO 40 KB=1,NM1
 KM1=N-KB
 K=KM1+1
 B(K)=B(K)/A(K,K)
 T=-B(K)
 DO 30 I=1,KM1
 B(I)=B(I)+A(I,K)*T
 CONTINUE
30 CONTINUE
40 CONTINUE
50 B(1)=B(1)/A(1,1)
 RETURN
 END

```

SUBROUTINE FACTSQ(N, A, D, COND, WORK)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

С ПІДПРОГРАМА ТРИКУТНОГО РОЗКЛАДАННЯ СИМЕТРИЧНОЇ С  
 С МАТРИЦІ: A=S(ТРАНСПОНОВАНА)\*D\*S, ДЕ S-ВЕРХНЯ С  
 С ТРИКУТНА МАТРИЦЯ, D-ДІАГОНАЛЬНА МАТРИЦЯ С  
 С ВХІДНІ ПАРАМЕТРИ: С

- 
- N - ПОРЯДОК МАТРИЦІ;
  - A - ОДНОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ  $N*(N+1)/2$ , СКЛАДЕНИЙ З ЕЛЕМЕНТІВ ВЕРХНЬОГО ТРИКУТНИКА СИМЕТРИЧНОЇ МАТРИЦІ А, РОЗТАШОВАНИХ ПО СТОВПЦЯХ.

ВІХІДНІ ПАРАМЕТРИ:

- 
- A - ВЕРХНЯ ТРИКУТНА МАТРИЦЯ S;
  - D - ОДНОВИМІРНИЙ МАСИВ ДІАГОНАЛЬНИХ ЕЛЕМЕНТІВ МАТРИЦІ D;
  - COND- ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ; ЯКЩО МАТРИЦЯ А ВИРОДЖЕНА -- COND=0;
  - WORK- РОБОЧИЙ МАСИВ РОЗМІРНІСТЮ N.
- ПІДПРОГРАМИ ТА ФУНКЦІЇ, ЩО ВИКЛИКАЮТЬСЯ:
- RESQ- П/П РОЗВ'ЯЗУВАННЯ СЛАР  $A*X=B$  З СИМЕТРИЧНОЮ МАТРИЦЕЮ А.

ПРИМІТКА:

НА ВИХОДІ З П/П FACTSQ МАСИВ А, В ЯКОМУ ЗАДАЄТЬСЯ ВЕРХНІЙ ТРИКУТНИК МАТРИЦІ А, НЕ ЗБЕРІГАЄТЬСЯ.

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

DIMENSION A(1), D(N), WORK(N)

DOUBLE PRECISION P, T

С ОБЧИСЛЕННЯ ПЕРШОЇ НОРМИ МАТРИЦІ А

```

ANORM=0.
J1=0
DO 30 J=1, N
 R=0.0
 JM1=J-1
 DO 10 I=1, JM1
 R=R+ABS(A(I+J1))
10 CONTINUE
 I1=J1
 DO 20 I=J, N
 R=R+ABS(A(I1+J))
20 I1=I1+I
 CONTINUE
 J1=J1+J
 IF(R. GT. ANORM) ANORM=R
30 CONTINUE
C ТРИКУТНЕ РОЗКЛАДАННЯ A=S(ТРАНСПОНОВАНА)*D*S
 COND=0.
 IO=0
 DO 100 I=1, N
 IO=IO+I
 I2=IO
 IM1=I-1
 DO 100 J=I, N
 T=0. DO
40 IF(IM1) 40, 60, 40
 DO 50 K=1, IM1
 KO=IO-K
 K2=I2-K
 T=T+DBLE(A(KO))*DBLE(A(K2))*
50 *
 DBLE(D(K))
 CONTINUE
60 T=DBLE(A(I2))-T
 IF(J-I) 90, 70, 80
70 IF(T) 80, 180, 80
80 P=DSQRT(DABS(T))
 D(I)=SIGN(1., T)
 A(IO)=P
 P=D(I)/P
 GOTO 100
90 A(I2)=T*P
100 I2=I2+J
C РОЗВ'ЗУВАННЯ СИСТЕМИ S(ТРАНСПОНОВАНА)*D*W=E
 K1=0
 DO 130 K=1, N
 T=0. DO
 IF(K. EQ. 1) GO TO 120
 KM1=K-1
 DO 110 I=1, KM1
 T=T+DBLE(A(K1+I))*D(I)*WORK(I)
110 CONTINUE
120 EK=1.
 IF(T. LT. 0. DO) EK=-1.

```



```

 K1=K1+K
 WORK(K)=- (EK+T)/(A(K1)*D(K))
130 CONTINUE
C РОЗВ'ЯЗУВАННЯ СИСТЕМИ S*Y=W
 NN=N*(N+1)/2
 WORK(N)=WORK(N)/A(NN)
 DO 150 J=2,N
 I=N+1-J
 IP1=I+1
 IO=I*IP1/2
 T=0. DO
 DO 140 K=IP1,N
 IK=I+K*(K-1)/2
 T=T+DBLE(A(IK))*WORK(K)
140 CONTINUE
 WORK(I)=(WORK(I)-T)/A(IO)
150 CONTINUE
C ОБЧИСЛЕННЯ НОРМИ Y
 YNORM=0.
 DO 160 I=1,N
 YNORM=YNORM+ABS(WORK(I))
160 CONTINUE
C РОЗВ'ЯЗУВАННЯ СИСТЕМИ A*Z=Y ТА ОБЧИСЛЕННЯ НОРМИ Z
 CALL RESQ(N, A, D, WORK)
 ZNORM=0.
 DO 170 I=1,N
 ZNORM=ZNORM+ABS(WORK(I))
170 CONTINUE
C ОЦІНЮВАННЯ ЧИСЛА ОБУМОВЛЕНОСТІ
 COND=ANORM*ZNORM/YNORM
 IF (COND.LT.1.) COND=1.
 RETURN
180 COND=0.
 RETURN
 END

```

SUBROUTINE RESQ(N, A, D, B)

```

CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ СЛАР A*X=B, ДЕ A-МАТРИЦЯ, С
C РОЗКЛАДЕНА П/П FACTSQ НА ТРИКУТНІ МНОЖНИКИ
C A=S (ТРАНСПОНОВАНА)*D*S
C ВХІДНІ ПАРАМЕТРИ:
C -----
C N - РОЗМІРНІСТЬ МАТРИЦІ;
C A - ВЕРХНЯ ТРИКУТНА МАТРИЦЯ S,
C ЗНАЙДЕНА ЗА П/П FACTSQ;
C D - МАСИВ ДІАГОНАЛЬНИХ ЕЛЕМЕНТІВ МАТРИЦІ D;
C B - МАСИВ ПРАВИХ ЧАСТИН СИСТЕМИ.
C ВИХІДНІ ПАРАМЕТРИ:
C -----
C B - МАСИВ РОЗВ'ЯЗКІВ СИСТЕМИ.
C ПРИМІТКА:
C ПІДПРОГРАМУ RESQ ЗАСТОСОВУВАТИ НЕ ТРЕБА, ЯКЩО

```



```

REAL X(N), Y(N)
IER=0
K=0
10 CONTINUE
CALL F(N, X, Y)
S=0.
DO 20 J=1, N
20 S=S+(Y(J)-X(J))**2
S=SQRT(S)
K=K+1
IF (K. GT. KO) GO TO 40
DO 30 I=1, N
30 X(I)=Y(I)
IF (S. GT. EPS) GO TO 10
RETURN
40 IER=-1
RETURN
END

```

```

SUBROUTINE NEWTS (N, X, A, F, G, EPS, K, KO)
CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ СИСТЕМИ N НЕЛІНІЙНИХ C
C РІВНЯНЬ З N НЕВІДОМИМИ МЕТОДОМ НЬЮТОНА C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C N-ПОРЯДОК СИСТЕМИ РІВНЯНЬ; C
C X-ВЕКТОР ПОЧАТКОВОГО НАБЛИЖЕННЯ; C
C A-РОБОЧИЙ МАСИВ, РОЗМІРНІСТЮ (N*N); C
C F, G-ІМЕНА ПІДПРОГРАМ-ФУНКЦІЙ, ЩО ОПИСУЮТЬ C
C СИСТЕМУ РІВНЯНЬ ТА ЇЇ МАТРИЦЮ ЯКОБІ; C
C EPS-ЗАДАНА ТОЧНІСТЬ РОЗВ'ЯЗКУ; C
C KO-МАКСИМАЛЬНО ДОПУСТИМА КІЛЬКІСТЬ ІТЕРАЦІЙ. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C X-НАБЛИЖЕНИЙ РОЗВ'ЯЗОК; C
C K-КІЛЬКІСТЬ ВИКОНАНИХ ІТЕРАЦІЙ. C
C ПІДПРОГРАМИ, ЩО ВИКЛИКАЮТЬСЯ: C
C DECOMP-П/П ТРИКУТНОГО РОЗКЛАДАННЯ МАТРИЦІ; C
C SOLVE-П/П РОЗВ'ЯЗУВАННЯ СЛАР. C
CC
REAL X(N), Y(10), X1(10), IPVT(10), A(N, N), WORK(10)
K=0
10 CONTINUE
C ФОРМУВАННЯ СЛАР З МАТРИЦЕЮ ЯКОБІ
CALL F(N, X, Y)
CALL G(N, X, A)
DO 20 J=1, N
20 Y(J)=-Y(J)
C РОЗВ'ЯЗУВАННЯ СЛАР
CALL DECOMP(N, N, A, COND, IPVT, WORK)
CALL SOLVE(N, N, A, Y, IPVT)
DO 30 J=1, N
30 X(J)=X(J)+Y(J)

```



```

C I КІНЦЕВОЇ (ANRMX) НОРМ
50 ANORM=0.
DO 70 I=1, N
DO 70 J=1, N
IF (I-J) 60, 70, 60
60 IA=I+(J*J-J)/2
ANORM=ANORM+A(IA)*A(IA)
70 CONTINUE
IF (ANORM) 330, 330, 80
80 ANORM=1.414*SQRT(ANORM)
ANRMX=ANORM*RANGE/FLOAT(N)
C УТВОРЕННЯ ІНДИКАТОРА Й ОБЧИСЛЕННЯ THR -ГРАНИЧНОГО
C ЗНАЧЕННЯ ДЛЯ НЕДІАГОНАЛЬНИХ ЕЛЕМЕНТІВ
IND=0
THR=ANORM
90 THR=THR/FLOAT(N)
100 L=1
110 M=L+1
; ОБЧИСЛЕННЯ SIN ТА COS
120 MQ=(M*M-M)/2
LQ=(L*L-L)/2
LM=L+MQ
IF (ABS(A(LM))-THR) 260, 130, 130
130 IND=1
LL=L+LQ
MM=M+MQ
X=0.5*(A(LL)-A(MM))
Y=-A(LM)/SQRT(A(LM)*A(LM)+X*X)
IF (X) 140, 150, 150
140 Y=-Y
150 SINX=Y/SQRT(2.0*(1.0+(SQRT(1.0-Y*Y))))
SINX2=SINX*SINX
COSX=SQRT(1.0-SINX2)
COSX2=COSX*COSX
SINCS=SINX*COSX
C ПЕРЕСТАНОВКА СТОВПЦІВ L I M
ILQ=N*(L-1)
IMQ=N*(M-1)
DO 250 J=1, N
IQ=(I*I-I)/2
IF (I-L) 160, 230, 160
IF (I-M) 170, 230, 180
160 IM=I+MQ
GO TO 190
180 IM=M+IQ
190 IF (I-L) 200, 210, 210
200 IL=I+LQ
GO TO 220
210 IL=L+IQ
220 X=A(IL)*COSX-A(IM)*SINX
A(IM)=A(IL)*SINX+A(IM)*COSX
A(IL)=X
230 IF (MV-1) 240, 250, 240

```

```

240 ILR=ILQ+1
 IMR=IMQ+1
 X=R(ILR)*COSX-R(IMR)*SINX
 R(IMR)=R(ILR)*SINX+R(IMR)*COSX
 R(ILR)=X
250 CONTINUE
 X=2.0*A(LM)*SINCS
 Y=A(LL)*COSX2+A(MM)*SINX2-X
 X=A(LL)*SINX2+A(MM)*COSX2+X
 A(LM)=(A(LL)-A(MM))*SINCS+A(LM)*(COSX2-SINX2)
 A(LL)=Y
 A(MM)=X
C ПЕРЕВІРКА НА ЗАВЕРШЕННЯ
260 IF(M-N)270,280,270
270 M=M+1
 GO TO 120
280 IF(L-(N-1))290,300,290
290 L=L+1
 GO TO 110
300 IF(IND-1)320,310,320
310 IND=0
 GO TO 100
C ПОРІВНЯННЯ ГРАНИЧНОГО ЗНАЧЕННЯ
C З КІНЦЕВОЮ НОРМОЮ
320 IF(THR-ANRMX)330,330,90
C УПОРЯДКУВАННЯ ВЛАСНИХ ЗНАЧЕНЬ
C І ВЛАСНИХ ВЕКТОРІВ
330 IQ=-N
 DO 370 I=1,N
 IQ=IQ+N
 LL=I+(I*I-I)/2
 JQ=N*(I-2)
 DO 370 J=1,N
 JQ=JQ+N
 MM=J+(J*J-J)/2
 IF(A(LL)-A(MM))340,370,370
340 X=A(LL)
 A(LL)=A(MM)
 A(MM)=X
 IF(MV-1)350,370,350
350 DO 360 K=1,N
 ILR=IQ+K
 IMR=JQ+K
 X=R(ILR)
 R(ILR)=R(IMR)
360 R(IMR)=X
370 CONTINUE
 RETURN
 END

```

```

SUBROUTINE SVISZ(X,N,ITER,A,Y,RL)
CC
C ПІДПРОГРАМА ЗНАХОДЖЕННЯ МАКСИМАЛЬНОГО ВЛАСНОГО C

```

```

C ЗНАЧЕННЯ І ВІДПОВІДНОГО ВЛАСНОГО ВЕКТОРА ДІЙСНОЇ
C СИМЕТРИЧНОЇ МАТРИЦІ
C ВХІДНІ ПАРАМЕТРИ:
C -----
C X -ПОЧАТКОВЕ НАБЛИЖЕННЯ ДО ВЛАСНОГО ВЕКТОРА;
C N -РОЗМІРНІСТЬ ВЕКТОРА X;
C ITER-МАКСИМАЛЬНО ДОПУСТИМА КІЛЬКІСТЬ ІТЕРАЦІЙ;
C A -ВХІДНА МАТРИЦЯ.
C ВИХІДНІ ПАРАМЕТРИ:
C -----
C RL -МАКСИМАЛЬНЕ ВЛАСНЕ ЗНАЧЕННЯ;
C Y -ШУКАНИЙ ВЛАСНИЙ ВЕКТОР.
C ПІДПРОГРАМИ, ЩО ВИКЛИКАЮТЬСЯ:
C ANOR-П/П, ЯКА ОБЧИСЛЮЄ НОРМУ ВЕКТОРА.
C CC
C DIMENSION X(1), Y(1), A(N, N)
C DO 50 K=1, ITER
C ОБЧИСЛЕННЯ НОРМИ ВЕКТОРА
C CALL ANOR(S, X, N, X)
C DO 10 I=1, N
C 10 Y(I)=X(I)/S
C ОБЧИСЛЕННЯ НАСТУПНОЇ ІТЕРАЦІЇ
C DO 30 I=1, N
C S=0.
C DO 20 J=1, N
C 20 S=S+A(I, J)*Y(J)
C 30 X(I)=S
C ОБЧИСЛЕННЯ НАСТУПНОЇ ІТЕРАЦІЇ ВЛАСНОГО
C ЗНАЧЕННЯ RL
C CALL ANOR(S, X, N, Y)
C НОРМУВАННЯ НАСТУПНОЇ ІТЕРАЦІЇ ВЕКТОРА
C CALL ANOR(S, X, N, X)
C DO 40 I=1, N
C 40 Y(I)=X(I)/S
C КРИТЕРІЙ ЗУПИНУ: ЗМІНА ВЛАСНОГО
C ЗНАЧЕННЯ НА <= 0.01%, АБО КІЛЬКІСТЬ ІТЕРАЦІЙ
C ПЕРЕВИЩУЄ ITER
C IF (ABS((RL-S)/S) .LE. 0.0001) GO TO 50
C 50 RL=S
C 60 ITER=K
C RL=S
C RETURN
C END

C SUBROUTINE ANOR(S, X, N, Y)
C ПІДПРОГРАМА ОБЧИСЛЕННЯ НОРМИ ВЕКТОРА
C DIMENSION X(N), Y(N)
C S=0.
C DO 10 I=1, N
C 10 S=S+X(I)*Y(I)
C S=SQRT(S)
C RETURN
C END

```

## SUBROUTINE GOR(X, Y, Z, S, N)

```

С ССС
С ПІДПРОГРАМА ОБЧИСЛЕННЯ ЗНАЧЕННЯ ІНТЕРПОЛЯЦІЙНОГО С
С МНОГОЧЛЕНА ЗА СХЕМОЮ ГОРНЕРА С
С ВХІДНІ ПАРАМЕТРИ : С
С ----- С
С N - СТЕПІНЬ ІНТЕРПОЛЯЦІЙНОГО ПОЛІНОМА ПЛЮС 1 С
С (РОЗМІРНІСТЬ МАСИВІВ X І Y); С
С X(N) - МАСИВ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ З УРАХУ- С
С ВАННЯМ ЇХ КРАТНОСТІ; С
С Y(N) - МАСИВ ВЕРХНЬОЇ ДІАГОНАЛІ ТАБЛИЦІ С
С РОЗДІЛЕНИХ РІЗНИЦЬ; С
С Z - ТОЧКА, В ЯКІЙ ПОТРІБНО ЗНАЙТИ ЗНАЧЕННЯ С
С ІНТЕРПОЛЯЦІЙНОГО МНОГОЧЛЕНА. С
С ВИХІДНІ ПАРАМЕТРИ : С
С ----- С
С S - ЗНАЧЕННЯ ІНТЕРПОЛЯЦІЙНОГО МНОГОЧЛЕНА С
С В ТОЧЦІ Z. С

```

```

С ССС
С DIMENSION X(1), Y(1)
С S=Y(N)
С N1=N-1
С DO 10 I=1, N1
С J=N-I
С S=S*(Z-X(J))+Y(J)
10 RETURN
С END

```

## SUBROUTINE PER(X, Y, N)

```

С ССС
С ПІДПРОГРАМА ПЕРЕТВОРЕННЯ МНОГОЧЛЕНА, ЗАПИСАНОГО С
С У ФОРМІ НЬЮТОНА, В МНОГОЧЛЕН, ЗАПИСАНИЙ ПО СТЕПЕ- С
С НЯХ X С
С ВХІДНІ ПАРАМЕТРИ : С
С ----- С
С X(N) - МАСИВ ВУЗЛІВ ІНТЕРПОЛЮВАННЯ З УРАХУ- С
С ВАННЯМ ЇХ КРАТНОСТІ; С
С Y(N) - МАСИВ ВЕРХНЬОЇ ДІАГОНАЛІ ТАБЛИЦІ С
С РОЗДІЛЕНИХ РІЗНИЦЬ. С
С ВИХІДНІ ПАРАМЕТРИ : С
С ----- С
С Y(N) - МАСИВ КОЕФІЦІЄНТІВ D(I) В ЗАПИСУ ПО- С
С ЛІНОМА $P(X)=D(1)+D(2)*X+\dots+D(N)*X**(N-1)$. С
С ССС
С DIMENSION X(1), Y(1)
С DO 10 I=1, N-1
С J=N-I
С DO 10 K=J, N-1
10 Y(K)=Y(K)-Y(K+1)*X(J)
С RETURN
С END

```





```

DIMENSION X(N), Y(N), M(N1), M1(N)
M2=0
DO 20 I=1, N1
 LF=1
 DO 10 J=1, M(I)
 IF (J. GT. 2) LF=LF*(J-1)
 Y(M2+J)=Y(M2+J)/LF
 M1(M2+J)=M2+1
10
20
 M2=M2+M(I)
 DO 40 K=1, N-1
 DO 40 J=K+1, N
 I=N-J+K+1
 Q=X(I)-X(I-K)
 IF (Q. NE. 0.) GO TO 30
 M1(I)=M1(I)+1
 GO TO 40
30
40
 Y(M1(I))=(Y(M1(I))-Y(M1(I-1)))/Q
 CONTINUE
RETURN
END

```

```

SUBROUTINE SPLINE(N, X, Y, M, F1, F2, B, C, D)
CC
C ПІДПРОГРАМА ОБЧИСЛЕННЯ КОЕФІЦІЄНТІВ КУБІЧНОГО C
C ІНТЕРПОЛЯЦІЙНОГО СПЛАЙНА C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C N - КІЛЬКІСТЬ ЗАДАНИХ ТОЧОК АБО ВУЗЛІВ (N > 1); C
C X - МАСИВ АБСЦИС ВУЗЛІВ У СТРОГО C
C ЗРОСТАЮЧОМУ ПОРЯДКУ; C
C Y - МАСИВ ОРДИНАТ ВУЗЛІВ; C
C M - ПАРАМЕТР ЗАДАВАННЯ КРАЙОВИХ УМОВ: C
C ЯКЩО M=1, ТО F1, F2 ДОРІВНЮЮТЬ ЗНАЧЕННЯМ C
C ПЕРШОЇ ПОХІДНОЇ НА КІНЦЯХ ВІДРІЗКА; C
C ЯКЩО M=2, ТО НА КІНЦЯХ ВІДРІЗКА ЗАДАНО C
C НУЛЬОВІ ЗНАЧЕННЯ ДРУГОЇ ПОХІДНОЇ; C
C ЯКЩО M=3, ТО НА КІНЦЯХ ВІДРІЗКА ЗНАЧЕННЯ C
C ТРЕТЬОЇ ПОХІДНОЇ ВИРАЖАЮТЬСЯ C
C ЧЕРЕЗ ЗАДАНІ ЗНАЧЕННЯ ФУНКЦІЇ; C
C F1, F2 - МІСТЯТЬ ЗНАЧЕННЯ ПЕРШОЇ ПОХІДНОЇ НА КІН- C
C ЦЯХ ВІДРІЗКА, КОЛИ M=1, В ІНШИХ ВИПАДКАХ C
C F1=F2=0. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C B, C, D - ОДНОВИМІРНІ МАСИВИ РОЗМІРНІСТЮ N, ЩО C
C МІСТЯТЬ КОЕФІЦІЄНТИ СПЛАЙНА. C
CC
REAL X(N), Y(N), B(N), C(N), D(N)
NM1=N-1
IF (N. LT. 2) RETURN
IF (N. LT. 3) GO TO 130
C ПОБУДУВАТИ ТРИДАГОНАЛЬНУ СИСТЕМУ
C В-ДАГОНАЛЬ, D-НАДДІАГОНАЛЬ, C-ПРАВИ ЧАСТИНИ

```

```

D(1)=X(2)-X(1)
C(2)=(Y(2)-Y(1))/D(1)
DO 10 I=2,N
 D(I)=X(I+1)-X(I)
 B(I)=2.*(D(I-1)+D(I))
 C(I+1)=(Y(I+1)-Y(I))/D(I)
 C(I)=C(I+1)-C(I)
10 CONTINUE
С КРАЙОВІ УМОВИ
 IF(M.EQ.1) GO TO 20
 B(1)=-D(1)
 B(N)=-D(N-1)
 GO TO 30
20 B(1)=2*D(1)
 B(N)=2*D(N-1)
30 C(1)=0.
 C(N)=0.
 IF(N.EQ.3) GO TO 50
 IF(M.EQ.1) GO TO 40
 C(1)=C(3)/(X(4)-X(2))-C(2)/(X(3)-X(1))
 C(N)=C(N-1)/(X(N)-X(N-2))-C(N-2)/(X(N-1)-
* X(N-3))
 C(1)=C(1)*D(1)**2/(X(4)-X(1))
 C(N)=-C(N)*D(N-1)**2/(X(N)-X(N-3))
 GO TO 50
40 C(1)=(Y(2)-Y(1))/(X(2)-X(1))-F1
 C(N)=(Y(N)-Y(N-1))/(X(N)-X(N-1))+F2
 B(1)=2*D(A)
 B(N)=2*D(N-1)
С ПРЯМИЙ ХІД
50 DO 60 I=2,N
 T=D(I-1)/B(I-1)
 B(I)=B(I)-T*D(I-1)
 C(I)=C(I)-T*C(I-1)
60 CONTINUE
С ЗВОРОТНА ПІДСТАНОВКА
 IF(M.EQ.2) GO TO 70
 C(N)=C(N)/B(N)
 GO TO 80
70 C(N-1)=C(N-1)/B(N-1)
80 DO 90 IB=1,NM1
 I=N-IB
 C(I)=(C(I)-D(I)*C(I+1))/B(I)
90 CONTINUE
 IF(M.NE.2) GO TO 110
100 C(1)=0.
 C(N)=0.
С В C(I) ЗБЕРІГАЄТЬСЯ ВЕЛИЧИНА SIGMA.
С ОБЧИСЛЕННЯ КОЕФІЦІЄНТІВ ПОЛІНОМІВ
110 B(N)=(Y(N)-Y(NM1))/D(NM1)+D(NM1)*(C(NM1)+2.*
* C(N))
 DO 120 I=1,NM1

```

```

* B(I)=(Y(I+1)-Y(I))/D(I)-D(I)*(C(I+1)+2.*
 C(I))
 D(I)=(C(I+1)-C(I))/D(I)
 C(I)=3.*C(I)
120 CONTINUE
 C(N)=3.*C(N)
 D(N)=D(N-1)
 RETURN
130 IF(M.EQ.1) GO TO 140
 B(1)=(Y(2)-Y(1))/(X(2)-X(1))
 B(1)=B(2)
 GO TO 150
140 B(1)=F1
 B(N)=F2
150 C(1)=0.
 D(1)=0.
 C(2)=0.
 D(2)=0.
 RETURN
 END

```

```

 FUNCTION SEVAL(N,U,X,Y,B,C,D)
CC
C ПІДПРОГРАМА-ФУНКЦІЯ ЗНАХОДЖЕННЯ ЗНАЧЕННЯ СПЛАЙНА C
C В ТОЧЦІ U
C ВХІДНІ ПАРАМЕТРИ:
C -----
C N- РОЗМІРНІСТЬ МАСИВІВ X,Y,B,C,D;
C U- ТОЧКА, В ЯКІЙ НЕОБХІДНО ЗНАЙТИ ЗНАЧЕННЯ
C СПЛАЙНА;
C X- МАСИВ АБСЦИС РОЗМІРНІСТЮ N (N>1);
C Y- МАСИВ ОРДИНАТ РОЗМІРНІСТЮ N (N>1);
C B,C,D - МАСИВИ КОЕФІЦІЄНТІВ СПЛАЙНА, ЗНАЙДЕНІ
C ЗА П/П SPLINE.
CC
 REAL U,X(N),Y(N),B(N),C(N),D(N)
 I=1
 IF(I.GE.N) I=1
 IF(U.LT.X(I)) GO TO 10
 IF(U.LE.X(I+1)) GO TO 30
10 I=1
 J=N+1
20 K=(I+J)/2
 IF(U.LT.X(K)) J=K
 IF(U.GE.X(K)) I=K
 IF(J.GT.I+1) GO TO 20
30 DX=U-X(I)
 SEVAL=Y(I)+DX*(B(I)+DX*(C(I)+DX*D(I)))
 RETURN
 END

```

## SUBROUTINE SGL(N, P, X, F, MU, BA, CA, DA)

```

С ССС
С ПІДПРОГРАМА ОБЧИСЛЕННЯ КОЕФІЦІЄНТІВ КУБІЧНОГО С
С ЗГЛАДЖУЮЧОГО СПЛАЙНА С
С ВХІДНІ ПАРАМЕТРИ: С
С ----- С
С N - ЧИСЛО ЗАДАНИХ ТОЧОК (АБО ВУЗЛІВ); С
С P - МАСИВ ВАГОВИХ ПАРАМЕТРІВ; С
С X - МАСИВ АБСЦИС ВУЗЛІВ У СТРОГО ЗРОСТАЮЧОМУ С
С ПОРЯДКУ; С
С F - МАСИВ ОРДИНАТ ВУЗЛІВ. С
С ВИХІДНІ ПАРАМЕТРИ: С
С ----- С
С MU - МАСИВ ЗГЛАДЖЕНИХ ЗНАЧЕНЬ ФУНКЦІЇ; С
С BA, CA, DA - МАСИВИ КОЕФІЦІЄНТІВ СПЛАЙНА. С

```

```

С ССС
С REAL X(N), F(N), P(N), H(100), Q1(100), Q2(100),
С * Q3(100), S1(100), S2(100), S3(100),
С * Y(100), MH(100), MU(N), B(100),
С * BA(N), CA(N), DA(N), M(100)
С DO 10 I=1, N
С H(I)=X(I+1)-X(I)
10 CONTINUE

```

```

С ОБЧИСЛЕННЯ ТРЬОХ ДІАГОНАЛЕЙ МАТРИЦІ ЗГЛАДЖУВАННЯ Q
С ТА ВЕКТОРА ПРАВИХ ЧАСТИН B

```

```

К=N-3
IF(K.LT.1) GO TO 30
DO 20 I=1, K
С1=H(I+1)
С=H(I)+С1
С2=1./H(I)
С3=1./H(I+1)
С4=1./H(I+2)
P1=1./P(I)
P2=1./P(I+1)
P3=1./P(I+2)
СС=H(I)*H(I)
С1С=С1*С1
Q1(I)=С/3.+P1/СС+(С2+С3)*(С2+С3)*P2+P3/С1С
Q2(I)=С1/6.-(С2+С3)*P2*С3-(С3+С4)*P3*С3
Q3(I)=P3*С3*С4
B(I)=F(I)*С2-(С2+С3)*F(I+1)+F(I+2)*С3
CONTINUE

```

20  
30

```

L=N-1
DO 40 I=1, L
MM=L+1-I
M1=MM+1
M2=MM+2
С1=H(M1)
С=H(MM)+С1
С2=1./H(MM)

```

```

C3=1./H(M1)
C4=C2+C3
P1=1./P(MM)
P2=1./P(M1)
P3=1./P(M2)
CC=H(MM)*H(MM)
C1C=C1*C1
Q1(MM)=C/3.+P1/CC+C4*C4*P2+P3/C1C
B(MM)=F(MM)*C2-C4*F(M1)+C3*F(M2)
CONTINUE

```

40

```

L=N-2
C5=C3+1./H(M2)
Q2(L)=C1/6.-C4*P2*C3-C5*P3*C3

```

C РОЗВ'ЯЗУВАННЯ ЗНАЙДЕНОЇ СИСТЕМИ МЕТОДОМ КВАДРАТНОГО  
C КОРЕНЯ

```

S1(1)=SQRT(Q1(1))
S2(1)=Q2(1)/S1(1)
S1(2)=SQRT(Q1(2)-S2(1)*S2(1))
IF(N.EQ.3) GO TO 80
S2(2)=(Q2(2)-S2(1)*S3(1))/S1(2)
S3(1)=Q3(1)/S1(1)
IF(N.EQ.4) GO TO 70
S3(2)=Q3(2)/S1(2)
IF(N.EQ.5) GO TO 60
DO 50 I=3,K
 S1(I)=SQRT(Q1(I)-
* (S3(I-2)*S3(I-2)+S2(I-1)*S2(I-1)))
 S2(I)=(Q2(I)-S2(I-1)*S3(I-1))/S1(I)
 S3(I)=Q3(I)/S1(I)
CONTINUE

```

50

60

```

K=N-2
K1=K-1
K2=K-2
S1(K)=SQRT(Q1(K)-(S3(K2)*S3(K2)+S2(K1)*
* S2(K1)))
S2(K)=(Q2(K)-S2(K1)*S3(K1))/S1(K)

```

70

```

K=N-1
S1(K)=SQRT(Q1(K)-(S3(K-2)*S3(K-2)+S2(K-1)*
* S2(K-1)))

```

C ПРЯМИЙ ХІД МЕТОДУ КВАДРАТНОГО КОРЕНЯ

80

```

Y(1)=B(1)/S1(1)
Y(2)=(B(2)-S2(1)*Y(1))/S1(2)
IF(K.LT.3) GO TO 100
DO 90 I=3,K
 Y(I)=(B(I)-S3(I-2)*Y(I-2)-S2(I-1)*Y(I-1))
* /S1(I)
CONTINUE

```

90

C ЗВОРОТНИЙ ХІД МЕТОДУ КВАДРАТНОГО КОРЕНЯ

100

```

M(N-1)=Y(N-1)/S1(N-1)
IF(N.LE.2) GO TO 120
M(N-2)=(Y(N-2)-S2(N-2)*M(N-1))/S1(N-2)
N3=N-3

```



```

C RHMIN -ЗНАЧЕННЯ КРОКУ ІНТЕГРУВАННЯ ПРИ ЗАКІН-
C ЧЕННІ ОБЧИСЛЕНЬ;
C IERR -ОЗНАКА ЗАКІНЧЕННЯ РОБОТИ. ЯКЩО IERR=0 -
C НОРМАЛЬНЕ ЗАКІНЧЕННЯ РОБОТИ (ТОЧНІСТЬ
C EPS ДОСЯГНУТО); ЯКЩО IERR=1 - АВАРІЙНЕ
C ЗАКІНЧЕННЯ (КРОК ІНТЕГРУВАННЯ СТАВ МЕН-
C ШИМ ВІД HMIN, ТОЧНІСТЬ НЕ ДОСЯГНУТО);
C S -ЗНАЧЕННЯ ІНТЕГРАЛА. В РАЗІ АВАРІЙНОГО
C ЗАВЕРШЕННЯ ЗНАЧЕННЯМ S БУДЕ ОСТАННІМ
C НАБЛИЖЕННЯМ.

```

```

CC

```

```

 IERR=0
 N=3
 H=(B-A)*0.5
 H1=H/3.
 S1=(F(A)+F(B))*H1
 S2=4.*F(A+H)*H1
 S=S1+S2
10 N=N*2-1
 N1=N-1
 H=H*0.5
 S1=S1*0.5+S2*0.25
 S2=0.
 DO 20 I=1,N1,2
 S2=S2+4.*F(A+I*H)
20 CONTINUE
 S2=S2*H/3.
 SI2=S1+S2
 EROR=(S-SI2)/15.
 S=SI2+EROR
 RHMIN=H
 IF (ABS(EROR).LE.EPS) RETURN
 IF (H.GE.HMIN) GO TO 10
 IERR=1
 RETURN
 END

```

#### SUBROUTINE GAUSS(A, B, F, G)

```

CC

```

```

C ПІДПРОГРАМА ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІНТЕГРАЛА
C ФУНКЦІЇ F(X) НА ВІДРІЗКУ [A, B] ЗА КВАДРАТУРНОЮ
C ФОРМУЛОЮ ГАУССА З ВІСЬМОМА ВУЗЛАМИ
C ВХІДНІ ПАРАМЕТРИ:

```

```

C -----
C A-НИЖНЯ МЕЖА ІНТЕГРУВАННЯ;
C B-ВЕРХНЯ МЕЖА ІНТЕГРУВАННЯ;
C F-ІМ'Я ЗОВНІШНЬОЇ ПІДПРОГРАМИ, ОБЧИСЛЕННЯ
C ЗНАЧЕННЯ ПІДІНТЕГРАЛЬНОЇ ФУНКЦІЇ.
C ВИХІДНІ ПАРАМЕТРИ:

```

```

C -----
C G-НАБЛИЖЕНЕ ЗНАЧЕННЯ ІНТЕГРАЛА.

```

```

CC

```

```

 DIMENSION AG(8), XG(8)

```



```

* DATA AG/0. 10122854, 0. 22238104, 0. 31370664,
* 0. 36278378, 0. 36278378,
* 0. 31370664, 0. 22238104, 0. 10122854/
DATA XG/-0. 96028986, -0. 79666648, -0. 52553242,
* -0. 18343464, 0. 18343464,
* 0. 52553242, 0. 79666648, 0. 96028986/

```

```

A1=(B+A)*0.5
A2=(B-A)*0.5
G=0.
DO 10 I=1,8
 X=A1+A2*XG(I)
 G=G+AG(I)*F(X)
CONTINUE

```

10

```

G=G*A2
RETURN
END

```

```

SUBROUTINE QUANC8 (FUN, A, B, ABSERR, RELERR,
* RESULT, ERREST, NOFUN, FLAG)

```

```

CC
C АДАПТИВНА ПІДПРОГРАМА ОБЧИСЛЕННЯ ВИЗНАЧЕНОГО ІН- C
C ТЕГРАЛА ВІД ФУНКЦІЇ F(X) НА ВІДРІЗКУ [A,B] З C
C ВИКОРИСТАННЯМ КВАДРАТНОЇ ФОРМУЛИ НЬЮТОНА-КОТЕСА. C
C ПІДПРОГРАМА АВТОМАТИЧНО ВИЗНАЧАЄ ПІДІНТЕРВАЛИ C
C ТАК, ЩОБ ЗНАЙДЕНИЙ РЕЗУЛЬТАТ ЗАДОВОЛЬНЯВ C
C ЗАДАНУ ТОЧНІСТЬ. У РІЗНИХ ЧАСТИНАХ ІНТЕРВАЛУ МО- C
C ЖУТЬ ВИКОРИСТОВУВАТИСЬ СІТКИ РІЗНИХ РОЗМІРІВ, ПО- C
C РІВНЯНО ГРУБІ ТАМ, ДЕ ПІДІНТЕГРАЛЬНА ФУНКЦІЯ ГЛА- C
C ДКА І ЗМІНЮЄТЬСЯ ПОВІЛЬНО, А ГУСТІ В ОБЛАСТЯХ, C
C ДЕ ВОНА ЗМІНЮЄТЬСЯ СИЛЬНО. C
C ПІДПРОГРАМУ ЗАПОЗИЧЕНО З [26, С.113-122] C
C ВХІДНІ ПАРАМЕТРИ: C

```

```

C FUN -ІМ'Я ПІДПРОГРАМИ-ФУНКЦІЇ ОБЧИСЛЕННЯ C
C ЗНАЧЕННЯ ПІДІНТЕГРАЛЬНОЇ ФУНКЦІЇ; C
C A -НИЖНЯ МЕЖА ІНТЕГРУВАННЯ; C
C B -ВЕРХНЯ МЕЖА ІНТЕГРУВАННЯ (МОЖЕ БУТИ C
C МЕНШОЮ, НІЖ МЕЖА А); C
C ABSERR -МЕЖА АБСОЛЮТНОЇ ПОХИБКИ; C
C RELERR -МЕЖА ВІДНОСНОЇ ПОХИБКИ. C

```

```

ВХІДНІ ПАРАМЕТРИ:

```

```

C RESULT -НАБЛИЖЕНЕ ЗНАЧЕННЯ ІНТЕГРАЛА; C
C ERREST -ОЦІНКА РЕАЛЬНОЇ ПОХИБКИ; C
C NOFUN -КІЛЬКІСТЬ ЗНАЧЕНЬ ФУНКЦІЇ, ВИКОРИС- C
C ТАНИХ ПРИ ОБЧИСЛЕННІ RESULT; C
C FLAG -ІНДИКАТОР НАДІЙНОСТІ. ЯКЩО FLAG=0 - C
C RESULT ЗАДОВОЛЬНЯЄ ЗАДАНУ МЕЖУ ПОХИБКИ. C
C ЯКЩО FLAG=XXX.YYY, ТО XXX-КІЛЬКІСТЬ C
C ІНТЕРВАЛІВ, ДЛЯ ЯКИХ НЕ БУЛО ЗБІЖНОСТІ, C
C А 0.YYY - ЧАСТИНА ОСНОВНОГО ІНТЕРВАЛУ, C
C ЩО ЗАЛИШИЛАСЬ ДЛЯ ОБРОБКИ НА ТОЙ ЧАС. C

```



\* F(13)+W4\*F(12))\*STEP

QNOW=QLEFT+QRIGHT(LEV+1)

QDIFF=QNOW-QPREV

AREA=AREA+QDIFF

ESTERR=ABS(QDIFF)/1023.

TOLERR=AMAX1(ABSERR, RELERR\*ABS(AREA))\*(STEP/  
STONE)

\*

IF(LEV. LT. LEVMIN) GO TO 50

IF(LEV. GE. LEVMAX) GO TO 62

IF(NOFUN. GT. NOFIN) GO TO 60

50

IF(ESTERR. LE. TOLERR) GO TO 70

NIM=2\*NIM

LEV=LEV+1

DO 52 I=1,8

FSAVE(I, LEV)=F(I+8)

XSAVE(I, LEV)=X(I+8)

52

CONTINUE

QPREV=QLEFT

DO 55 I=1,8

J=-I

F(2\*J+18)=F(J+9)

X(2\*J+18)=X(J+9)

55

CONTINUE

GO TO 30

60

NOFIN=2\*NOFIN

LEVMAX=LEVOUT

FLAG=FLAG+(B-X0)/(B-A)

GO TO 70

62

FLAG=FLAG+1.

70

RESULT=RESULT+QNOW

ERREST=ERREST+ESTERR

COR11=COR11+QDIFF/1023.

72

IF(NIM. EQ. 2\*(NIM/2)) GO TO 75

NIM=NIM/2

LEV=LEV-1

GO TO 72

75

NIM=NIM+1

IF(LEV. LE. 0) GO TO 80

QPREV=QRIGHT(LEV)

X0=X(16)

F0=F(16)

DO 78 I=1,8

F(2\*I)=FSAVE(I, LEV)

X(2\*I)=XSAVE(I, LEV)

78

CONTINUE

GO TO 30

80

RESULT=RESULT+COR11

IF(ERREST. EQ. 0. 0) RETURN

82

TEMP=ABS(RESULT)+ERREST

IF(TEMP. NE. ABS(RESULT)) RETURN

ERREST=2.\*ERREST

GO TO 82

END



LOGICAL MATU, MATV

IERR=0

DO 100 I=1, M

DO 100 J=1, N

U(I, J)=A(I, J)

CONTINUE

G=0.0

SCALE=0.0

ANORM=0.0

DO 300 I=1, N

L=I+1

RV1(I)=SCALE\*G

G=0.0

S=0.0

SCALE=0.0

IF(I.GT.M) GO TO 210

DO 120 K=I, M

SCALE=SCALE+ABS(U(K, I))

IF(SCALE.EQ.0.0) GO TO 210

DO 130 K=I, M

U(K, I)=U(K, I)/SCALE

S=S+U(K, I)\*\*2

CONTINUE

F=U(I, I)

G=-SIGN(SQRT(S), F)

H=F\*G-S

U(I, I)=F-G

IF(I.EQ.N) GO TO 190

DO 150 J=L, N

S=0.0

DO 140 K=I, M

S=S+U(K, I)\*U(K, J)

F=S/H

DO 150 K=I, M

U(K, J)=U(K, J)+F\*U(K, I)

CONTINUE

DO 200 K=I, M

U(K, I)=SCALE\*U(K, I)

W(I)=SCALE\*G

G=0.0

S=0.0

SCALE=0.0

IF(I.GT.M.OR.I.EQ.N) GO TO 290

DO 220 K=L, N

SCALE=SCALE+ABS(U(I, K))

IF(SCALE.EQ.0.0) GO TO 290

DO 230 K=L, N

U(I, K)=U(I, K)/SCALE

S=S+U(I, K)\*\*2

CONTINUE

F=U(I, L)

G=-SIGN(SQRT(S) F)

```

H=F*G-S
U(I, L)=F-G
DO 240 K=L, N
 RV1(K)=U(I, K)/H
 IF(I.EQ.M) GO TO 270
 DO 260 J=L, M
 S=0.0
 DO 250 K=L, N
 S=S+U(J, K)*U(I, K)
 DO 260 K=L, N
 U(J, K)=U(J, K)+S*RV1(K)
 CONTINUE
 DO 280 K=L, N
 U(I, K)=SCALE*U(I, K)
 ANORM=AMAX1(ANORM, ABS(W(I))+ABS(RV1(I)))
 CONTINUE
 IF(.NOT.MATV) GO TO 410
 DO 400 II=1, N
 I=N+1-II
 IF(I.EQ.N) GO TO 390
 IF(G.EQ.0.0) GO TO 360
 DO 320 J=L, N
 V(J, I)=(U(I, J)/U(I, L))/G
 DO 350 J=L, N
 S=0.0
 DO 340 K=L, N
 S=S+U(I, K)*V(K, J)
 DO 350 K=L, N
 V(K, J)=V(K, J)+S*V(K, I)
 CONTINUE
 DO 380 J=L, N
 V(I, J)=0.0
 V(J, I)=0.0
 CONTINUE
 V(I, I)=1.0
 G=RV1(I)
 LI
 CONTINUE
 IF(.NOT.MATU) GO TO 510
 MN=N
 IF(M.LT.N) MN=M
 DO 500 II=1, MN
 I=MN+1-II
 L=I+1
 G=W(I)
 IF(I.EQ.N) GO TO 430
 DO 420 J=L, N
 U(I, J)=0.0
 IF(G.EQ.0.0) GO TO 475
 IF(I.EQ.MN) GO TO 460
 DO 450 J=L, N
 S=0.0

```

```

440 DO 440 K=L, M
 S=S+U(K, I)*U(K, J)
 F=(S/U(I, I))/G
 DO 450 K=I, M
 U(K, J)=U(K, J)+F*U(K, I)
450 CONTINUE
460 DO 470 J=I, M
470 U(J, I)=U(J, I)/G
 GO TO 490
475 DO 480 J=I, M
480 U(J, I)=0.0
490 U(I, I)=U(I, I)+1.0
500 CONTINUE
510 DO 700 KK=1, N
 K1=N-KK
 K=K1+1
 ITS=0
520 DO 530 LL=1, K
 L1=K-LL
 L=L1+1
 IF (ABS(RV1(L))+ANORM.EQ.ANORM) GO TO 565
 IF (ABS(W(L1))+ANORM.EQ.ANORM) GO TO 540
530 CONTINUE
540 C=0.0
 S=1.0
 DO 560 I=L, K
 F=S*RV1(I)
 RV1(I)=C*RV1(I)
 IF (ABS(F)+ANORM.EQ.ANORM) GO TO 565
 G=W(I)
 H=SQRT(F*F+C*C)
 W(I)=H
 C=G/H
 S=-F/H
 IF (.NOT.MATU) GO TO 560
 DO 550 J=1, M
 Y=U(J, L1)
 Z=U(J, I)
 U(J, L1)=Y*C+Z*S
 U(J, I)=-Y*S+Z*C
 CONTINUE
 CONTINUE
 Z=W(K)
 IF (L.EQ.K) GO TO 650
 IF (ITS.EQ.30) GO TO 1000
 ITS=ITS+1
 X=W(L)
 Y=W(K1)
 G=RV1(K1)
 H=RV1(K)
 F=((Y-Z)*(Y+Z)+(G-H)*(G+H))/(2.0*H*Y)
 G=SQRT(F*F+1.0)

```

$F = ((X-Z) * (X+Z) + H * (Y / (F + \text{SIGN}(G F)) - H))$

C=1.0

S=1.0

DO 600 I1=L, K1

I=I1+1

G=RV1(I)

Y=W(I)

H=S\*G

G=C\*G

Z=SQRT(F\*F+H\*H)

RV1(I1)=Z

C=F/Z

S=H/Z

F=X\*C+G\*S

G=-X\*S+G\*C

H=Y\*S

Y=Y\*C

IF(.NOT.MATV) GO TO 575

DO 570 J=1, N

X=V(J, I1)

Z=V(J, I)

V(J, I1)=X\*C+Z\*S

V(J, I)=-X\*S+Z\*C

CONTINUE

Z=SQRT(F\*F+H\*H)

W(I1)=Z

IF(Z.EQ.0.0) GO TO 580

C=F/Z

S=H/Z

F=C\*G+S\*Y

X=-S\*G+C\*Y

IF(.NOT.MATU) GO TO 600

DO 590 J=1, M

Y=U(J, I1)

Z=U(J, I)

U(J, I1)=Y\*C+Z\*S

U(J, I)=-Y\*S+Z\*C

CONTINUE

CONTINUE

RV1(L)=0.0

RV1(K)=F

W(K)=X

GO TO 520

IF(Z.GE.0.0) GO TO 700

W(K)=-Z

IF(.NOT.MATV) GO TO 700

DO 690 J=1, N

V(J, K)=-V(J, K)

CONTINUE

GO TO 1001

1000 IERR=K

1001 RETURN

END





```

C ТЕ, ЩО РОЗВ'ЯЗОК ПЕРЕТВОРИВСЯ В НУЛЬ. ТЕСТ ТІЛЬКИ
C ВІДНОСНОЇ ПОХИБКИ НЕ ПРОХОДИТЬ. ДЛЯ ПРОДОВЖЕННЯ
C РОЗВ'ЯЗУВАННЯ ПОТРІБНО ЗАДАТИ НЕНУЛЬОВЕ
C ЗНАЧЕННЯ ПАРАМЕТРА ABSERR:
C IFLAG=6 - ІНТЕГРУВАННЯ НЕ БУЛО ЗАВЕРШЕНО ЧЕРЕЗ
C ТЕ, ЩО ЗАДАНА ТОЧНІСТЬ НЕ МОЖЕ БУТИ
C ДОСЯГНУТА НАВІТЬ ПРИ НАЙМЕНШОМУ ДОПУСТИМОМУ
C ЗНАЧЕННІ КРОКУ. НЕОБХІДНО ЗБІЛЬШИТИ ЗНАЧЕННЯ
C ABSERR;
C IFLAG=7 - РКФ45 Є НЕЕФЕКТИВНОЮ ПРИ
C РОЗВ'ЯЗУВАННІ ЗАДАЧІ. СЛІД ЗАСТОСУВАТИ
C РЕЖИМ ПОКРОКОВОГО ІНТЕГРУВАННЯ;
C IFLAG=8 - НЕПРАВИЛЬНЕ ЗАДАВАННЯ ВХІДНИХ
C ПАРАМЕТРІВ.
C CCC

```

```

 INTEGER NEQN, IFLAG, IWORK(5)
 INTEGER K1, K2, K3, K4, K5, K6, K1M
 REAL Y(NEQN), T, TOUT, RELERR, ABSERR, WORK(1)
 EXTERNAL F
 K1M=NEQN+1
 K1=K1M+1
 K2=K1+NEQN
 K3=K2+NEQN
 K4=K3+NEQN
 K5=K4+NEQN
 K6=K5+NEQN
 CALL RKFS(F, NEQN, Y, T, TOUT, RELERR, ABSERR, IFLAG,
* WORK(1), WORK(K1M), WORK(K1), WOK(K2),
* WORK(K3), WORK(K4), WORK(K5), WORK(K6),
* WORK(K6+1), IWORK(1), IWORK(2),
* IWORK(3), IWORK(4), IWORK(5))

```

```

 RETURN
 END

```

```

SUBROUTINE RKFS(F, NEQN, Y, T, TOUT, RELERR, ABSERR,
* IFLAG, YP, H, F1, F2, F3, F4, F5, SAVRE,
* SAVAE, NFE, KOP, INIT, JFLAG, KFLAG)
 EXTERNAL F
 LOGICAL HFAILD, OUTPUT
 REAL Y(NEQN), T, TOUT, RELERR, ABSERR, H, YP(NEQN),
* F1(NEQN), F2(NEQN), F3(NEQN), F4(NEQN),
* F5(NEQN), SAVRE, SAVAE
 REAL A, AE, DT, EE, EEOET, ESTTOL, ET, HMIN, REMIN,
* RER, S, SCALE, TOL, TOLN, U26, EPSP1, EPS,
* YPK, AMAX1, AMIN1
 INTEGER K, MAXNFE, MFLAG
 DATA REMIN /1.E-12/, MAXNFE /3000/
 IF (NEQN.LT.1) GO TO 20
 IF ((RELERR.LT.0.0).OR.(ABSERR.LT.0.0))
* GO TO 20
 MFLAG=IABS(IFLAG)
 IF((MFLAG.EQ.0).OR.(MFLAG.GT.8)) GO TO 20
 IF(MFLAG.NE.1) GO TO 30

```

```

10 EPS=1.0
 EPS=EPS/2.0
 EPSP1=EPS+1.0
 IF(EPSP1.GT.1.0) GO TO 10
 U26=26.0*EPS
 GO TO 80
20 IFLAG=8
 RETURN
30 IF((T.EQ.TOUT).AND.(KFLAG.NE.3)) GO TO 20
 IF(MFLAG.NE.2) GO TO 40
 IF((KFLAG.EQ.3).OR.(INIT.EQ.0)) GO TO 70
 IF(KFLAG.EQ.4) GO TO 60
 IF((KFLAG.EQ.5).AND.(ABSERR.EQ.0.0)) GO TO 50
 IF((KFLAG.EQ.6).AND.(RELERR.LE.SAVRE).AND.
* (ABSERR.LE.SAVAE)) GO TO 50
 GO TO 80
40 IF(IFLAG.EQ.3) GO TO 70
 IF(IFLAG.EQ.4) GO TO 60
 IF((IFLAG.EQ.5).AND.(ABSERR.GT.0.0)) GO TO 70
50 STOP
60 NFE=0
 IF(MFLAG.EQ.2) GO TO 80
70 IFLAG=JFLAG
 IF(KFLAG.EQ.3)MFLAG=IABS(IFLAG)
80 JFLAG=IFLAG
 KFLAG=0
 SAVRE=RELERR
 SAVAE=ABSERR
 RER=2.0*EPS+REMIN
 IF(RELERR.GE.RER) GO TO 90
 RELERR=RER
 IFLAG=3
 RETURN
90 DT=TOUT-T
 IF(MFLAG.EQ.1) GO TO 100
 IF(INIT.EQ.0) GO TO 110
 GO TO 130
100 INIT=0
 KOP=0
 A=T
 CALL F(A,Y,YP)
 NFE=1
 IF(T.NE.TOUT) GO TO 110
 IFLAG=2
 RETURN
110 INIT=1
 H=ABS(DT)
 TOLN=0.
 DO 120 K=1,NEQN
 TOL=RELERR*ABS(Y(K))+ABSERR
 IF(TOL.LE.0.) GO TO 120
 TOLN=TOL

```

```

 YPK=ABS (YP (K))
 IF (YPK*H**5. GT. TOL) H=(TOL/YPK)**0.2
120 CONTINUE
 IF (TOLN. LE. 0.) H=0.0
 H=AMAX1 (H, U26*AMAX1 (ABS (T), ABS (DT)))
 JFLAG=ISIGN (2, IFLAG)
130 H=SIGN (H, DT)
 IF (ABS (H). GE. 2.0*ABS (DT)) KOP=KOP+1
 IF (KOP. NE. 100) GO TO 140
 KOP=0
 IFLAG=7
 RETURN
140 IF (ABS (DT). GT. U26*ABS (T)) GO TO 160
 DO 150 K=1, NEQN
150 Y (K)=Y (K)+DT*YP (K)
 A=TOUT
 CALL F (A, Y, YP)
 NFE=NFE+1
 GO TO 250
160 OUTPUT=.FALSE.
 SCALE=2.0/RELERR
 AE=SCALE*ABSERR
170 HFAILD=.FALSE.
 HMIN=U26*ABS (T)
 DT=TOUT-T
 IF (ABS (DT). GE. 2.0*ABS (H)) GO TO 190
 OUTPUT=.TRUE.
 H=DT
 GO TO 190
180 H=0.5*DT
190 IF (NFE. LE. MAXNFE) GO TO 200
 IFLAG=4
 KFLAG=4
 RETURN
200 CALL FEHL (F, NEQN, Y, T, H, YP, F1, F2, F3, F4, F5, F1)
 NFE=NFE+5
 EEOET=0.0
 DO 220 K=1, NEQN
 ET=ABS (Y (K))+ABS (F1 (K))+AE
 IF (ET. GT. 0.0) GO TO 210
 IFLAG=5
 RETURN
210 EE=ABS ((-2090.0*YP (K)+(21970.0*F3 (K)-
 * 15048.0*F4 (K)))+(22528.0*F2 (K)-27360.0*
 * F5 (K)))
220 EEOET=AMAX1 (EEOET, EE/ET)
 ESTTOL=ABS (H)*EEOET*SCALE/752400.0
 IF (ESTTOL. LE. 1.0) GO TO 230
 HFAILD=.TRUE.
 OUTPUT=.FALSE.
 S=0.1
 IF (ESTTOL. LT. 59049.0) S=0.9/ESTTOL**0.2

```

```

H=S*H
IF (ABS(H).GT.HMIN) GO TO 190
IFLAG=6
KFLAG=6
RETURN
230 T=T+H
DO 240 K=1,NEQN
240 Y(K)=F1(K)
A=T
CALL F(A,Y,YP)
NFE=NFE+1
S=5.0
IF (HFAILD)S=AMIN1(S,1.0)
H=SIGN(AMAX1(S*ABS(H),HMIN),H)
IF (OUTPUT) GO TO 250
IF (IFLAG.GT.0) GO TO 170
IFLAG=-2
RETURN
250 T=TOUT
IFLAG=2
RETURN
END

```

```

SUBROUTINE FEHL(F,NEQN,Y,T,H,YP,F1,F2,F3,F4,
* F5,S)

```

```

* INTEGER NEQN,K
REAL Y(NEQN),T,H,YP(NEQN),F1(NEQN),F2(NEQN),
* F3(NEQN),F4(NEQN),F5(NEQN),S(NEQN),CH
EXTERNAL F
CH=H/4.0
DO 10 K=1,NEQN
10 F5(K)=Y(K)+CH*YP(K)
CALL F(T+CH,F5,F1)
CH=3.0*H/32.0
DO 20 K=1,NEQN
20 F5(K)=Y(K)+CH*(YP(K)+3.0*F1(K))
CALL F(T+3.0*H/8.0,F5,F2)
CH=H/2197.0
DO 30 K=1,NEQN
30 F5(K)=Y(K)+CH*(1932.0*YP(K)+(7296.0*
* F2(K)-7200.0*F1(K)))
CALL F(T+12.0*H/13.0,F5,F3)
CH=H/4104.0
DO 40 K=1,NEQN
40 F5(K)=Y(K)+CH*((8341.0*YP(K)-845.0*
* F3(K))+(29440.0*F2(K)-32832.0*F1(K)))
CALL F(T+H,F5,F4)
CH=H/20520.0
DO 50 K=1,NEQN
50 F1(K)=Y(K)+CH*((-6080.0*YP(K)+(9295.0*
* F3(K)-5643.0*F4(K)))+(41040.0*F1(K)-
* 28352.0*F2(K)))
CALL F(T+H/2.0,F1,F5)

```

CH=N/7618050.0

DO 60 K=1, NEQN

60

\* S(K)=Y(K)+CH\*((902880.0\*Y(K)+(3855735.0\*  
\* F3(K)-1371249.0\*F4(K)))+(3953664.0\*F2(K)+  
\* 277020.0\*F5(K))

RETURN  
END

SUBROUTINE DRIVE(N, TO, HO, YO, TOUT, EPS, MF, INDEX)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C СКЛАВ С.В. ЗОЛУТУХІН, "КІІЦА", КАФ. "ПОВС", C  
C 28.04.86 C

C ПАКЕТ ПІДПРОГРАМ GEARZ РОЗВ'ЯЗУВАННЯ ЗАДАЧІ КОШІ C  
C ДЛЯ ЗДР: C

C  $DU/DT = F(Y, T), Y = (Y(1), Y(2), \dots, Y(N)).$  C

C ПАКЕТ ГРУНТУЄТЬСЯ НА ВЕРСІЇ GEAR C

C ВІД 20 ГРУДНЯ 1974 Р. C

C ПІДПРОГРАМА DRIVE С ДРАЙВЕРОМ ДЛЯ ПАКЕТА C

C ПІДПРОГРАМ GEARZ. C

C СПИСОК ЛІТЕРАТУРИ : C

C 1. HINDMARSH A. C. GEAR.. ORDINARY DIFFERENTIAL C  
C EQUATION SYSTEM SOLVER, UCID-30001 REV. 3, C  
C LAWRENCE LIVERMORE LABORATORY, P. O. BOX 808, C  
C LIVERMORE, CA 94550, DEC. 1974. C

C 2. HINDMARSH A. C. LINEAR MULTISTEP METHODS FOR C  
C ORDINARY DIFFERENTIAL EQUATIONS.. METHOD C  
C FORMULATIONS, STABILITY, AND THE METHODS OF C  
C NORDSIECK AND GEAR, UCRL-51186 REV. 1, L. L. L. , C  
C MARCH 1972. C

C 3. HINDMARSH A. C. CONSTRUCTION OF MATHEMATICAL C  
C SOFTWARE, PART III.. THE CONTROL OF ERROR IN C  
C THE GEAR PACKAGE FOR ORDINARY DIFFERENTIAL C  
C EQUATIONS, UCID-30050 PART 3, L. L. L. , C  
C AUGUST 1972. C

C 4. ЗОЛУТУХІН С.В.: ПРОЦЕДУРА SGEARZ ЧИСЛЕННОГО C  
C РЕШЕННЯ СИСТЕМ ЖЕСТКИХ ОБЫКНОВЕННЫХ ДИФФЕРЕН- C  
C ЦИАЛЬНЫХ УРАВНЕНИЙ. РУКОПИСЬ ДЕП. В ВИНТИ C  
C 6 ДЕКАБРЯ 1985 Г., 8389-В . C

C ----- C

C ДЛЯ КОЖНОГО СКІНЧЕНОГО ЗНАЧЕННЯ Т ВИКЛИК П/П C  
C DRIVE ПРОВОДИТЬСЯ ОДИН РАЗ. П/П DRIVE БАГАТО РАЗІВ C  
C ЗВЕРТАЄТЬСЯ ДО ОСНОВНОГО ІНТЕГРАТОРА - П/П STIFF. C  
C ВХІДНІ ПАРАМЕТРИ: C

C ----- C

C N - КІЛЬКІСТЬ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ПЕРШОГО C  
C ПОРЯДКУ. В ПРОЦЕСІ ІНТЕГРУВАННЯ ЧИСЛО N C  
C МОЖНА ЗМІНЮВАТИ ТІЛЬКИ В БІК ЙОГО ЗМЕНШЕННЯ; C

C TO - ПОЧАТКОВЕ ЗНАЧЕННЯ АГУМЕНТУ T C  
C (ЗАДАЄТЬСЯ ТІЛЬКИ ПРИ ПЕРШОМУ ЗВЕРНЕННІ); C

C HO - ПОЧАТКОВИЙ КРОК ПО T (ЗАДАЄТЬСЯ ТІЛЬКИ C  
C ПРИ ПЕРШОМУ ЗВЕРНЕННІ); C

C YO - МАСИВ РОЗМІРНІСТЮ N, ЯКИЙ МІСТИТЬ ПОЧАТКОВІ C  
C ЗНАЧЕННЯ Y (ЗАДАЄТЬСЯ ТІЛЬКИ ПРИ ПЕРШОМУ C

- ЗВЕРНЕННІ):**
- TOUT** - ЗНАЧЕННЯ АРГУМЕНТУ T В КІНЦІ ЧЕРГОВОГО ІНТЕРВАЛУ ІНТЕГРУВАННЯ. ПРОЦЕС ІНТЕГРУВАННЯ ЗВЧАЙНО ЗАКІНЧУЄТЬСЯ ПРИ  $T > \text{TOUT}$ , ДАЛІ АВТОМАТИЧНО ЗДІЙСНЮЄТЬСЯ ІНТЕРПОЛЮВАННЯ ДЛЯ ТОЧКИ  $T = \text{TOUT}$ ;
- EPS** - НЕОБХІДНА ТОЧНІСТЬ РЕЗУЛЬТАТУ НА КРОЦІ (ДЛЯ ЗНАЧЕНЬ INDEX, ВІДМІННИХ ВІД -1, ЗАДАЄТЬСЯ ТІЛЬКИ ПРИ ПЕРШОМУ ЗВЕРНЕННІ). ОЦІНКА ЛОКАЛЬНОЇ ПОХИБКИ ДИСКРЕТИЗАЦІЇ ПРОВОДИТЬСЯ ЗА СЕРЕДНЬОКВАДРАТИЧНОЮ НОРМОЮ З УРАХУВАННЯМ ВАГОВИХ ЗНАЧЕНЬ  $UMAX(I)$ , ЯКІ ОБЧИСЛЮЮТЬСЯ В П/П DRIVE. ТЕСТ ПОХИБКИ ВИКОНУЄТЬСЯ, ЯКЩО ЗНАЙДЕНЕ ЗНАЧЕННЯ  $< \text{EPS}$ ;
- MF** - ПРАПОР МЕТОДУ (ЗАДАЄТЬСЯ ТІЛЬКИ ПРИ ПЕРШОМУ ЗВЕРНЕННІ, ЯКЩО INDEX ВІДМІННИЙ ВІД -1). ДОПУСТИМИМИ ЗНАЧЕННЯМИ MF Є: 10, 11, 12, 13, 20, 21, 22, 23, 30, 31, 32, 33 .  
 МОЖНА ЗАПИСАТИ :  $MF = 10 * \text{METH} + \text{MITER}$  ;  
 METH - ПЕРША ЦИФРА MF - ВКАЗУЄ НА ФОРМУЛИ ІНТЕГРУВАННЯ:  
 METH = 1 - ФОРМУЛИ АДАМСА,  
 METH = 2 - ЖОРСТКО СТІЙКІ ФОРМУЛИ ГІРА (АБО BDF),  
 METH = 3 - ЖОРСТКО СТІЙКІ МОДИФІКАЦІЇ ФОРМУЛ ГІРА З ПОЛІПШЕНИМИ ПОБЛИЗУ УВНОЇ ОСІ ХАРАКТЕРИСТИКАМИ АБСОЛЮТНОЇ СТІЙКОСТІ;  
 MITER - ДРУГА ЦИФРА MF - ВКАЗУЄ НА СПОСІБ РОЗВ'ЯЗУВАННЯ НЕЯВНИХ РІВНЯНЬ КОРЕКТОРА:  
 MITER = 0 - МЕТОД ПРОСТИХ ІТЕРАЦІЙ (БЕЗ МАТРИЦІ ЯКОБІ),  
 MITER = 1 - МЕТОД НЬЮТОНА З АНАЛІТИЧНИМ ЗАПИСОМ МАТРИЦІ ЯКОБІ ПРАВОЇ ЧАСТИНИ. В ЦЬОМУ ВИПАДКУ ВИКОРИСТОВУЄТЬСЯ П/П PEDERV (ДИВ. НИЖЧЕ),  
 MITER = 2 - МЕТОД НЬЮТОНА. МАТРИЦЯ ЯКОБІ ОБЧИСЛЮЄТЬСЯ АВТОМАТИЧНО ЧИСЕЛЬНИМ ДИФЕРЕНЦІЮВАННЯМ,  
 MITER = 3 - МЕТОД НЬЮТОНА З ДІАГОНАЛЬНОЮ АПРОКСИМАЦІЄЮ МАТРИЦІ ЯКОБІ;
- INDEX** - ВХІДНИЙ ІНДЕКС, ЯКИЙ ВКАЗУЄ НА ТИП ЗВЕРНЕННЯ ДО П/П DRIVE. МОЖЕ НАБУВАТИ ТАКИХ ЗНАЧЕНЬ:  
 1 - П/П DRIVE ВИКЛИКАЄТЬСЯ ПЕРШИЙ РАЗ (ЗАДАЧА РОЗВ'ЯЗУЄТЬСЯ СПОЧАТКУ),  
 0 - ПРОЦЕС ІНТЕГРУВАННЯ ПОТРІБНО ПРОДОВЖИТИ З ПОПЕРЕДНІМИ ЗНАЧЕННЯМИ ПАРАМЕТРІВ N, EPS ТА MF,

- 1 - ПРОЦЕС ІНТЕГРУВАННЯ ПОТРІБНО ПРО-  
ДОВЖИТИ З ІНШИМИ ЗНАЧЕННЯМИ ПАРА-  
МЕТРІВ N, EPS I (АБО) MF,  
2 - ТЕ САМЕ, ЩО Й ПРИ INDEX = 0. ПОТРІБ-  
НО ЗАВЕРШИТИ ІНТЕГРУВАННЯ ПРИ ЗАДАНО-  
МУ ЗНАЧЕННІ TOUT БЕЗ ІТЕРПОЛЮВАННЯ,  
3 - АНАЛОГІЧНО INDEX = 0. ПІСЛЯ ВИКЛИКУ  
П/П DRIVE ЗДІЙСНЮЮТЬСЯ ОДИН КРОК І  
ПОВЕРНЕННЯ ДЛЯ КОНТРОЛЮ У ВИХІДНУ  
ПРОГРАМУ. ЗНАЧЕННЯ TOUT НЕ ПОТРІБНЕ.  
ПРИМІТКА: ПРИ УСПІШНОМУ ЗАВЕРШЕННІ П/П  
DRIVE ВХІДНИЙ ІНДЕКС INDEX МАЄ ЗНАЧЕННЯ  
НУЛЬ, І ДЛЯ ПРОДОВЖЕННЯ ІНТЕГРУВАННЯ НЕ-  
МАЄ ПОТРЕБИ В ЙОГО ЗАНУЛЕННІ. ДОСИТЬ  
ПРИСВОІТИ НАСТУПНІ ЗНАЧЕННЯ TOUT І ПО-  
ВТОРИТИ ВИКЛИК П/П DRIVE. ВАРІАНТ  
INDEX = -1 МОЖЕ ЗАСТОСОВУВАТИСЬ ЯК ПІСЛЯ  
УСПІШНОГО, ТАК І ПІСЛЯ НЕВДАЛОГО  
ЗАВЕРШЕННЯ РОБОТИ П/П DRIVE.

**ВИХІДНІ ПАРАМЕТРИ:**

- NO - ЗНАЧЕННЯ ОСТАНЬОГО КРОКУ ІНТЕГРУВАННЯ (ЯК  
ПРИ УСПІШНОМУ, ТАК І ПРИ НЕВДАЛОМУ  
ЗАВЕРШЕННІ РОБОТИ П/П DRIVE);  
YO - МАСИВ ЗНАЧЕНЬ Y ПРИ T = TOUT;  
TOUT - ВИХІДНЕ ЗНАЧЕННЯ T ПІСЛЯ ЗАВЕРШЕННЯ РО-  
БОТИ П/П DRIVE. ЗБІГАЄТЬСЯ З ПОЧАТКОВИМ  
ЗНАЧЕННЯМ TOUT (ЗА ВИНЯТКОМ ВИПАДКУ НЕ-  
ВДАЛОГО ЗАВЕРШЕННЯ DRIVE АБО ЗАДАВАННЯ  
ВХІДНОГО ЗНАЧЕННЯ INDEX = 3);  
INDEX - ВИХІДНИЙ ІНДЕКС:  
0 - ІНТЕГРУВАННЯ ПРИ ЗАДАНИХ ПАРАМЕТРАХ  
ВИКОНАНО УСПІШНО,  
-1 - ІНТЕГРУВАННЯ ПЕРЕРВАНО: КРОК ЗМЕН-  
ШИВСЯ В  $1.E10$  РАЗІВ ПОРІВНЯНО З  
ПОЧАТКОВИМ І ТЕСТ ПОХИБКИ НЕ  
ВИКОНУЄТЬСЯ,  
-2 - НЕОБХІДНА ТОЧНІСТЬ ДОСИТЬ ВЕЛИКА  
ДЛЯ ДАНОЇ ЗАДАЧІ. ПІСЛЯ НЕВИКОНАННЯ  
ТЕСТУ ПОХИБКИ СІМ РАЗІВ ПІДРЯД ПРО-  
ЦЕС ІНТЕГРУВАННЯ ПРИПИНЕНО,  
-3 - ІНТЕГРУВАННЯ ПЕРЕРВАНО: КРОК ЗМЕН-  
ШИВСЯ В  $1.E10$  РАЗІВ ПОРІВНЯНО З ПО-  
ЧАТКОВИМ І НЕМАЄ ЗБІЖНОСТІ В РОЗ-  
В'ЯЗКУ РІВНЯНЬ КОРЕКТОРА,  
-4 - НЕПРАВИЛЬНІ ЗНАЧЕННЯ ВХІДНИХ ПАРА-  
МЕТРІВ (ДИВ. ДРУКОВАНІ ПОВІДОМЛЕН-  
НЯ),  
-5 - ДЛЯ ВХІДНОГО ЗНАЧЕННЯ INDEX = -1  
ПРИ ПОВТОРНОМУ ЗВЕРНЕННІ ДО П/П  
DRIVE ПОВИННО БУТИ ЗАДАНЕ НОВЕ  
ЗНАЧЕННЯ TOUT.



С ПОРЯД 3 П/П DRIVE В ПАКЕТИ GEARZ ВИКОРИСТОВУЮТЬСЯ С  
С ПІДПРОГРАМИ: С  
С INTERP (TOUT, Y, NO, YO) - ІНТЕРПОЛОВАННЯ ЗНАЧЕНЬ С  
С МАСИВУ Y ДЛЯ ТОЧКИ T = TOUT; С  
С STIFF (Y, NO) - ОСНОВНА ПІДПРОГРАМА-ІНТЕГРАТОР. С  
С ВИКОНУЄ ОДИН КРОК І ВІДПОВІДНИЙ С  
С КОНТРОЛЬ ПОХИБКИ; С  
С COSET (METH, NQ, EL, TQ, MAXDER) - ВИЗНАЧАЄ КОНСТАНТИ С  
С ДЛЯ П/П STIFF; С  
С COAD (EL, NQ) - ВИЗНАЧАЄ КОЕФІЦІЄНТИ ДЛЯ ФОРМУЛ С  
С АДАМСА; С  
С PSET (Y, NO, CON, MITER, IER) - ОБЧИСЛЮЄ Й ОБРОБЛЯЄ С  
С МАТРИЦЮ ЯКОБІ J = DF/DY; С  
С DEC (N, NO, A, IP, IER) - ВИКОНУЄ LU-ДЕКОМПОЗИЦІЮ С  
С РЕЗУЛЬТУЮЧОЇ МАТРИЦІ A; С  
С SOL (N, NO, A, B, IP) - РОЗВ'ЯЗУЄ СЛАР A\*X=B . С  
С ПІДГОТОВЛЕНУ П/П DEC. С  
С ПРИМІТКА: П/П PSET, DEC І SOL ЗАСТОСОВУЮТЬСЯ ДЛЯ С  
С ОБЧИСЛЕНЬ ТІЛЬКИ ПРИ ЗНАЧЕННЯХ MITER = 1 АБО 2 С  
С ВХІДНОГО ПАРАМЕТРУ MF. П/П COAD ВИКОРИСТОВУЄТЬСЯ С  
С ДЛЯ ОБЧИСЛЕНЬ ТІЛЬКИ ПРИ ЗНАЧЕННІ METH = 1 С  
С ВХІДНОГО ПАРАМЕТРА MF. С  
С ДЛЯ ВИКОРИСТАННЯ СКЛАДАЮТЬСЯ ТАКІ ПІДПРОГРАМИ: С  
С DIFFUN (N, T, Y, YDOT) - ОБЧИСЛЮЄ ВЕКТОР ЗАВДОВЖКИ N С  
С ПРАВОЇ ЧАСТИНИ СИСТЕМИ ЗДР: YDOT = (Y, T); С  
С PEDERV (N, T, Y, PD, NO) - ОБЧИСЛЮЄ МАТРИЦЮ ЯКОБІ С  
С РОЗМІРНІСТЮ N\*N ПРАВОЇ ЧАСТИНИ І ЗАПАМ'ЯТО- С  
С ВУЄ ЇЇ У ВІДПОВІДНОМУ ДВОВИМІРНОМУ МАСИВІ С  
С PD. ЕЛЕМЕНТ PD(I, J) ДОРІВНЮЄ ЧАСТИННІЙ ПО- С  
С ХІДНІЙ YDOT(I) ПО ЗМІННІЙ Y(J). П/П PEDERV С  
С ЗАСТОСОВУЄТЬСЯ В ОБЧИСЛЕННЯХ ТІЛЬКИ У ВИ- С  
С ПАДКУ, КОЛИ ВХІДНЕ ЗНАЧЕННЯ MITER = 1. ДЛЯ С  
С ІНШИХ ЗНАЧЕНЬ MITER ПОВИННА БУТИ ПІДГОТОВ- С  
С ЛЕНА ФІКТИВНА П/П PEDERV. С  
С РОЗМІР МАСИВІВ В ОПЕРАТОРАХ COMMON І DIMENSION С  
С ДАНОЇ ПІДПРОГРАМИ РОЗРАХОВАНИЙ НЕ БІЛЬШ ЯК НА 20 С  
С РІВНЯНЬ. ЯКЩО N БІЛЬШЕ НІЖ 20, ТО СЛІД ВІДПОВІДНО С  
С ЗБІЛЬШИТИ ДОВЖИНУ МАСИВІВ. ДОВЖИНА МАСИВУ PW МАЄ С  
С БУТИ НЕ МЕНШОЮ НІЖ N\*\*2 ДЛЯ MITER = 1 АБО С  
С MITER = 2. ПРИ MITER = 3 ЦЯ ДОВЖИНА МОЖЕ С  
С БУТИ ЗМЕНШЕНА ДО N, А ПРИ MITER = 0 - ДО 1. С  
С В БУДЬ-ЯКОМУ ВИПАДКУ РОЗМІРИ МАСИВІВ YMAX, ERROR, С  
С SAVE1, SAVE2, IPIV В ОПЕРАТОРІ COMMON І ДОВЖИНА С  
С СТОВПЦЯ (ПЕРШИЙ ПАРАМЕТР) МАСИВУ Y В ОПЕРАТОРІ С  
С DIMENSION ПОВИННІ МАТИ ЗНАЧЕННЯ НЕ МЕНШ ЯК N. С  
С ПРИ ВХІДНИХ ЗНАЧЕННЯХ METH = 2 І 3 ДОВЖИНА РЯДКА С  
С МАСИВУ Y МОЖЕ БУТИ ЗМЕНШЕНА З 13 ДО 6. С  
С МАСИВ IPIV ВИКОРИСТОВУЄТЬСЯ В ОБЧИСЛЕННЯХ ТІЛЬКИ С  
С ТОДІ, КОЛИ MITER = 1 АБО 2. С  
С БЛОК COMMON З ПОМІТКОЮ GEAR9 ПРИ ЙОГО ВИКОРИС- С  
С ТАННІ ДАЄ ЗМОГУ МАТИ ДОСТУП ДО ТАКИХ ПАРАМЕТРІВ: С  
С HUSED - ЗНАЧЕННЯ ОСТАНЬОГО УСПІШНО ВИКОНАНОГО С

```

C КРОКУ;
C NSTEP - КІЛЬКІСТЬ УСІХ УСПІШНИХ КРОКІВ;
C NFE - КІЛЬКІСТЬ УСІХ ЗВЕРНЕНЬ ДО П/П DIFFUN;
C NJE - КІЛЬКІСТЬ УСІХ ОБЧИСЛЕНЬ МАТРИЦІ ЯКОБІ
C (П/П PEDEEV МОЖЕ Й НЕ ЗНАДОБИТИСЯ).
C ЗНАЧЕННЯ НАСТУПНИХ ПАРАМЕТРІВ В ОПЕРАТОРІ DATA
C ПОВИННІ ЗАДАВАТИСЯ ПРИ ВИКОРИСТАННІ:
C UROUND - КОНСТАНТА ЗАОКРУГЛЕННЯ ДЛЯ ДАНОГО КОМ-
C П'ЮТЕРА: НАЙМЕНШЕ ДОДАТНЕ ЧИСЛО U
C TAKE, ЩО (1.+U) НЕ ДОРІВНЮЄ 1;
C IPR - ЛОГІЧНИЙ НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ ПОВІДОМ-
C ЛЕНЬ У ХОДІ ІНТЕГРУВАННЯ.
CC

```

```

DIMENSION Y(20, 13)
DIMENSION YO(N)
COMMON/GEAR1/T, H, HMIN, HMAX, EPSO, RROUND, NO, MFO,
* KFLAG, JSTART
COMMON/GEAR2/YMAX(20)
COMMON/GEAR3/ERROR(20)
COMMON/GEAR4/SAVE1(20)
COMMON/GEAR5/SAVE2(20)
COMMON/GEAR6/PW(400)
COMMON/GEAR7/IPIV(20)
COMMON/GEAR8/EPSJ, NSQ
COMMON/GEAR9/HUSED, NQUSED, NSTEP, NFE, NJE
COMMON/NOUSTR/IPR
DATA UROUND/1.2E-7/
RROUND=UROUND
IF(INDEX.EQ.0) GO TO 20
IF(INDEX.EQ.2) GO TO 30
IF(INDEX.EQ.-1) GO TO 40
IF(INDEX.EQ.3) GO TO 50
IF(INDEX.NE.1) GO TO 200
IF(EPS.LE.0.) GO TO 170
IF(N.LE.0) GO TO 180
IF((TO-TOUT)*NO.GE.0.) GO TO 190

```

```

C ЯКЩО ПОТРІБНІ ВІДМІННІ ВІД НАВЕДЕНИХ НИЖЧЕ ЗНА-
C ЧЕННЯ ВАГОВИХ КОЕФІЦІЄНТІВ YMAX(I) І ОБМЕЖЕННЯ
C АБСОЛЮТНОГО ЗНАЧЕННЯ КРОКУ HMIN І HMAX, ТО ВОНИ
C МОЖУТЬ БУТИ ТУТ ЗМІНЕНІ. ВСІ ЗНАЧЕННЯ - ДОДАТНІ.
DO 10 I=1, N

```

```

10 YMAX(I)=ABS(YO(I))
 IF(YMAX(I).EQ.0.) YMAX(I)=1.
 Y(I, 1)=YO(I)

```

```

NO=N
T=TO
H=HO
IF((T+H).EQ.T) WRITE(IPR, 300)
HMIN=ABS(HO)
HMAX=.5
HMAX=ABS(TO-TOUT)*10.
EPSO=EPS
MFO=MF
JSTART=0

```

```

NO=N
NSQ=NO*N
EPSJ=SQRT(UROUND)
NHOOT=0
GO TO 60
C TOUTP - ПОПЕРЕДНЄ ЗНАЧЕННЯ TOUT ДЛЯ
C ВИЗНАЧЕННЯ HMAX.
20 HMAX=ABS(TOUT-TOUTP)*10.
GO TO 90
30 HMAX=ABS(TOUT-TOUTP)*10.
IF((T-TOUT)*H.GE.O.) GO TO 220
GO TO 100
40 IF((T-TOUT)*H.GE.O.) GO TO 210
JSTART=-1
NO=N
EPSO=EPS
MFO=MF
50 IF((T+H).EQ.T) WRITE(IPR,300)
60 CALL STIFF(Y,NO)
KGO=1-KFLAG
GO TO (70,110,140,160),KGO
C KFLAG = 0, -1, -2, -3 .
70 CONTINUE
C УСПІШНЕ ЗАВЕРШЕННЯ РОБОТИ П/П STIFF.
C
C ПЕРЕВИЗНАЧАЮТЬСЯ ВАГОВІ ЗНАЧЕННЯ YMAX(I). ЯКЩО
C ДЛЯ YMAX(I) ПОТРІБНІ ІНШІ ВЕЛИЧИНИ, ЇХ МОЖНА ТУТ
C ЖЕ ВСТАНОВИТИ. ПРОВІДУЄТЬСЯ ТЕСТ НА ВІДПОВІДНІСТЬ
C ТОЧНОСТІ КОМП'ЮТЕРА І ЗНАЧЕННЯ EPS.
C У ЦЬОЇ ЧАСТИНІ ПІДПРОГРАМИ МОЖНА РОЗМІСТИТИ ІНШІ
C ТЕСТИ ТА ОБЧИСЛЕННЯ, НЕОБХІДНІ ПІСЛЯ ПОВЕРНЕННЯ З
C П/П STIFF.
C ЯКЩО БУЛО ВСТАНОВЛЕНО ЗНАЧЕННЯ INDEX = 3, ТО YO
C НАБУВАЄ ПОТОЧНОГО ЗНАЧЕННЯ Y. ПРИ ВСТАНОВЛЕННІ
C INDEX = 2 КРОК Н КОНТРОЛЮЄТЬСЯ В П/П STIFF ТАК,
C ЩОБ ВЛУЧИТИ В ТОЧКУ TOUT БЕЗ ДОПОМОГИ
C П/П INTERP.
C ДЛЯ ВСІХ ЗНАЧЕНЬ INDEX, КРІМ 3, YO НАБУВАЄ
C ЗНАЧЕННЯ Y В ТОЧЦІ T = TOUT.
C ЯКЩО НЕМАЄ ПОТРЕБИ В ІНТЕРПОЛЮВАННІ, ТО СЛІД
C ВИЛУЧИТИ ОПЕРАТОРИ CALL INTERP(TOUT,Y,NO,YO) I
C GO TO 240 I ЗАМІСТЬ НИХ УВЕСТИ ОПЕРАТОР
C GO TO 220 .
D=0.
DO 80 I=1,N
AYI=ABS(Y(I,1))
YMAX(I)=AMAX1(YMAX(I),AYI)
80 D=D+(AYI/YMAX(I))**2
D=D*(UROUND/EPS)**2
IF(D.GT.FLOAT(N)) GO TO 150
IF(INDEX.EQ.3) GO TO 220
IF(INDEX.EQ.2) GO TO 100

```

```

90 IF((T-TOUT)*H.LT.O.) GO TO 50
 CALL INTERP(TOUT, Y, NO, YO)
 GO TO 240
100 IF(((T+H)-TOUT)*H.LE.O.) GO TO 50
 IF(ABS(T-TOUT).LE.256.*UROUND*HMAX) GO TO 220
 IF((T-TOUT)*H.GE.O.) GO TO 220
 H=(TOUT-T)*(1.-4.*UROUND)
 JSTART=-1
 GO TO 50
C ПРИ НЕВДАЛОМУ ЗАВЕРШЕННІ П/П STIFF І ЗНАЧЕННІ
C KFLAG, ВІДМІННОМУ ВІД -2, ДО 10 РАЗІВ ПРОВОДЯТЬСЯ
C СПРОБИ ПРОДОВЖИТИ ІНТЕГРУВАННЯ. КОЖНОГО РАЗУ ЗНА-
C ЧЕННЯ Н І HMIN ЗМЕНШУЮТЬСЯ В 10 РАЗІВ.
C ПРИ KFLAG = -2 ПРОЦЕС ІНТЕГРУВАННЯ ПЕРЕРИВАЄТЬСЯ.
110 WRITE (IPR, 301)T
120 IF(NHOUT.EQ.10) GO TO 130
 NHOUT=NHOUT+1
 HMIN=.1*HMIN
 H=.1*H
 WRITE (IPR, 302)H
 JSTART=-1
 GO TO 50
130 WRITE (IPR, 303)
 GO TO 220
140 WRITE (IPR, 304)T, H
 GO TO 220
150 WRITE (IPR, 305)T
 KFLAG=-2
 GO TO 220
160 WRITE (IPR, 306)T
 GO TO 120
170 WRITE (IPR, 307)
 INDEX=-4
 RETURN
180 WRITE (IPR, 308)
 INDEX=-4
 RETURN
190 WRITE (IPR, 309)
 INDEX=-4
 RETURN
200 WRITE (IPR, 310)INDEX
 INDEX=-4
 RETURN
210 WRITE (IPR, 311) T, TOUT, H
 CALL INTERP(TOUT, Y, NO, YO)
 INDEX=-5
 RETURN
220 TOUT=T
 DO 230 I=1, N
230 YO(I)=Y(I, 1)
240 INDEX=KFLAG
 TOUTP=TOUT

```

```

NO=HUSED
IF(KFLAG.NE.0) NO=H
RETURN
300 FORMAT(' ПОПЕРЕДЖЕННЯ : T+N=T НА НАСТУПНОМУ ',
* ' КРОКІ. ')
301 FORMAT(/// KFLAG=-1 - ДЛЯ T=',E16.8/
* ' НЕ ВИКОНУЄТЬСЯ ТЕСТ ПОХИБКИ ПРИ',
* ' ABS(N)=HMIN'//)
302 FORMAT(' N ЗМЕНШИЛОСЬ ДО ',E16.8/
* ' I КРОК БУДЕ ПОВТОРЕНИЙ'//)
303 FORMAT(/// ЗАДАЧА НЕ РОЗВ'ЯЗУЄТЬСЯ З ',
* ' ДАНИМИ УМОВАМИ'//)
304 FORMAT(/// KFLAG=-2 - ДЛЯ T=',E16.8,' H=',
* ' E16.8/' НЕОБХІДНЕ ЗНАЧЕННЯ ПОХИБКИ ДУЖЕ МАЛЕ',
* ' ДЛЯ ЦЬОЇ ЗАДАЧІ'//)
305 FORMAT(/// КІНЕЦЬ ІНТЕГРУВАННЯ В DRIVE ПРИ T=',
* ' E16.8'/// ЗНАЧЕННЯ EPS ДУЖЕ МАЛЕ ПОРІВ',
* ' НЯНО З ТОЧНІСТЮ КОМП'ЮТЕРА'//)
306 FORMAT(/// KFLAG=-3 - ПРИ T=',E16.8/
* ' НЕМАЄ ЗБІЖНОСТІ ПРИ РОЗВ'ЯЗУВАННІ РІВНЯНЬ',
* ' КОРЕКТОРА'//)
307 FORMAT(/// НЕПРАВИЛЬНЕ ВХІДНЕ ЗНАЧЕННЯ EPS'//)
308 FORMAT(/// НЕПРАВИЛЬНЕ ВХІДНЕ ЗНАЧЕННЯ N'//)
309 FORMAT(/// НЕПРАВИЛЬНЕ ВХІДНЕ ЗНАЧЕННЯ :'/
* ' ЗНАЧЕННЯ (T-TOUT)*N НЕВІД'ЄМНЕ'//)
310 FORMAT(/// НЕПРАВИЛЬНЕ ВХІДНЕ ЗНАЧЕННЯ :',
* ' INDEX=',I5//)
311 FORMAT(/// ПРИ ВВЕДЕННІ НЕВІД'ЄМНОГО (T-TOUT)*N',
* ' INDEX=-1'/// T=',E16.8,' TOUT=',E16.8,
* ' H=',E16.8/' ІНТЕРПОЛЮВАННЯ ПРОВЕДЕНО ЯК ',
* ' ПРИ НОРМАЛЬНОМУ ПОВЕРНЕННІ.'// НЕ БУЛИ ',
* ' ВИКОНАНІ НЕОБХІДНІ ЗМІНИ ПАРАМЕТРІВ.')
C----- КІНЕЦЬ П/П DRIVE -----C
END

```

SUBROUTINE SOL(N,NDIM,A,B,IP)

```

CC
C РОЗВ'ЯЗУВАННЯ СЛАР A*X = B
C ВХІДНІ ПАРАМЕТРИ:

```

- C N - ПОРЯДОК МАТРИЦІ;
- C NDIM - РОЗМІРНІСТЬ МАСИВУ A;
- C A - МАТРИЦЯ, ЗВЕДЕНА ДО ВИГЛЯДУ L-U ЗА
- C ДОПОМОГОЮ П/П DEC;
- C B - ВЕКТОР ПРАВИХ ЧАСТИН;
- C IP - ВЕКТОР, ЯКИЙ МІСТИТЬ ІНДЕКСИ ВУЗЛОВИХ
- C СТРИЧОК (ДИВ. П/П DEC).

```

C П/П SOL ВИКОРИСТОВУЄТЬСЯ ТІЛЬКИ У ВИПАДКУ
C IER = 0 (ДИВ. П/П DEC).
C ВИХІДНІ ПАРАМЕТРИ:
C-----

```

```

C B - ВЕКТОР РОЗВ'ЯЗКУ X. C
CC
DIMENSION A(NDIM, N), B(N), IP(N)
IF(N.EQ.1) GO TO 50
NM1=N-1
DO 20 K=1, NM1
 KP1=K+1
 M=IP(K)
 T=B(M)
 B(M)=B(K)
 B(K)=T
 DO 10 I=KP1, N
 B(I)=B(I)+A(I, K)*T
10 CONTINUE
20 DO 40 KB=1, NM1
 KM1=N-KB
 K=KM1+1
 B(K)=B(K)/A(K, K)
 T=-B(K)
 DO 30 I=1, KM1
 B(I)=B(I)+A(I, K)*T
30 CONTINUE
40 B(1)=B(1)/A(1, 1)
50 RETURN

```

C----- КІНЕЦЬ П/П SOL -----C  
 END

```

SUBROUTINE DEC(N, NDIM, A, IP, IER)
CC
C LU-ДЕКОМПОЗИЦІЯ МАТРИЦІ А ЗА ДОПОМОГОЮ МЕТОДУ C
C ВИКЛЮЧЕНЬ ГАУССА C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C N - ПОРЯДОК МАТРИЦІ; C
C NDIM - РОЗМІРНІСТЬ МАСИВУ А; C
C A - ВХІДНА МАТРИЦЯ. C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C
C A(I, J), I=1, ..., J - ВЕРХНЯ ТРИКУТНА C
C ЧАСТИНА: U; C
C A(I, J), I=J+1, ..., N - НИЖНЯ ТРИКУТНА C
C ЧАСТИНА: I - L; C
C IP(K), K < N - ІНДЕКС К-ГО C
C ВУЗЛОВОГО РЯДКА; C
C IP(N) - (-1)**(ЧИСЛО ПЕРЕСТАНОВОК) АБО 0; C
C IER - 0, ЯКЩО МАТРИЦЯ А НЕВИРОДЖЕНА; C
C IER - K, ЯКЩО ВИРОДЖЕНІСТЬ МАТРИЦІ А ВИЯВИЛАСЯ C
C НА К-МУ КРОЦІ; C
C DETERMINANT(A) - IP(N)*A(1, 1)*A(2, 2)* C
C *...*A(N, N). C

```

```

C У ВИПАДКУ ВИРОДЖЕНОСТІ А ІР(N)=0 І ПРОВОДИТЬСЯ C
C ДІЛЕННЯ НА НУЛЬ У П/П SOL. C
C ЛІТЕРАТУРА : C
C MOLER C.B., ALGORITHM 423, LINEAR EQUATION C
C SOLVER, C.A.C.M. 15 (1972), P. 274. C
CC

```

```

DIMENSION A(NDIM, N), IP(N)

```

```

IER=0

```

```

IP(N)=1

```

```

IF(N.EQ.1) GO TO 70

```

```

NM1=N-1

```

```

DO 60 K=1, NM1

```

```

 KP1=K+1

```

```

 M=K

```

```

 DO 10 I=KP1, N

```

```

10 IF(ABS(A(I, K)).GT.ABS(A(M, K))) M=I

```

```

 IP(K)=M

```

```

 T=A(M, K)

```

```

 IF(M.EQ.K) GO TO 20

```

```

 IP(N)=-IP(N)

```

```

 A(M, K)=A(K, K)

```

```

 A(K, K)=T

```

```

20 IF(T.EQ.0) GO TO 80

```

```

 T=1./T

```

```

 DO 30 I=KP1, N

```

```

30 A(I, K)=-A(I, K)*T

```

```

 DO 50 J=KP1, N

```

```

 T=A(M, J)

```

```

 A(M, J)=A(K, J)

```

```

 A(K, J)=T

```

```

 IF(T.EQ.0.) GO TO 50

```

```

 DO 40 I=KP1, N

```

```

40 A(I, J)=A(I, J)A(I, K)*T

```

```

50 CONTINUE

```

```

60 CONTINUE

```

```

70 K=N

```

```

 IF(A(N, N).EQ.0.) GO TO 80

```

```

 RETURN

```

```

80 IER=K

```

```

 IP(N)=0

```

```

 RETURN

```

```

C----- КІНЕЦЬ П/П DEC -----C

```

```

END

```

```

SUBROUTINE STIFF(Y, NO)

```

```

CC

```

```

C П/П STIFF ВИКОНУЄ ОДИН КРОК ІНТЕГРУВАННЯ ЗАДАЧІ C

```

```

C КОШІ ДЛЯ СИСТЕМИ ЗДР. ЗВ'ЯЗОК З П/П STIFF ВСТА- C

```

```

C НОВЛЮЄТЬСЯ ЗА ДОПОМОГОЮ ТАКИХ ЗМІННИХ: C

```

```

C Y - МАСИВ (NO, LMAX). МІСТИТЬ ЗАЛЕЖНІ ЗМІННІ ТА C

```

```

C ІХ МАСШТАБОВАНІ ПОХІДНІ. ДЛЯ ФОРМУЛ АДАМСА C

```

LMAX= 13, ДЛЯ ФОРМУЛ ГІРА LMAX= 6. МАКСИМАЛЬ- С  
НИЙ ПОРЯДОК ФОРМУЛ=LMAX-1(ДИВ. П/П COSET). С

Y(I, J+1) МІСТИТЬ J-ТУ ПОХІДНУ Y(I), ПОДІЛЕНУ С  
НА N\*\*J/FACTORIAL(J) (J =0, 1, ..., NQ); С

NO - ЦІЛА КОНСТАНТА, НЕ МЕНША НІЖ N, ЯКА ВИКОРИ- С  
СТОВУЄТЬСЯ В ОПЕРАТОРАХ DIMENSION ДЛЯ ДВО- С  
ВИМІРНИХ МАСИВІВ. С

ПАРАМЕТРИ, ЩО ПЕРЕДАЮТЬСЯ ЧЕРЕЗ COMMON БЛОКИ: С  
T - НЕЗАЛЕЖНА ЗМІННА (ЗМІНЮЄТЬСЯ НА КОЖНОМУ С  
КРОЦІ); С

H - ЗНАЧЕННЯ КРОКУ, ЩО ПРОГНОЗУЄТЬСЯ. ЗМІННА H ЗА- С  
ЛЕЖИТЬ ВІД РЕЗУЛЬТАТУ РОБОТИ АЛГОРИТМУ КОНТ- С  
РОЛЮ ЛОКАЛЬНОЇ ПОХИБКИ МЕТОДУ. H МОЖЕ БУТИ ЯК С  
ДОДАТНОЮ, ТАК І ВІД'ЄМНОЮ ВЕЛИЧИНОЮ. В РАМ- С  
КАХ ОДНІЄЇ ЗАДАЧІ ЗНАК КРОКУ H МАЄ БУТИ С  
СТАЛИМ; С

HMIN, HMAX - МІНІМАЛЬНЕ ТА МАКСИМАЛЬНЕ ЗНАЧЕННЯ С  
КРОКУ, ЩО ПРОГНОЗУЄТЬСЯ; С

EPS - ВЕРХНЯ МЕЖА ПОХИБКИ (ДИВ. П/П DRIVE); С

UROUND - КОНСТАНТА ЗАОКРУГЛЕННЯ ДЛЯ ДАНОГО С  
КОМП'ЮТЕРА; С

N - КІЛЬКІСТЬ ДИФЕРЕНЦІАЛЬНИХ РІВНЯНЬ ПЕРШОГО С  
ПОРЯДКУ; С

MF - ПРАПОР МЕТОДУ (ДИВ. П/П DRIVE); С

KFLAG - ОЗНАКА ЗАВЕРШЕННЯ П/П STIFF. С

МАЄ ТАКІ ЗНАЧЕННЯ: С

0 - КРОК БУВ УСПІШНИМ, С

-1 - НЕОБХІДНЕ ЗНАЧЕННЯ EPS ДОСИТЬ МАЛЕ ДЛЯ С  
ABS(H) = HMIN, С

-2 - НЕОБХІДНЕ ЗНАЧЕННЯ EPS ДОСИТЬ МАЛЕ ДЛЯ С  
ДАНОЇ ЗАДАЧІ, С

-3 - НЕМАЄ ЗБІЖНОСТІ ПРИ РОЗВ'ЯЗУВАННІ РІВ- С  
НЯНЬ КОРЕКТОРА ДЛЯ ABS(H) = HMIN. С

ЯКЩО ПРИ ЗАВЕРШЕННІ П/П STIFF KFLAG < 0, ВЕ- С  
ЛИЧИНИ T І Y ЗБЕРІГАЮТЬ ТІ САМІ ЗНАЧЕННЯ, ЩО С  
Й НА ПОЧАТКУ КРОКУ. ЗМІННА H МАЄ ПРИ ЦЬОМУ С  
ЗНАЧЕННЯ ОСТАНЬОГО НЕВДАЛОГО КРОКУ; С

JSTART - ЦІЛА ЗМІННА, ЩО ВИКОРИСТОВУЄТЬСЯ ЯК С  
ВХІДНИЙ І ВИХІДНИЙ ПАРАМЕТРИ. ЗНАЧЕННЯ С  
JSTART НА ВХОДІ П/П STIFF МАЮТЬ ТАКИЙ С  
ЗМІСТ: С

JSTART = 0 - ВИКОНАТИ ПЕРШИЙ КРОК, С

JSTART > 0 - ВИКОНАТИ НОВИЙ КРОК ЯК ПРОДОВ- С  
ЖЕННЯ СТАРОГО, С

JSTART < 0 - ВИКОНАТИ НАСТУПНИЙ КРОК З НОВИМИ С  
ЗНАЧЕННЯМИ H, EPS, N І (ЧИ) MF. С

НА ВИХОДІ П/П STIFF JSTART МАЄ ЗНАЧЕННЯ NQ - С  
ПОТОЧНОГО ПОРЯДКУ МЕТОДУ; С

YMAX - МАСИВ ІЗ N ЕЛЕМЕНТІВ ДЛЯ ОБЧИСЛЕННЯ ВІД- С  
НОСНОЇ ПОХИБКИ МЕТОДУ НА КРОЦІ; С

ERROR - МАСИВ ІЗ N ЕЛЕМЕНТІВ. ЗНАЧЕННЯ С  
ERROR(1)/TQ(2) Є ОЦІНКОЮ ЛОКАЛЬНОЇ ПОХИБ- С



```
C КИ МЕТОДУ НА КРОЦІ ДЛЯ КОМПОНЕНТИ Y(I); C
C SAVE1,SAVE2 - РОБОЧИ МАСИВИ, КОЖНИЙ ЗАВДОВЖКИ N; C
C PW - МАСИВ, ЩО МІСТИТЬ ЗНАЧЕННЯ ЕЛЕМЕНТІВ МАТРИЦІ C
C ЯКОБІ ДЛЯ ВХІДНОГО ПАРАМЕТРА MITER, ЯКИЙ НЕ C
C ДОРІВНЮЄ НУЛЮ (ДИВ. П/П DRIVE); C
C IPIV - ЦІЛИЙ МАСИВ ЗАВДОВЖКИ N ДЛЯ LU-ДЕКОМПОЗИЦІЇ C
C РЕЗУЛЬТУЮЧОЇ МАТРИЦІ А (ДИВ. П/П DEC). C
```

```
CC
DIMENSION EL(13), TQ(4), Y(NO, N)
COMMON/GEAR1/T, H, HMIN, HMAX, EPS, UROUND, N, MF,
 KFLAG, JSTART
COMMON/GEAR2/YMAX(1)
COMMON/GEAR3/ERROR(1)
COMMON/GEAR4/SAVE1(1)
COMMON/GEAR5/SAVE2(1)
COMMON/GEAR6/PW(1)
COMMON/GEAR7/IPIV(1)
COMMON/GEAR9/HUSED, NQUSED, NSTEP, NFE, NJE
DATA EL(2)/1./, OLDLO/1./
KFLAG=0
TOLD=T
IF(JSTART.GT.0) GO TO 100
IF(JSTART.NE.0) GO TO 20
```

```
C-----C
C ПРИ ПЕРШОМУ ЗВЕРНЕННІ ДО П/П STIFF ПОРЯДОК МЕТОДУ C
C ДОРІВНЮЄ ОДИНИЦІ Й ОБЧИСЛЮЮТЬСЯ ПОЧАТКОВІ ЗНАЧЕН- C
C НЯ YDOT. RMAX - ПАРАМЕТР, ЯКИЙ ПОКАЗУЄ, В СКІЛЬКИ C
C РАЗІВ МОЖЕ БУТИ ЗБІЛЬШЕНЕ ЗНАЧЕННЯ Н НА КРОЦІ. C
C ВРАХОВУЮЧИ МАЛІ ВИХІДНІ ЗНАЧЕННЯ Н, СПОЧАТКУ C
C RMAX = 1.E4. ДАЛІ RMAX = 10. ПРИ НЕВДАЛОМУ ЗАВЕР- C
C ШЕННІ КРОКУ (ВІДСУТНОСТІ ЗБІЖНОСТІ ПРИ РОЗВ'ЯЗУ- C
C ВАННІ РІВНЯНЬ КОРЕКТОРА АБО НЕВИКОНАННІ ТЕСТУ C
C ПОХИБКИ) RMAX ВСТАНОВЛЮЄТЬСЯ НА ЗНАЧЕННЯ 2. C
C-----C
```

```
CALL DIFFUN(N, T, Y, SAVE1)
DO 10 I=1, N
10 Y(I, 2)=H*SAVE1(I)
METH=MF/10
MITER=MF-10*METH
NQ=1
L=2
IDOUB=3
RMAX=1.E4
RC=0.
CRATE=1.
EPSOLD=EPS
HOLD=H
MFOLD=MF
NOLD=N
NSTEP=0
NSTEPJ=0
NFE=1
```

NJE=0  
IRET=1  
GO TO 30

-----C  
C У РАЗІ ЗМІНИ ВХІДНОГО ПАРАМЕТРА METH ВИКЛИКА- C  
C ЄТЬСЯ П/П COSET ДЛЯ ВИЗНАЧЕННЯ КОЕФІЦІЄНТІВ МЕТО- C  
C ДУ. ПРИ ЗМІНІ ПРОГРАМІСТОМ ПАРАМЕТРІВ N, EPS АБО C  
C METH ПЕРЕВИЗНАЧАЮТЬСЯ ЗНАЧЕННЯ E, EDN, EUP І BND. C  
C ПАРАМЕТРИ E, EDN І EUP ВИКОРИСТОВУЮТЬСЯ ДЛЯ ОЦІНКИ C  
C ЛОКАЛЬНОЇ ПОХИБКИ ДИСКРЕТИЗАЦІЇ МЕТОДУ З ПОРЯДКОМ C  
C ТОЧНОСТІ АПРОКСИМАЦІЇ ВІДПОВІДНО NQ, NQ-1 ТА NQ+1. C  
C ПАРАМЕТР BND ВИКОРИСТОВУЄТЬСЯ ДЛЯ ВИЗНАЧЕННЯ C  
C ЗБІЖНОСТІ ПРИ РОЗВ'ЯЗУВАННІ НЕЯВНИХ РІВНЯНЬ КОРЕК- C  
C ТОРА. ЕЛЕМЕНТИ МАСИВУ Y ЗВОДЯТЬСЯ ДО НОВОГО КРОК- C  
C КУ, ЯКЩО ПРИ ЗВЕРНЕННІ ДО П/П STIFF ЗНАЧЕННЯ N ЗМІ- C  
C НИЛОСЬ. ПРИ НОВИХ ЗНАЧЕННЯХ N АБО METH ЛІЧИЛЬНИК C  
C IDOUB ВСТАНОВЛЮЄТЬСЯ НА ЗНАЧЕННЯ L+1 ДЛЯ НАСТУП- C  
C НОЇ ПІДТРИМКИ КРОКУ НЕЗМІННИМ L+1 РАЗІВ ПІДРЯД. C  
-----C

20 IF(MF.EQ.MFOLD) GO TO 50  
MEO=METH  
MIO=MITER  
METH=MF/10  
MITER=MF-10\*METH  
MFOLD=MF  
IF(MITER.NE.MIO) IWEVAL=MITER  
IF(METH.EQ.MEO) GO TO 50  
IDOUB=L+1  
IRET=1  
30 CALL COSET(METH,NQ,EL,TQ,MAXDER)  
LMAX=MAXDER+1  
RC=RC\*EL(1)/OLDLO  
OLDLO=EL(1)  
40 FN=FLOAT(N)  
EDN=FN\*(TQ(1)\*EPS)\*\*2  
E=FN\*(TQ(2)\*EPS)\*\*2  
EUP=FN\*(TQ(3)\*EPS)\*\*2  
BND=FN\*(TQ(4)\*EPS)\*\*2  
GO TO (60,70,100),IRET  
50 IF((EPS.EQ.EPSOLD).AND.(N.EQ.NOLD)) GO TO 60  
EPSOLD=EPS  
NOLD=N  
IRET=1  
GO TO 40  
60 IF(H.EQ.HOLD) GO TO 100  
RH=H/HOLD  
H=HOLD  
IREDO=3  
GO TO 80  
70 RH=AMAX1(RH,HMIN/ABS(H))  
80 RH=AMIN1(RH,HMAX/ABS(H),RMAX)  
R1=1.

```

DO 90 J=2, L
 R1=R1*RH
 DO 90 I=1, N
 Y(I, J)=Y(I, J)*R1
90
 H=H*RH
 RC=RC*RH
 IDOUB=L+1
 IF(IREDO.EQ.0) GO TO 570

```

```

-----C
C У ЦЬОЇ ЧАСТИНИ ПРОГРАМИ ПРОГНОЗУЮТЬСЯ ЗНАЧЕННЯ Y C
C ЗА ДОПОМОГОЮ МНОЖЕННЯ МАСИВУ Y НА ТРИКУТНУ C
C МАТРИЦЮ ПАСКАЛЯ. C
C RC - МНОЖНИК, ЯКИЙ ВИЗНАЧАЄ, В СКІЛЬКИ РАЗІВ НОВЕ C
C ЗНАЧЕННЯ H*EL(1) ПЕРЕВИШУЄ ПОПЕРЕДНЄ. ПРИ C
C ЗМІНІ ВХІДНОГО ПАРАМЕТРА MITER АБО ВІДХИЛЕН- C
C НІ ЗНАЧЕННЯ RC ВІД ОДИНИЦІ БІЛЬШЕ, НІЖ НА 30 C
C ПРОЦЕНТІВ, ОЗНАКА IWEVAL ДОРІВНЮЄ ЗНАЧЕННЮ C
C MITER. У ЦЬОМУ ВИПАДКУ ДЛЯ ЗНАЧЕННЯ MITER, C
C ВІДМІННОГО ВІД НУЛЯ, РОБИТЬСЯ ПЕРЕВИЗНАЧЕННЯ C
C МАТРИЦІ ЯКОБІ ПРАВОЇ ЧАСТИНИ. КОЖНОГО РАЗУ, C
C КОЛИ MITER НЕ ДОРІВНЮЄ НУЛЮ, ОБЧИСЛЕННЯ МАТ- C
C РИЦІ ЯКОБІ ПРОВОДИТЬСЯ НЕ МЕНШЕ ОДНОГО C
C РАЗУ ЗА 20 КРОКІВ. C
-----C

```

```

100 IF (ABS(RC-1.) .GT. 0.3) IWEVAL=MITER
 IF (NSTEP.GE.NSTEPJ+20) IWEVAL=MITER
 T=T+H
 DO 110 J1=1, NQ
 DO 110 J2=J1, NQ
 J=(NQ+J1)-J2
 DO 110 I=1, N
 Y(I, J)=Y(I, J)+Y(I, J+1)
110

```

```

-----C
C ВИКОНУЄТЬСЯ ДО ТРЬОХ ІТЕРАЦІЙ РОЗВ'ЯЗУВАННЯ НЕЯВН- C
C ИХ РІВНЯНЬ КОРЕКТОРА. ДЛЯ КОЖНОЇ ІТЕРАЦІЇ ЗБІЖНІСТЬ C
C ВИЗНАЧАЄТЬСЯ ЗА ЕВКЛІДОВОЮ НОРМОЮ З ВИКОРИСТАННЯМ C
C ПАРАМЕТРА BND, ЯКИЙ ЗАЛЕЖИТЬ ВІД EPS. СУМАРНА ПО- C
C ХИБКА НАКОПИЧУЄТЬСЯ В МАСИВІ ERROR(I). В ПРОЦЕСІ C
C РОЗВ'ЯЗУВАННЯ РІВНЯНЬ КОРЕКТОРА УТОЧНЕНІ ЗНАЧЕННЯ C
C Y ТИМЧАСОВО ЗАПАМ'ЯТОВУЮТЬСЯ В МАСИВІ SAVE1, А C
C САМ МАСИВ Y ЗБЕРІГАЄ ПРОГНОЗОВАНІ ЗНАЧЕННЯ. C
-----C

```

```

120 DO 130 I=1, N
130 ERROR(I)=0.
 M=0
 CALL DIFFUN(N, T, Y, SAVE2)
 NFE=NFE+1
 IF(IWEVAL.LE.0) GO TO 190

```

```

-----C
C ОЗНАКА IWEVAL МОЖЕ ВКАЗУВАТИ НА НЕОБХІДНІСТЬ C
C ПОПЕРЕДНЬОГО ОБЧИСЛЕННЯ МАТРИЦІ ІТЕРАЦІЇ C
C P = I - H*EL(1)*J . ЗНАЧЕННЯ IWEVAL=0 ВКАЗУЄ НА C

```

C ВІДСУТНІСТЬ ПОТРЕБИ ОБЧИСЛЕННЯ P. ДЛЯ ВХІДНИХ C  
C ПАРАМЕТРІВ MITER=1 АБО 2 ОБЧИСЛЕННЯ МАТРИЦІ C  
C P ПРОВОДИТЬСЯ ЗА П/П PSET . ПРИ ЗАДАННІ MITER=3 C  
C ВИКОРИСТОВУЄТЬСЯ МАТРИЦЯ  $P = I - EL(1)*D$ , ДЕ C  
C D - ДІАГОНАЛЬНА МАТРИЦЯ, I - ОДИНИЧНА МАТРИЦЯ. C  
C-----C

IWEVAL=0  
RC=1.  
NJE=NJE+1  
NSTEPJ=NSTEP  
GO TO (150, 140, 160), MITER  
140 NFE=NFE+N  
150 CON=-H\*EL(1)  
CALL PSET(Y, NO, CON, MITER, IER)  
IF (IER.NE.0) GO TO 310  
GO TO 250  
160 R=EL(1)\*.1  
DO 170 I=1, N  
170 PW(I)=Y(I, 1)+R\*(H\*SAVE2(I)-Y(I, 2))  
CALL DIFFUN(N, T, PW, SAVE1)  
NFE=NFE+1  
HLO=H\*EL(1)  
DO 180 I=1, N  
RO=H\*SAVE2(I)-Y(I, 2)  
PW(I)=1.  
D=.1\*RO-H\*(SAVE1(I)-SAVE2(I))  
SAVE1(I)=0.  
IF (ABS(RO).LT.UROUND\*YMAX(I)) GO TO 180  
IF (ABS(D).EQ.0.) GO TO 310  
PW(I)=.1\*RO/D  
SAVE1(I)=PW(I)\*RO  
180 CONTINUE  
GO TO 270  
190 IF (MITER.NE.0) GO TO (250, 250, 210), MITER

C-----C  
C У РАЗІ ФУНКЦІОНАЛЬНИХ ІТЕРАЦІЙ ПЕРЕВИЗНАЧЕННЯ Y C  
C ЗДІЙСНЮЄТЬСЯ БЕЗПОСЕРЕДНЬО ЗА ДОПОМОГОЮ П/П DIFFUN. C  
C-----C

D=0.  
DO 200 I=1, N  
R=H\*SAVE2(I)-Y(I, 2)  
D=D+((R-ERROR(I))/YMAX(I))\*\*2  
SAVE1(I)=Y(I, 1)+EL(1)\*R  
200 ERROR(I)=R  
GO TO 290

C-----C  
C МЕТОД НЬЮТОНА ЗВОДИТЬСЯ ДО РОЗВ'ЯЗУВАННЯ СЛАР, ДЕ C  
C ЯК МНОЖНИКИ ЗАСТОСОВУЄТЬСЯ МАТРИЦЯ P, РОЗКЛАДЕНА C  
C ЗА ДОПОМОГОЮ LU-ДЕКОМПОЗИЦІЮ, КОЛИ MITER=1 АБО 2, C  
C I ВЕКТОР НЕВ'ЯЗКИ КОРЕКТОРА ПРИ ОБЧИСЛЕННІ ПРАВОЇ C  
C ЧАСТИНИ F(Y, T). ЯКЩО MITER=3, ТО ДЛЯ МАСИВУ C  
C P ЗНАЧЕННЯ H\*EL(1) ПЕРЕВИЗНАЧАЄТЬСЯ. C  
C-----C

```

210 PHLO=HLO
 HLO=H*EL(1)
 IF(HLO.EQ.PHLO) GO TO 230
 R=HLO/PHLO
 DO 220 I=1,N
 D=1.-R*(1.-1./PW(I))
 IF(ABS(D).EQ.0.) GO TO 330
220 PW(I)=1./D
230 DO 240 I=1,N
240 SAVE1(I)=PW(I)*(H*SAVE2(I)-(Y(I,2)+
 * ERROR(I)))
 GO TO 270
250 DO 260 I=1,N
260 SAVE1(I)=H*SAVE2(I)-(Y(I,2)+ERROR(I))
 CALL SOL(N,NO,PW,SAVE1,IPIV)
270 D=0.
 DO 280 I=1,N
 ERROR(I)=ERROR(I)+SAVE1(I)
 D=D+(SAVE1(I)/YMAX(I))*2
280 SAVE1(I)=Y(I,1)+EL(1)*ERROR(I)

```

-----C  
C ПЕРЕВІРКА УМОВИ ЗБІЖНОСТІ. ДЛЯ ЗНАЧЕНЬ  $M > 0$  C  
C ПРОВІДИТЬСЯ ОБЧИСЛЕННЯ КОНСТАНТИ ШВИДКОСТІ ЗБІЖ- C  
C НОСТІ CRATE. C  
C-----C

```

290 IF(M.NE.0) CRATE=AMAX1(.9*CRATE,D/D1)
 IF((D*AMIN1(1.,2.*CRATE)).LE.BND) GO TO 340
 D1=D
 M=M+1
 IF(M.EQ.3) GO TO 300
 CALL DIFFUN(N,T,SAVE1,SAVE2)
 GO TO 190

```

-----C  
C ЗБІЖНІСТЬ ПРИ РОЗВ'ЯЗУВАННІ РІВНЯНЬ КОРЕКТОРА ЗА C  
C З ІТЕРАЦІЇ НЕ ДОСЯГНУТО. ЯКЩО ПРИ ОБЧИСЛЕННЯХ ВИ- C  
C КОРИСТОВАЛОСЬ ПОПЕРЕДНЄ ЗНАЧЕННЯ МАТРИЦІ ЯКОБІ C  
C J, ТО ДЛЯ ДАЛЬШИХ ІТЕРАЦІЙ ПРОВІДИТЬСЯ ЇЇ ПЕРЕ- C  
C ВИЗНАЧЕННЯ. В РЕШТІ ВИПАДКІВ ВІДТВОРЮЮТЬСЯ C  
C ЗНАЧЕННЯ ЕЛЕМЕНТІВ МАСИВУ Y, ЯКІ ВІДПОВІДАЮТЬ C  
C КІНЦЮ ПОПЕРЕДНЬОГО КРОКУ. ПРОВІДИТЬСЯ ЗМЕНШЕННЯ C  
C ЗНАЧЕННЯ N. ЯКЩО  $N < N_{MIN}$ , ТО СТВОРЮЄТЬСЯ ОЗНАКА C  
C АВАРІЙНОГО ЗАВЕРШЕННЯ (KFLAG=-3). C  
C-----C

```

300 NFE=NFE+2
 IF(IWEVAL.EQ.-1) GO TO 330
310 T=TOLD
 RMAX=2.
 DO 320 J1=1,NQ
 DO 320 J2=J1,NQ
 J=(NQ+J1)-J2
 DO 320 I=1,N
 Y(I,J)=Y(I,J)-Y(I,J+1)
320 IF(ABS(H).LE.HMIN*1.00001) GO TO 560

```

RH=.25  
IREDO=1  
GO TO 70  
IWEVAL=MITER  
GO TO 120

330

-----C  
C ЗБІЖНІСТЬ РОЗВ'ЯЗКУ РІВНЯНЬ КОРЕКТОРА ДОСЯГНУТО. C  
C ЯКЩО ПРИ ЦЬОМУ В ОБЧИСЛЕННЯХ ВИКОРИСТОВУВАЛАСЬ C  
C МАТРИЦЯ ЯКОБІ J, ТО ОЗНАКА IWEVAL ВСТАНОВЛЮЄТЬСЯ C  
C НА -1, ЩО ВКАЗУЄ НА МОЖЛИВІСТЬ НАСТУПНОГО C  
C ПЕРЕВИЗНАЧЕННЯ МАТРИЦІ J. ПРОВІДИТЬСЯ ТЕСТ ПО- C  
C ХИБКИ. ЯКЩО ЦЕЙ ТЕСТ НЕ ВИКОНУЄТЬСЯ, ТО ЗДІЙ- C  
C СНЮЄТЬСЯ ПЕРЕХІД ДО ОПЕРАТОРА З ПОЗНАЧКОЮ 380. C  
C-----C

340 IF(MITER.NE.0) IWEVAL=-1  
NFE=NFE+M  
D=0.  
DO 350 I=1,N  
350 D=D+(ERROR(I)/YMAX(I))\*2  
IF(D.GT.E) GO TO 380

-----C  
C ТЕСТ ПОХИБКИ ВИКОНУЄТЬСЯ (КРОК БУВ УСПІШНИМ). C  
C ПЕРЕВИЗНАЧАЮТЬСЯ ЕЛЕМЕНТИ МАСИВУ Y. C  
C ПРИ ПОТОЧНОМУ ЗНАЧЕННІ ЛІЧИЛЬНИКА IDOUB=1 РОЗ- C  
C ГЛЯДАЄТЬСЯ МОЖЛИВІСТЬ ЗМІНИ КРОКУ H ДЛЯ ЛІНІЙНИХ C  
C ФОРМУЛ ІНТЕГРУВАННЯ З ПОРЯДКОМ ТОЧНОСТІ NQ-1, NQ, C  
C NQ+1. ЯКЩО ЗНАЙДЕНЕ ЗНАЧЕННЯ ВІДРІЗНЯЄТЬСЯ ВІД ПО- C  
C ПЕРЕДНЬОГО H МЕНШ, НІЖ В 1.1 РАЗА, ТО ЗНАЧЕННЯ H C  
C НЕ ЗМІНЮЄТЬСЯ, ЛІЧИЛЬНИК IDOUB ВСТАНОВЛЮЄТЬСЯ C  
C НА 10, ЛІКВІДУЮЧИ МОЖЛИВІСТЬ ВІДПОВІДНИХ ОЦІНОК C  
C ПРОТЯГОМ НАЙБЛИЖЧИХ 10 КРОКІВ. ДЛЯ ПОТОЧНОГО C  
C ЗНАЧЕННЯ IDOUB > 1 ЛІЧИЛЬНИК ЗМЕНШУЄТЬСЯ НА C  
C ОДИНИЦЮ. ЯКЩО ПРИ ЦЬОМУ ЗНАЧЕННЯ IDOUB ДОРІВНЮЄ C  
C 1 І ПОТОЧНИЙ ПОРЯДОК МЕТОДУ NQ < MAXDER, ТО C  
C ВЕКТОР НЕВ'ЯЗОК ERROR ЗАПАМ'ЯТОВУЄТЬСЯ ДЛЯ МОЖЛИ- C  
C ВОГО ЗБІЛЬШЕННЯ NQ НА НАСТУПНОМУ КРОЦІ. C  
C-----C

KFLAG=0  
IREDO=0  
NSTEP=NSTEP+1  
HUSED=H  
NQUSED=NQ  
DO 360 J=1,L  
DO 360 I=1,N  
360 Y(I,J)=Y(I,J)+EL(J)\*ERROR(I)  
IF(IDOUB.EQ.1) GO TO 400  
IDOUB=IDOUB-1  
IF(IDOUB.GT.1) GO TO 580  
IF(L.EQ.LMAX) GO TO 580  
DO 370 I=1,N  
370 Y(I,LMAX)=ERROR(I)  
GO TO 580

С-----С  
С ТЕСТ ПОХИБКИ НЕ ВИКОНУЄТЬСЯ. ОЗНАКА KFLAG НАКО- С  
С ПИЧУЄ ІНФОРМАЦІЮ ПРО ПОВТОРНІ НЕВДАЛІ СПРОБИ. С  
С ЗМІННІ Т І У НАБУВАЮТЬ СВОЇХ ПОЧАТКОВИХ ЗНАЧЕНЬ. С  
С ПРОВОДИТЬСЯ ОЦІНКА КРОКУ Н ДЛЯ ЛІНІЙНИХ С  
С ФОРМУЛ ПОРЯДКУ NQ І NQ-1 . С

380 KFLAG=KFLAG-1  
T=TOLD  
DO 390 J1=1,NQ  
DO 390 J2=J1,NQ  
J=(NQ+J1)-J2  
DO 390 I=1,N  
390 Y(I,J)=Y(I,J)-Y(I,J+1)  
RMAX=2.  
IF(ABS(H).LE.HMIN\*1.00001) GO TO 540  
IF(KFLAG.LE.-3) GO TO 520  
IREDO=2  
PR3=1.E+20  
GO TO 420

С-----С  
С ЯК ПРИ ВДАЛОМУ, ТАК І ПРИ НЕВДАЛОМУ ВИКОНАННІ С  
С КРОКУ (ВИКОНАННЯ АБО НЕВИКОНАННЯ ТЕСТУ ПОХИБКИ) С  
С ОБЧИСЛЮЮТЬСЯ ЗНАЧЕННЯ PR1, PR2 І PR3 - КОЕФІЦІЄН- С  
С ТІВ ПРОГНОЗУВАННЯ КРОКУ ДЛЯ ФОРМУЛ ІНТЕГРУВАННЯ С  
С ПОРЯДКУ NQ-1, NQ І NQ+1 ВІДПОВІДНО. С  
С ПРИ НЕВИКОНАННІ ТЕСТУ ПОХИБКИ ПОКЛАДАЄТЬСЯ С  
С PR3 = 1.E20 І ФОРМУЛИ ПОРЯДКУ NQ+1 НЕ РОЗГЛЯ- С  
С ДАЮТЬСЯ. ВИБИРАЮТЬСЯ НАЙМЕНШЕ ЗІ ЗНАЧЕНЬ PR1, PR2 С  
С АБО PR3 І ФОРМУЛИ ІНТЕГРУВАННЯ ВІДПОВІДНОГО ПО- С  
С РЯДКУ ТОЧНОСТІ. ПРИ ВИБОРІ ЗНАЧЕННЯ PR3 (ПОРЯДОК С  
С ТОЧНОСТІ NQ+1) ДЛЯ КОЖНОЇ ЗМІННОЇ Y(I) ОБЧИСЛЮ- С  
С ЄТЬСЯ ОДНА ДОДАТКОВА ПОХІДНА. С

400 PR3=1.E+20  
IF(L.EQ.LMAX) GO TO 420  
D1=0.  
DO 410 I=1,N  
410 D1=D1+((ERROR(I)-Y(I,LMAX))/YMAX(I))\*\*2  
ENQ3=.5/FLOAT(L+1)  
PR3=((D1/EUP)\*\*ENQ3)\*1.4+1.4E-6  
420 ENQ2=.5/FLOAT(L)  
PR2=((D/E)\*\*ENQ2)\*1.2+1.2E-6  
PR1=1.E+20  
IF(NQ.EQ.1) GO TO 440  
D=0.  
DO 430 I=1,N  
430 D=D+(Y(I,L)/YMAX(I))\*\*2  
ENQ1=.5/FLOAT(NQ)  
PR1=((D/EDN)\*\*ENQ1)\*1.3+1.3E-6  
440 IF(PR2.LE.PR3) GO TO 450  
IF(PR3.LT.PR1) GO TO 470

```

GO TO 460
450 IF (PR2.GT.PR1) GO TO 460
NEWQ=NQ
RH=1./PR2
GO TO 500
460 NEWQ=NQ-1
RH=1./PR1
GO TO 500
470 NEWQ=L
RH=1./PR3
IF (RH.LT.1.1) GO TO 490
DO 480 I=1,N
480 Y(I,NEWQ+1)=ERROR(I)*EL(L)/FLOAT(L)
GO TO 510
490 IDOUB=10
GO TO 580
500 IF ((KFLAG.EQ.0).AND.(RH.LT.1.1)) GO TO 490

```

```

C-----C
C ПРИ ЗМІНІ ПОРЯДКУ ТОЧНОСТІ ФОРМУЛ ІНТЕГРУВАННЯ C
C ПЕРЕВИЗНАЧАЮТЬСЯ ЗНАЧЕННЯ NQ, L І КОЕФІЦІЄНТИ C
C EL У П/П COSET, COAD. У БУДЬ-ЯКОМУ ВИПАДКУ C
C ОБЧИСЛЮЄТЬСЯ НОВЕ ЗНАЧЕННЯ КРОКУ H ЗА ДОПОМОГОЮ C
C МНОЖНИКА RH І ВІДПОВІДНО ПЕРЕВИЗНАЧАЮТЬСЯ C
C ЕЛЕМЕНТИ МАСИВУ Y. C
C ЯКЩО ТЕСТ ПОХИБКИ ВИКОНАВСЯ РАНІШЕ, ТО ПРОВОДИТЬ- C
C СЯ ВИХІД З П/П STIFF, А ЯКЩО НІ - КРОК ПОВТОРЮ- C
C ЄТЬСЯ З НОВИМ ЗНАЧЕННЯМ H. C
C-----C

```

```

IF (NEWQ.EQ.NQ) GO TO 70
510 NQ=NEWQ
L=NQ+1
IRET=2
GO TO 30

```

```

C-----C
C У РАЗІ НЕВИКОНАННЯ ТЕСТУ ПОХИБКИ ТРИ РАЗИ ПІД- C
C РЯД КЕРУВАННЯ ПЕРЕДАЄТЬСЯ ДАНОМУ МІСЦЮ ПРОГРАМИ. C
C ВВАЖАЄТЬСЯ, ЩО ПОХИБКИ, НАКОПИЧЕНІ ПІД ЧАС ОБ- C
C ЧИСЛЕННЯ ПОХІДНИХ Y, ПРИЗВОДЯТЬ ДО "ВИКРИВЛЕННЯ" C
C РЕЗУЛЬТАТУ. ПОРЯДОК МЕТОДУ ДОРІВНЮЄ 1, ПЕРЕВИЗНА- C
C ЧАЄТЬСЯ ЗНАЧЕННЯ ПЕРШОЇ ПОХІДНОЇ Y. ЗНАЧЕННЯ H C
C ЗМЕНШУЄТЬСЯ В 10 РАЗІВ І КРОК ПОВТОРЮЄТЬСЯ. C
C ПРИ СЕМИКРАТНОМУ НЕВИКОНАННІ ТЕСТУ ПОХИБКИ C
C ЗДІЙСНЮЄТЬСЯ АВАРІЙНЕ ЗАВЕРШЕННЯ П/П STIFF. C
C ОЗНАКА KFLAG ДОРІВНЮЄ -2. C
C-----C

```

```

520 IF (KFLAG.EQ.-7) GO TO 550
RH=.1
RH=AMAX1(HMIN/ABS(H),RH)
H=H*RH
CALL DIFFUN(N,T,Y,SAVE1)
NFE=NFE+1
DO 530 I=1,N

```



```

530 Y(I, 2)=H*SAVE1(I)
 IWEVAL=MITER
 IDOUB=10
 IF(NQ.EQ.1) GO TO 100
 NQ=1
 L=2
 IRET=3
 GO TO 30

```

```

C-----C
C ЗДІЙСНЮЮТЬСЯ ВСІ ВИХОДИ З П/П STIFF. ЗНАЧЕННЯ Н C
C ЗАПАМ'ЯТОВУЄТЬСЯ ПІД ІМЕНЕМ HOLD. ЦЕ ДАЄ ЗМОГУ C
C ТОМУ, ХТО КОРИСТУЄТЬСЯ П/П STIFF, ЗМІНЮВАТИ ЗНА- C
C ЧЕННЯ Н НА ЧЕРГОВОМУ КРОЦІ (ДИВ. П/П DRIVE). C
C-----C

```

```

540 KFLAG=-1
 GO TO 580
550 KFLAG=-2
 GO TO 580
560 KFLAG=-3
 GO TO 580
570 RMAX=10.
580 HOLD=N
 JSTART=NQ
 RETURN

```

```

C-----C КІНЕЦЬ П/П STIFF -----C
 END

```

#### SUBROUTINE COSET(METH, NQ, EL, TQ, MAXDER)

```

CC
C П/П COSET ВИКЛИКАЄТЬСЯ ІНТЕГРАТОРОМ STIFF І C
C ВСТАНОВЛЮЄ ДЛЯ НЬОГО ЗНАЧЕННЯ КОЕФІЦІЕНТІВ. ВЕК- C
C ТОР EL ЗАВДОВЖКИ NQ+1 ВИЗНАЧАЄ МЕТОД ІНТЕГРУВАННЯ. C
C ВЕКТОР TQ ЗАВДОВЖКИ 4 ВИКОРИСТОВУЄТЬСЯ ДЛЯ ВИ- C
C ЗНАЧЕННЯ КРОКУ ВІДПОВІДНО ДО ДОПУСТИМОГО ЗНА- C
C ЧЕННЯ ЛОКАЛЬНОЇ ПОХИБКИ ДИСКРЕТИЗАЦІЇ МЕТОДУ. C
C ЕЛЕМЕНТИ TQ(1), TQ(2) І TQ(3) ЗАДАЮТЬСЯ В C
C МАСИВІ PERTST. ЗНАЧЕННЯ EL І TQ ЗАЛЕЖАТЬ ВІД C
C ПАРАМЕТРІВ METH І NQ. П/П COSET ВИЗНАЧАЄ ТАКОЖ C
C ПАРАМЕТР MAXDER - МАКСИМАЛЬНИЙ ПОРЯДОК ФОРМУЛ C
C ІНТЕГРУВАННЯ, ЩО ЗАСТОСОВУЮТЬСЯ. В ЦІЙ ВЕРСІЇ ВИ- C
C КОРИСТОВУЮТЬСЯ ФОРМУЛИ АДАМСА ПОРЯДКУ 1-12, ФОР- C
C МУЛИ ГІРА ПОРЯДКУ 1 - 5 І МОДИФІКОВАНІ ФОРМУЛИ C
C ГІРА ПОРЯДКУ 4 - 6. ЗНАЧЕННЯ MAXDER МОЖУТЬ БУТИ C
C ЗМЕНШЕНІ ВІДПОВІДНОЮ ЗАМІНОЮ ОПЕРАТОРІВ З ПО- C
C ЗНАЧКАМИ 10, 20 І 30. C
C ЗНАЧЕННЯ LMAX=MAXDER+1 ВИЗНАЧАЄ КІЛЬКІСТЬ СТОВП- C
C ЦІВ ДВОВИМІРНОГО МАСИВУ Y, ЯКІ ВИКОРИСТОВУЮТЬСЯ C
C ДЛЯ ДАЛЬШИХ ОБЧИСЛЕНЬ В П/П STIFF. C
C П/П COSET ВИКЛИКАЄ П/П COAD ДЛЯ ВИЗНАЧЕННЯ C
C ЗНАЧЕНЬ ВЕКТОРА EL, ЩО ВІДПОВІДАЮТЬ ФОРМУЛАМ C
C АДАМСА. ВСІ ЗНАЧЕННЯ КОЕФІЦІЕНТІВ EL(I) ПОДАНО C
C З МАКСИМАЛЬНОЮ ДЛЯ ДАНОГО КОМП'ЮТЕРА ТОЧНІСТЮ. C
C ВИМОГИ ДО ТОЧНОСТІ КОЕФІЦІЕНТІВ МАСИВУ PERTST C

```

С НЕ ТАКІ ВИСОКІ - ПОРЯДКУ ОДНОГО ПРОЦЕНТА. ЕЛЕМЕН- С  
 С ТИ PERTST РОЗМІЩУЮТЬСЯ ГРУПАМИ ВІДПОВІДНО ДО ПО- С  
 С РЯДКІВ ТОЧНОСТІ, NQ-1, NQ, NQ+1. С  
 С ВСЕРЕДИНИ КОЖНОЇ ГРУПИ ЗНАХОДЯТЬСЯ КОЕФІЦІЄНТИ С  
 С ФОРМУЛ АДАМСА, ГІРА ТА МОДИФІКОВАНИХ ФОРМУЛ ГІРА. С  
 ССССССССССССССССССССССССССССССССССССССССССССССССССССССССС

```

DIMENSION PERTST(12, 3, 3), EL(13), TQ(4)
DATA PERTST/1., 1., 2., 1., 3158., 07407., 01391.,
* .002182, .0002945, .00003492, .000003692,
* .0000003524, 1., 1., 5., 1667., 04167, 1., 1., 1., 1.,
* 1., 1., 1., 1., 1., 5., 1667., 04167., 008267, 1., 1.,
* 1., 1., 1., 1., 2., 12., 24., 37.89, 53.33, 70.08,
* 87.97, 106.9, 126.7, 147.4, 168.8, 191.0,
* 2.0, 4.5, 7.333, 10.42, 13.7, 1., 1., 1., 1., 1., 1., 1.,
* 2.0, 4.5, 7.333, 7.58, 10.7, 13.94, 1., 1., 1., 1., 1.,
* 1., 12.0, 24.0, 37.89, 53.33, 70.08, 87.97, 106.9,
* 126.7, 147.4, 168.8, 191.0, 1.,
* 3.0, 6.0, 9.167, 12.5, 1., 1., 1., 1., 1., 1., 1.,
* 3.0, 6.0, 9.167, 9.096, 12.48, 1., 1., 1., 1., 1., 1., /
GO TO (10, 20, 30), METH
10 MAXDER=12
 CALL COAD(EL, NQ)
 GO TO 120
20 MAXDER=5
 GO TO (40, 50, 60, 70, 80), NQ
30 MAXDER=6
 GO TO (40, 50, 60, 90, 100, 110), NQ

```

С-----С  
 С КОЕФІЦІЄНТИ НЕЯВНИХ ФОРМУЛ ГІРА МОЖУТЬ БУТИ ОБ- С  
 С ЧИСЛЕНІ ТАКОЖ ЗА ДОПОМОГОЮ ГЕНЕРУЮЧОГО ПОЛІНОМА С  
 С L(T) З ВИКОРИСТАННЯМ ЗАЛЕЖНОСТЕЙ: С  
 С  $L(T) = EL(1) + EL(2)*T + \dots + EL(NQ+1)*T**NQ$  ; С  
 С  $DL/DT = (T+1)*(T+2)* \dots *(T+NQ)/K$  С  
 С ДЕ K = FACTORIAL(NQ)\*(1 + 1/2 + ... + 1/NQ) С

```

40 EL(1)=1.0
 GO TO 120
50 EL(1)=6.666666666666666E-01
 EL(3)=3.333333333333333E-01
 GO TO 120
60 EL(1)=5.454545454545455E-01
 EL(3)=EL(1)
 EL(4)=9.0909090909091E-02
 GO TO 120
70 EL(1)=4.8E-01
 EL(3)=7.0E-01
 EL(4)=2.0E-01
 EL(5)=2.0E-02
 GO TO 120
80 EL(1)=4.3795620437956E-01
 EL(3)=8.2116788321168E-01
 EL(4)=3.1021897810219E-01

```

EL(5)=5.4744525547445E-02  
EL(6)=3.6496350364964E-03  
GO TO 120

-----C  
C НИЖЧЕ ПОДАНО КОЕФІЦІЄНТИ НЕЯВНИХ МОДИФІКОВАНИХ C  
C ФОРМУЛ ГІРА З ПОРЯДКУ 4, 5 І 6. ОДЕРЖАНО ІХ ЗА C  
C ДОПОМОГОЮ ЕКВІВАЛЕНТНИХ МАТРИЧНИХ ПЕРЕТВОРЕНЬ. C  
-----C

90 EL(1)=5.3811659192825E-01  
EL(3)=5.6278026905830E-01  
EL(4)=1.0313901345291E-01  
EL(5)=2.2421524663677E-03  
GO TO 120

100 EL(1)=4.7543581616482E-01  
EL(3)=7.1315372424723E-01  
EL(4)=2.1196513470681E-01  
EL(5)=2.3771790808241E-02  
EL(6)=3.9619651347068E-04  
GO TO 120

110 EL(1)=4.3478260869565E-01  
EL(3)=8.3176328502415E-01  
EL(4)=3.2155797101449E-01  
EL(5)=5.9480676328502E-02  
EL(6)=4.5289855072464E-03  
EL(7)=6.0386473429952E-05

120 DO 130 K=1,3

130 TQ(K)=PERTST(NQ, METH, K)

TQ(4)=0.5/FLOAT(NQ+2)

RETURN

C----- КІНЕЦЬ П/П COSET -----C  
END

SUBROUTINE COAD(EL, NQ)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C П/П COAD ВИЗНАЧАЄ КОЕФІЦІЄНТИ EL(I) ДЛЯ НЕЯВНИХ C  
C ФОРМУЛ АДАМСА ПОРЯДКУ NQ. ЦІ КОЕФІЦІЄНТИ МОЖУТЬ C  
C БУТИ ОБЧИСЛЕНІ ТАКОЖ ЗА ДОПОМОГОЮ ГЕНЕРУЮЧОГО C  
C ПОЛІНОМА L(T) З ВИКОРИСТАННЯМ ЗАЛЕЖНОСТЕЙ: C  
C  $L(T) = EL(1) + EL(2)*T + \dots + EL(NQ+1)*T**NQ ;$  C  
C  $DL/DT = (T+1)*(T+2)* \dots *(T+NQ-1)/K , L(-1) = 0 ;$  C  
C ДЕ K = FACTORIAL(NQ-1) C  
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

DIMENSION EL(1)

GO TO (10, 20, 30, 40, 50, 60, 70, 80, 90, 100, 110,

\* 120), NQ

10 EL(1)=1.0  
RETURN

20 EL(1)=0.5  
EL(3)=0.5  
RETURN

30 EL(1)=4.16666666666667E-01  
EL(3)=0.75

EL (4) = 1. 66666666666667E-01  
RETURN

40

EL (1) = 0. 375  
EL (3) = 9. 16666666666667E-01  
EL (4) = 3. 33333333333333E-01  
EL (5) = 4. 16666666666667E-02  
RETURN

50

EL (1) = 3. 48611111111111E-01  
EL (3) = 1. 04166666666667  
EL (4) = 4. 86111111111111E-01  
EL (5) = 1. 04166666666667E-01  
EL (6) = 8. 33333333333333E-03  
RETURN

60

EL (1) = 3. 29861111111111E-01  
EL (3) = 1. 14166666666667  
EL (4) = 0. 625  
EL (5) = 1. 77083333333333E-01  
EL (6) = 0. 025  
EL (7) = 1. 3888888888889E-03  
RETURN

70

EL (1) = 3. 1559193121693E-01  
EL (3) = 1. 225  
EL (4) = 7. 5185185185185E-01  
EL (5) = 2. 55208333333333E-01  
EL (6) = 4. 86111111111111E-02  
EL (7) = 4. 86111111111111E-03  
EL (8) = 1. 9841269841270E-04  
RETURN

80

EL (1) = 3. 0422453703704E-01  
EL (3) = 1. 2964285714286  
EL (4) = 8. 6851851851852E-01  
EL (5) = 3. 3576388888889E-01  
EL (6) = 7. 7777777777778E-02  
EL (7) = 1. 0648148148148E-02  
EL (8) = 7. 9365079365079E-04  
EL (9) = 2. 4801587301587E-05  
RETURN

90

EL (1) = 2. 9486800044092E-01  
EL (3) = 1. 3589285714286  
EL (4) = 9. 7655423280423E-01  
EL (5) = 0. 4171875  
EL (6) = 1. 1135416666667E-01  
EL (7) = 0. 01875  
EL (8) = 1. 9345238095238E-03  
EL (9) = 1. 1160714285714E-04  
EL (10) = 2. 7557319223986E-06  
RETURN

100

EL (1) = 2. 8697544642857E-01  
EL (3) = 1. 4144841269841  
EL (4) = 1. 0772156084656  
EL (5) = 4. 9856701940035E-01  
EL (6) = 0. 1484375

EL(7)=2.9060570987654E-02  
 EL(8)=3.7202380952381E-03  
 EL(9)=2.9968584656085E-04  
 EL(10)=1.3778659611993E-05  
 EL(11)=2.7557319223986E-07  
 RETURN

110

EL(1)=2.8018959644394E-01  
 EL(3)=1.4644841269841  
 EL(4)=1.1715145502646  
 EL(5)=5.7935819003527E-01  
 EL(6)=1.8832286155203E-01  
 EL(7)=4.1430362654321E-02  
 EL(8)=6.2111441798942E-03  
 EL(9)=6.2520667989418E-04  
 EL(10)=4.0417401528513E-05  
 EL(11)=1.5156525573192E-06  
 EL(12)=2.5052108385442E-08  
 RETURN

120

EL(1)=2.7426554003160E-01  
 EL(3)=1.5099386724387  
 EL(4)=1.2602711640212  
 EL(5)=6.5923418209877E-01  
 EL(6)=2.3045800264550E-01  
 EL(7)=5.5697246105232E-02  
 EL(8)=9.4394841269841E-03  
 EL(9)=1.1192749669312E-03  
 EL(10)=9.0939153439153E-05  
 EL(11)=4.8225308641975E-06  
 EL(12)=1.5031265031265E-07  
 EL(13)=2.0876756987868E-09  
 RETURN

C----- КІНЕЦЬ П/П COAD -----C  
 END

SUBROUTINE INTERP(TOUT, Y, NO, YO)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
 C ПІДПРОГРАМА INTERP ОБЧИСЛЮЄ ЗА ДОПОМОГОЮ МАСИВУ C  
 C ПЕРЕДІСТОРІЇ НОРДСІКА Y ІНТЕРПОЛЬОВАНІ ЗНАЧЕННЯ C  
 C ЗАЛЕЖНОЇ ЗМІННОЇ Y У ТОЧЦІ T = TOUT C  
 C І ЗАПАМ'ЯТОВУЄ ЇХ ПОТІМ У МАСИВІ YO : C  
 C NQ C  
 C YO(I) = SUM Y(I, J+1)\*S\*\*J . C  
 C J=0 C  
 C DE S = -(T-TOUT)/H . C  
 CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

DIMENSION YO(NO), Y(NO, N)  
 COMMON/GEAR1/T, H, DUMMY(4), N, IDUMMY(2), JSTART  
 DO 10 I=1, N  
 YO(I)=Y(I, 1)  
 L=JSTART+1  
 S=(TOUT-T)/H  
 S1=1.  
 DO 30 J=2, L

10

```
S1=S1*S
DO 20 I=1,N
YO(I)=YO(I)+S1*Y(I,J)
```

20  
30

CONTINUE

RETURN

C----- КІНЕЦЬ П/П INTERP -----C  
END

SUBROUTINE PSET(Y,NO,CON,MITER,IER)

```
CC
C П/П PSET ОБЧИСЛЮЄ І ПЕРЕТВОРЮЄ МАТРИЦЮ ІТЕРАЦІЇ C
C $P = I - H * EL(1) * J$, ДЕ J - АПРОКСИМАЦІЯ ЯКОБІАНА, C
C I - ОДИНИЧНА МАТРИЦЯ C
C МАТРИЦЯ J ОБЧИСЛЮЄТЬСЯ ЧИ ЗА ДОПОМОГОЮ ПІДГОТОВ- C
C ЛЕНОЇ ПРОГРАМІСТОМ П/П PEDERV (ПРИ MITER = 1), C
C ЧИ ЧИСЕЛЬНИМ ДИФЕРЕНЦІЮВАННЯМ (ПРИ MITER = 2). C
C J ЗБЕРІГАЄТЬСЯ В МАСИВІ PW І ЗАМІНЮЄТЬСЯ НА C
C P, ПРИ ЦЬОМУ ВИКОРИСТОВУЄТЬСЯ ЗНАЧЕННЯ C
C $CON = -H * EL(1)$. НАД МАТРИЦЕЮ ІТЕРАЦІЇ P ПРОВО- C
C ДИТЬСЯ ПОТІМ ОПЕРАЦІЯ LU-ДЕКОМПОЗИЦІЇ. РЕЗУЛЬТАТ C
C ЗБЕРІГАЄТЬСЯ В МАСИВІ PW І ЗАСТОСОВУЄТЬСЯ ДЛЯ C
C НАСТУПНОГО РОЗВ'ЯЗУВАННЯ СЛАР. C
C В П/П PSET С ТАКОЖ ТАКІ ПАРАМЕТРИ: C
C $EPSJ = \sqrt{UROUND}$ - ВИКОРИСТОВУЄТЬСЯ ДЛЯ C
C ОЦІНКИ ПРИРОСТІВ ПРИ ЧИСЕЛЬНОМУ ДИФЕРЕНЦІЮВАННІ; C
C $NSQ = NQ ** 2$. C
CC
```

```
DIMENSION Y(NO,N)
COMMON/GEAR1/T,H,DUMMY(3),UROUND,N,IDUMMY(3)
COMMON/GEAR2/YMAX(1)
COMMON/GEAR4/SAVE1(1)
COMMON/GEAR5/SAVE2(1)
COMMON/GEAR6/PW(1)
COMMON/GEAR7/IPIV(1)
COMMON/GEAR8/EPSJ,NSQ
IF(MITER.EQ.2) GO TO 20
```

C ДЛЯ MITER = 1 ОБЧИСЛЕНЕ ЗА ДОПОМОГОЮ П/П PEDERV  
C ЗНАЧЕННЯ PW МНОЖИТЬСЯ НА СКАЛЯР.

CALL PEDERV(N,T,Y,PW,NO)

DO 10 I=1,NSQ

10 PW(I)=PW(I)\*CON

GO TO 60

C ДЛЯ MITER = 2 ВИКЛИКАЄТЬСЯ N РАЗІВ П/П DIFFUN  
C З МЕТОЮ АПРОКСИМАЦІЇ J.

20 D=0.

DO 30 I=1,N

30 D=D+SAVE2()\*\*2

RO=ABS(H)\*SQRT(D)\*1.E03\*UROUND

J1=0

DO 50 J=1,N

YJ=Y(J,1)

R=EPSJ\*YMAX(J)

```

R=AMAX1(R,RO)
Y(J,1)=Y(J,1)+R
D=CON/R
CALL DIFFUN(N,T,Y,SAVE1)
DO 40 I=1,N
 IJ1=I+J1
 PW(IJ1)=(SAVE1(I)-SAVE2(I))*D
40 Y(J,1)=YJ
 J1=J1+NO
50 CONTINUE
C ПІДСУМОВУВАННЯ З ОДИНИЧНОЮ МАТРИЦЕЮ І.
60 J=1
 DO 70 I=1,N
 PW(J)=PW(J)+1.
70 J=J+(NO+1)
C ПРОВОДИТЬСЯ LU-ДЕКОМПОЗИЦІЯ ІТЕРАЦІЙНОЇ МАТРИЦІ Р.
CALL DEC(N,NO,PW,IPIV,IER)
RETURN
C----- КІНЕЦЬ П/П PSET -----C
END

```

```

SUBROUTINE SHTNG (CASE,F,N,K,KTORT,EPS,AEPS,
* TVIV,IPVT,TORT,A1,OM,Z,Z1,Z2)
CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ ЛІНІЙНОЇ КРАЙОВОЇ ЗАДАЧІС
C ДЛЯ СИСТЕМИ ЗДР ІЗ ЗАСТОСУВАННЯМ РІЗНИХ ВАРІАНТІВ
C МЕТОДУ СТРІЛЬБИ
C ВХІДНІ ПАРАМЕТРИ :
C -----C
C CASE - РЕЖИМ РОБОТИ ПІДПРОГРАМИ :
C = 0 - ЗВИЧАЙНИЙ МЕТОД СТРІЛЬБИ; ПРИ ЗВО-
C РОТНОМУ ХОДІ ПОВТОРНО РОЗВ'ЯЗУЮТЬСЯ ЗАДАЧІС
C КОШІ З ПОЧАТКОВОЮ УМОВОЮ (11) І ВИВЕДЕН-
C ННЯМ РОЗВ'ЯЗКУ В ТОЧКАХ TORT(I);
C = 1 - ЗВИЧАЙНИЙ МЕТОД СТРІЛЬБИ З ВІДНОВ-
C ЛЕННЯМ РОЗВ'ЯЗКУ ПРИ ЗВОРОТНОМУ ХОДІ ЗА
C ДОПОМОГОЮ ФОРМУЛИ (8) У ТОЧКАХ TORT(I)
C І ВИВЕДЕННЯМ ЦЬОГО РОЗВ'ЯЗКУ ЗА УМОВИ
C TVIV(I)=.FALSE.;
C = 2 - МЕТОД СТРІЛЬБИ З ОРТОГОНАЛІЗАЦІЄЮ
C В ТОЧКАХ TORT(I) І ВІДНОВЛЕННЯМ РОЗВ'ЯЗКУ
C ПРИ ЗВОРОТНОМУ ХОДІ ЗА ДОПОМОГОЮ ФОРМУЛИ
C (8) У ТОЧКАХ TORT(I) І ВИВЕДЕННЯМ ЦЬОГО
C РОЗВ'ЯЗКУ ЗА УМОВИ TVIV(I)=.FALSE.;
C F - ІМ'Я ПІДПРОГРАМИ ОБЧИСЛЕННЯ ПОХІДНИХ ПРИ
C ІНТЕГРУВАННІ ЗА МЕТОДОМ РУНГЕ-КУТТИ;
C K - КІЛЬКІСТЬ УМОВ, ЗАДАНИХ У ПОЧАТКОВІЙ
C ТОЧЦІ, K>=N-K;
C KTORT- КІЛЬКІСТЬ ТОЧОК ОРТОГОНАЛІЗАЦІЇ ПРИ CASE=2С
C І КІЛЬКІСТЬ ТОЧОК ВИВЕДЕННЯ В РЕШТІ ВИ-
C ПАДКІВ;
C EPS - ВІДНОСНА ПОХИБКА ІНТЕГРУВАННЯ МЕТОДОМ
C РУНГЕ-КУТТИ;
C

```





```

C TYPE 402, (A1(I), I=1, NGR)
C РОЗВ'ЯЗУВАННЯ СЛАБ: ФОРМУВАННЯ МАТРИЦІ
 IF (K.GT.1)
 * CALL DECOMP(K,K, A1, COND, IPVT, Z2)
 TYPE 403, COND
 IF (COND.NE.O.) GO TO 30
 TYPE 404
 RETURN
30 I1=K+1
 II=0
 IJ=K*K
 JJ=N+1
 DO 60 J=1, M
 DO 40 I=1, K
 II=II+1
 IJ=IJ+1
40 Z(II)=-A1(IJ)
 DO 50 I=1, M
 II=II+1
50 Z(II)=0.
 Z(I1)=1.
60 II=I1+JJ
 DO 70 I=1, K
 II=II+1
 IJ=IJ+1
70 Z(II)=A1(IJ)
 DO 80 I=1, M
 II=II+1
80 Z(II)=0.
 I1=1
 DO 90 J=1, MM
 CALL SOLVE (K,K, A1, Z(I1), IPVT)
90 I1=I1+N
 TYPE 405, (Z(I), I=1, N1)
 IF1=1
 IF2=1
C ОРТОГОНАЛІЗАЦІЯ І ЗАПИС МАТРИЦІ У ФАЙЛ ДЛЯ X=A
 IF (CASE.EQ.2) CALL ORT (Z, N, MM, A1)
 WRITE(3' IF1) (Z(II), II=1, N1)
C ПРЯМИЙ ХІД МЕТОДУ СТРІЛЬБИ
 I1=1
 IZ=0
100 INDF=0
 II=1
110 JJ=(II-1)*N+1
 IFLAG=1
 X=TORT(I1)
 IF (CASE.GT.0) GO TO 120
 IJ=KTORT
 GO TO 130
120 IJ=I1+1
130 X1=TORT(IJ)

```

```

CALL RKF45(F, N, Z(JJ), X, X1, EPS, AEPS,
* IFLAG, Z1, IPVT)
IF (IFLAG.EQ.2) GO TO 140
TYPE 406, IFLAG
RETURN
140 IF (INDF.EQ.1) GO TO 150
 II=II+1
 IF (II.EQ.M+1) INDF=1
 GO TO 110
150 IF (CASE.NE.2) GO TO 190
 CALL ORT(Z, N, MM, A1)
 IF (IZ.EQ.0) GO TO 170
 CALL PRTM (MM, A1, OM, Z1)
 DO 160 J=1, KK
 OM(J)=Z1(J)
160 GO TO 190
170 IZ=1
 DO 180 J=1, KK
 OM(J)=A1(J)
180 IF (IJ.EQ.KTORT) GO TO 220
190 IF (TVIV(IJ)) GO TO 210
 IF (CASE.EQ.1) GO TO 200
 IZ=0
 WRITE (4' IF2) (OM(I), I=1, KK)
200 WRITE (3' IF1) (Z(I), I=1, N1)
210 I1=I1+1
 GO TO 100

С ЗАДАВАННЯ ГРАНИЧНИХ УМОВ ПРИ X=B, ВИЗНАЧЕННЯ СТАЛИХ
С У ЛІНІЙНІЙ КОМБІНАЦІЇ
220 NGR=(N+1)*M
 READ (9,402) (A1(I), I=1, NGR)
 TYPE 402, (A1(I), I=1, NGR)
 CALL PRPM (M, N, MM, A1, Z, Z1)
 IF (M.GT.1)
* CALL DECOMP(M, M, Z1, COND, IPVT, Z2)
 TYPE 403, COND
 IF (COND.NE.0.) GO TO 230
 TYPE 407
 RETURN
230 II=M*M+1
 IJ=N*M+1
 DO 240 J=1, M
 A1(J)=A1(IJ)-Z1(II)
 II=II+1
240 IJ=IJ+1
 CALL SOLVE (M, M, Z1, A1, IPVT)

С ЗВОРОТНИЙ ХІД МЕТОДУ СТІЛЬБИ
 IF (CASE.GT.0) GO TO 270
 IFLAG=1
 IF1=1
 INDF=1
 I1=1

```

```

A1(MM)=1.
READ (3' IF1) (Z1(II), II=1, N1)
CALL PRPM (N, MM, 1, Z1, A1, Z)
X=TORT(I1)
GO TO 260
250 X1=TORT(I1)
CALL RKF45 (F, N, Z, X, X1, EPS, AEPS,
* IFLAG, Z1, IPVT)
IF (IFLAG.EQ. 2) GO TO 260
TYPE 406, IFLAG
RETURN
260 WRITE (2, 408) X, (Z(II), II=1, N)
I1=I1+1
IF (I1.LE.KTORT) GO TO 250
RETURN
270 I=KTORT
IF1=IF1-1
IF2=IF2-1
C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ТА ЙОГО ВИВЕДЕННЯ
280 A1(MM)=1.
CALL PRPM (N, MM, 1, Z, A1, Z1)
WRITE (2, 408) TORT(I), (Z1(II), II=1, N)
IF (I.EQ. 1) RETURN
C ВВЕДЕННЯ ДАНИХ З ФАЙЛІВ І ВИЗНАЧЕННЯ СТАЛИХ У ТОЧЦІ
C ВИВЕДЕННЯ
IF (I.EQ.KTORT .OR. CASE.EQ. 1) GO TO 290
READ(4' IF2) (OM(II), II=1, KK)
IF2=IF2-2
290 READ(3' IF1) (Z(II), II=1, N1)
IF1=IF1-2
300 I=I-1
IF(TVIV(I)) GO TO 300
IF(CASE.EQ. 2) CALL RSTM (MM, OM, A1)
GO TO 280
400 FORMAT(' *LKZODY* : РЕЖИМ РОБОТИ ', I3/
* 6X, ' НЕ ВІДПОВІДАЄ ОПИСУ ПІДПРОГРАМИ')
401 FORMAT(' *LKZODY* : КІЛЬКІСТЬ УМОВ У ТОЧЦІ A',
* I2/I2X, ' В ТОЧЦІ B ', I2, '?')
402 FORMAT(///(5E12.5))
403 FORMAT(' ОЦІНКА ЧИСЛА ОБУМОВЛЕНОСТІ',
* ' МАТРИЦІ : ', ' COND=', E12.5)
404 FORMAT(' *LKZODY* : ВИРОДЖЕНІСТЬ МАТРИЦІ',
* ' ПРИ X=A')
405 FORMAT(' БАЗИС ЛІНІЙНОГО МНОГОВИДУ В ПОЧ.',
* ' ТОЧЦІ : '/(4E13.5))
406 FORMAT(' *LKZODY* : ПЕРЕРИВАННЯ ПРОЦЕСУ ПРИ '//
* ІНТЕГРУВАННІ ЗАДАЧІ КОШІ'//
* ' *RKF45* : IFLAG=', I3)
407 FORMAT(' *LKZODY* : ВИРОДЖЕНІСТЬ МАТРИЦІ',
* ' ПРИ X=B ')
408 FORMAT(' ВЕКТОР РОЗВ'ЯЗКУ В ТОЧЦІ X=',
* E12.5/(4E13.5))
END

```

SUBROUTINE PRPM ( N, M, L, A, B, C )

```

CC
C ПІДПРОГРАМА ПЕРЕМНОЖЕННЯ ПРЯМОКУТНИХ МАТРИЦЬ C=A*BC
C ВХІДНІ ПАРАМЕТРИ :
C -----
C N - КІЛЬКІСТЬ РЯДКІВ МАТРИЦЬ A ТА C;
C M - КІЛЬКІСТЬ СТОВПЦІВ МАТРИЦІ A ТА РЯДКІВ МАТ-
C РИЦІ B;
C L - КІЛЬКІСТЬ СТОВПЦІВ МАТРИЦЬ B ТА C;
C A - МАСИВ РОЗМІРНІСТЮ (N, M) ЕЛЕМЕНТІВ МАТРИЦІ A;
C B - МАСИВ РОЗМІРНІСТЮ (M, L) ЕЛЕМЕНТІВ МАТРИЦІ B.
C ВИХІДНІ ПАРАМЕТРИ :
C -----
C C - МАСИВ РОЗМІРНІСТЮ (N, L) МАТРИЦІ РЕЗУЛЬТАТУ.
CC
 INTEGER N, M, L, I, J, K
 REAL*4 A(N, M), B(M, L), C(N, L)
 REAL*8 S
 DO 30 I=1, N
 DO 20 J=1, L
 S=0.
 DO 10 K=1, M
 S=S+A(I, K)*B(K, J)
 C(I, J)=S
 10
 20
 30
 CONTINUE
 RETURN
 END

```

SUBROUTINE PRM ( N, A, B, C )

```

CC
C ПІДПРОГРАМА ПЕРЕМНОЖЕННЯ ТРИКУТНИХ МАТРИЦЬ C=A*B
C ВХІДНІ ПАРАМЕТРИ :
C -----
C N - РОЗМІРНІСТЬ МАТРИЦЬ A, B, C;
C A - МАСИВ N*(N+1)/2 НЕНУЛЬОВИХ ЕЛЕМЕНТІВ МАТРИЦІ
C A, ЗАДАНИЙ ПО СТОВПЦЯХ;
C B - МАСИВ N*(N+1)/2 НЕНУЛЬОВИХ ЕЛЕМЕНТІВ МАТРИЦІ
C B, ЗАДАНИЙ ПО СТОВПЦЯХ.
C ВИХІДНІ ПАРАМЕТРИ :
C -----
C C - МАСИВ N*(N+1)/2 НЕНУЛЬОВИХ ЕЛЕМЕНТІВ МА-
C ТРИЦІ РЕЗУЛЬТАТУ.
CC
 INTEGER N, IBS, IAN, IANS, I, IB, J, IA, IAS, K
 REAL*4 A(1), B(1), C(1)
 REAL*8 S
 IBS=0
 IAN=1
 IANS=1
 DO 40 I=1, N
 IB=IAN

```

```

DO 30 J=1, N
 S=0.
 IA=IAN
 IAS=IBS
 K=I
 S=S+A(IA)*B(IB)
10 IB=IB+1
 IF (K.EQ.J) GO TO 20
 IAS=IAS+1
 IA=IA+IAS
 K=K+1
 GO TO 10
20 C(IA)=S
30 IB=IB+IBS
 IBS=IBS+1
 IANS=IANS+1
40 IAN=IAN+IANS

RETURN
END

```

SUBROUTINE RSTM (N, A, B)

```

CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ СЛАР А*Х=В, ДЕ А - C
C ВЕРХНЯ ТРИКУТНА МАТРИЦЯ, ЗАДАНА ПО СТОВПЦІАХ C
C ВХІДНІ ПАРАМЕТРИ : C
C ----- C
C N - ПОРЯДОК СИСТЕМИ; C
C A - МАСИВ N*(N+1)/2 НЕНУЛЬОВИХ ЕЛЕМЕНТІВ МАТ- C
C РИЦІ СЛАР; C
C B - МАСИВ РОЗМІРНІСТЮ N ЕЛЕМЕНТІВ ПРАВОЇ ЧАС- C
C ТИНИ СЛАР. C
C ВИХІДНІ ПАРАМЕТРИ : C
C ----- C
C B - МАСИВ РОЗМІРНІСТЮ N ЕЛЕМЕНТІВ ВЕКТОРА РОЗ- C
C В'ЯЗКУ. C
CC
INTEGER N, LL, II, L, K, J, LS
REAL*4 A(1), B(1)
REAL*8 S
IF (N.GE.1) GO TO 10
TYPE *, ' *RSTMDO* : N=', N
STOP
10 K=N+1
 LL=N*K/2
 DO 40 II=1, N
 L=LL
 J=N
 LS=N
 S=0.
20 IF(K.GT.J) GO TO 30
 S=S+B(J)*A(L)
 J=J-1

```



```

LI=LI+1
IF(I.EQ.J) GO TO 40
W(LI)=X1
P=P+X1*X1
J=J+1
GO TO 20
40 W(LI)=DSQRT(X1-P)
C ОРТОГОНАЛІЗАЦІЯ I - ГО ВЕКТОРА :
IJ=0
J=0
50 J=J+1
IF(J.EQ.I) GO TO 70
IK=IK+1
X1=W(IK)
II=LN
DO 60 K1=1,N
II=II+1
IJ=IJ+1
60 Z(II)=Z(II)-Z(IJ)*X1
GO TO 50
70 IF(I.EQ.MM) GO TO 90
X1=1./W(LI)
DO 80 K1=1,N
LN=LN+1
80 Z(LN)=Z(LN)*X1
I=I+1
GO TO 10
90 W(LI)=1.
RETURN
END

```

SUBROUTINE GALOR(A,B,ALF,BET,GAM,DEL,N,  
\* EPS,QA,P,QU,FO,AM,FR)

```

CC
C ПІДПРОГРАМА ФОРМУЄ СИСТЕМУ ЛІНІЙНИХ АЛГЕБРАІЧНИХ C
C РІВНЯНЬ, ЯКА ВИНИКАЄ ПРИ РОЗВ'ЯЗУВАННІ C
C ОСНОВНИХ КРАЙОВИХ ЗАДАЧ ДЛЯ ЗДР ДРУГОГО ПОРЯДКУ C
C МЕТОДОМ БУБНОВА-ГАЛЬБОРКІНА C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C A - ЛІВИЙ КІНЕЦЬ ВІДРІЗКА [A,B], НА ЯКОМУ C
C ШУКАЄТЬСЯ РОЗВ'ЯЗОК; C
C B - ПРАВИЙ КІНЕЦЬ ВІДРІЗКА [A,B], НА ЯКОМУ C
C ШУКАЄТЬСЯ РОЗВ'ЯЗОК; C
C ALF,BET,GAM,DEL - ПАРАМЕТРИ КРАЙОВИХ УМОВ; C
C N - ЧИСЛО КООРДИНАТНИХ ФУНКЦІЙ; C
C EPS - ТОЧНІСТЬ ОБЧИСЛЕННЯ ІНТЕГРАЛІВ ЗА ФОРМУЛОЮ C
C СІМПСОНА; C
C QA,P,QU,FO - ІМЕНА ПІДПРОГРАМ-ФУНКЦІЙ, ЯКІ C
C ОПИСУЮТЬ КОЕФІЦІЄНТИ C
C РІВНЯННЯ K(X),P(X),Q(X),F(X). C
C ВИХІДНІ ПАРАМЕТРИ: C
C ----- C

```

```

C AM - МАТРИЦЯ КОЕФІЦІЄНТІВ РОЗМІРНІСТЮ N*N СЛАР
C МЕТОДУ БУБНОВА-ГАЛЬОРКІНА;
C FR - ВЕКТОР ПРАВОЇ ЧАСТИНИ РОЗМІРНІСТЮ N СЛАР
C МЕТОДУ БУБНОВА-ГАЛЬОРКІНА.
C ПАРАМЕТРИ, ЯКІ ВЗАЄМОДІЮТЬ ЧЕРЕЗ ПОІМЕНОВАНИЙ
C COMMON - БЛОК З ІМ'ЯМ В1 :
C ICOND - ОЗНАКА ПРАВИЛЬНОСТІ ПАРАМЕТРІВ У КРАЙОВИХ
C УМОВАХ: ЯКЩО ICOND=0 - ПАРАМЕТРИ ЗАДАНО
C ПРАВИЛЬНО; ЯКЩО ICOND=1 - НЕПРАВИЛЬНО;
C NF - НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ РЕЗУЛЬТАТІВ.
C ПІДПРОГРАМИ, ЩО ВИКОРИСТОВУЮТЬСЯ:

C SIMP - ОБЧИСЛЮЄ ІНТЕГРАЛИ ЗА ФОРМУЛОЮ СІМПСОНА;
C AIJ - ОБЧИСЛЮЄ ЗНАЧЕННЯ ПІДІНТЕГРАЛЬНОЇ
C ФУНКЦІЇ В КОЕФІЦІЄНТАХ МАТРИЦІ АМ;
C FII - ОБЧИСЛЮЄ ЗНАЧЕННЯ ПІДІНТЕГРАЛЬНОЇ ФУНКЦІЇ
C У ВЕКТОРІ ПРАВОЇ ЧАСТИНИ FR.
CC
EXTERNAL AIJ, FII
COMMON /B1/ ICOND, NF
COMMON /B2/ I, J
REAL AM(N, N), FR(N)
C IER- ОЗНАКА ЗАВЕРШЕННЯ ПІДПРОГРАМИ SIMP
IER = 0
ICOND = 0
ABGD=ALF*BET*GAM*DEL
IF (ALF. GE. 0. AND. BET. GE. 0. AND. GAM. GE. 0. AND.
* DEL. GE. 0) GO TO 30
10 TYPE *, ' НЕПРАВИЛЬНО ЗАДАНО ПАРАМЕТРИ В '
* ' КРАЙОВИХ УМОВАХ'
20 ICOND = 1
RETURN
30 IF ((ALF+BET). GT. 0. . AND. (DEL+GAM). GT. 0.)
* GO TO 40
GO TO 10
40 IF (ALF. EQ. 0. . AND. ALF. EQ. GAM) GO TO 50
IF (BET. EQ. 0. . AND. BET. EQ. DEL) GO TO 60
IF (ABGD. NE. 0.) GO TO 70
TYPE *, ' ЦЕЙ ТИП КРАЙОВИХ УМОВ У ПРОГРАМІ '
* ' НЕ РЕАЛІЗОВАНИЙ'
GO TO 20
50 RITE(NF, 200)
GO TO 80
60 WRITE(NF, 201)
GO TO 80
70 WRITE(NF, 202)
C ОБЧИСЛЕННЯ МАТРИЦІ АМ СИСТЕМИ МЕТОДУ
C БУБНОВА-ГАЛЬОРКІНА
80 DO 90 J=1, N
DO 90 I=1, N
C ПІДПРОГРАМА SIMP ОБЧИСЛЮЄ ІНТЕГРАЛИ ЗА ФОРМУЛОЮ
C СІМПСОНА

```



```

CALL SIMP(A, B, 3, EPS, 1. E-06, RHMIN, IERR
*
 AIJ, S)
IF (IERR. NE. 0) IER=IER+1
AM(I, J)=S
IF (ABGD. EQ. 0) GO TO 90
AM(I, J)=S+DEL/GAM*QA(B)*FI(B, J)*FI(B, I)
*
 +BET/ALF*QA(A)*FI(A, J)*FI(A, I)
90 CONTINUE
С ОБЧИСЛЕННЯ ВЕКТОРА ПРАВОЇ ЧАСТИНИ СЛАР МЕТОДУ
С БУБНОВА-ГАЛЬБОРКИНА - FR
DO 100 I=1, N
 CALL SIMP(A, B, 3, EPS, 1. E-06, RHMIN, IERR,
*
 FII, S)
IF (IERR. NE. 0) IER=IER+1
FR(I)=S
100 CONTINUE
IF (IER. NE. 0) WRITE(NF, 203) IER
IF (IER. EQ. 0) WRITE(NF, 204)
RETURN
200 FORMAT(3X, '*GALOR* КООРДИНАТНА СИСТЕМА, '
*
 'ЯКА ВІДПОВІДАЄ КРАЙОВИМ' /, 11X,
*
 'УМОВАМ ПЕРШОГО РОДУ ')
201 FORMAT(3X, '*GALOR* КООРДИНАТНА СИСТЕМА, '
*
 'ЯКА ВІДПОВІДАЄ КРАЙОВИМ' /, 11X,
*
 'УМОВАМ ДРУГОГО РОДУ ')
202 FORMAT(3X, '*GALOR* КООРДИНАТНА СИСТЕМА, '
*
 'ЯКА ВІДПОВІДАЄ КРАЙОВИМ' /, 11X,
*
 'УМОВАМ ТРЕТЬОГО РОДУ ')
203 FORMAT(3X, '! *SIMP* ПРИ ОБЧИСЛЕННІ , I4,
*
 ' ІНТЕГРАЛІВ НЕ ДОСЯГНУТО ЗАДАНОЇ '
*
 'ТОЧНОСТІ ')
204 FORMAT(3X, '*SIMP* УСІ ІНТЕГРАЛИ '
*
 'ОБЧИСЛЕНО З ЗАДАНОЮ ТОЧНІСТЮ ')
END

```

FUNCTION FI(X, I)

```

С ПІДПРОГРАМА-ФУНКЦІЯ, ЯКА ОБЧИСЛЮЄ ПОЛІНОМНУ
С КООРДИНАТНУ СИСТЕМУ
С ВХІДНІ ПАРАМЕТРИ:
С
С -----
С X - ЗНАЧЕННЯ АРГУМЕНТУ ФУНКЦІЇ;
С I - НОМЕР ФУНКЦІЇ.
С ПАРАМЕТРИ, ЯКІ ВЗАЄМОДІЮТЬ ЧЕРЕЗ ПОІМЕНОВАНИЙ
С COMMON - БЛОК З ІМ'ЯМ ВВ :
С A - ЛІВИЙ КІНЕЦЬ ВІДРІЗКА [A, B];
С B - ПРАВИЙ КІНЕЦЬ ВІДРІЗКА [A, B];
С ALFA, BETA, GAMMA, DELTA - ПАРАМЕТРИ КРАЙОВИХ УМОВ.
С
С
С
COMMON /BV/ A, B, ALF, BET, GAM, DEL
IF (I. EQ. 1) FI=(X-A)**2*(X-B-GAM*(B-A)/
*
 (2.*GAM+DEL*(B-A)))

```

```

~ / IF (I.EQ.2) FI=(X-X)**2*(X-A+ALF*(B-A)/
* (2.*ALF+BET*(B-A)))
IF (I.GE.3) FI=(X-A)**(I-1)*(B-X)**2
RETURN
END

```

```

FUNCTION FIP(X, I)
CC
C ПІДПРОГРАМА-ФУНКЦІЯ, ЯКА ОБЧИСЛЮЄ ПОХІДНІ ВІД C
C ПОЛІНОМНОЇ КООРДИНАТНОЇ СИСТЕМИ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C X - ЗНАЧЕННЯ АРГУМЕНТУ ФУНКЦІЇ; C
C I - НОМЕР ФУНКЦІЇ. C
C ПРИМІТКА: ПІДПРОГРАМА ВИКОРИСТОВУЄ ПОІМЕНОВАНИЙ C
C ----- COMMON - БЛОК ВВ, ЯКИЙ ОПИСАНИЙ C
C В ПІДПРОГРАМІ FI. C
CC
COMMON /BV/ A, B, ALF, BET, GAM, DEL
IF (I.EQ.1) FIP=(X-A)*((X-A)+2*(X-B-GAM*(B-A)/
* (2.*GAM+DEL*(B-A))))
IF (I.EQ.2) FIP=(B-X)*((B-X)-2*(X-A+ALF*(B-A)/
* (2.*ALF+BET*(B-A))))
IF (I.GE.3) FIP=(X-A)**(I-2)*(B-X)*((I-1)*
* (B-X)-2*(X-A))
RETURN
END

```

```

FUNCTION AIJ(X)
CC
C ПІДПРОГРАМА-ФУНКЦІЯ, ЯКА ОБЧИСЛЮЄ ЗНАЧЕННЯ C
C ПІДІНТЕГРАЛЬНОЇ ФУНКЦІЇ В КОЕФІЦІЕНТАХ МАТРИЦІ АМ C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C X - ЗНАЧЕННЯ АРГУМЕНТУ ФУНКЦІЇ. C
CC
COMMON /B2/ I, J
S=0.
IF (QU(X).NE.0.) S=S+QU(X)*FI(X, J)
IF (P(X).NE.0.) S=S+P(X)*FIP(X, J)
IF (S.NE.0.) S=S*FI(X, I)
AIJ=QA(X)*FIP(X, I)*FIP(X, J)+S
RETURN
END

```

```

FUNCTION FII(X)
CC
C ПІДПРОГРАМА-ФУНКЦІЯ, ЯКА ОБЧИСЛЮЄ ЗНАЧЕННЯ C
C ПІДІНТЕГРАЛЬНОЇ ФУНКЦІЇ У ВЕКТОРІ ПРАВОЇ ЧАСТИНИ C
C FR C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C X - ЗНАЧЕННЯ АРГУМЕНТУ ФУНКЦІЇ. C
CC

```

```

COMMON /B2/ I
FII=FO(X)*FI(X, I)
RETURN
END

```

SUBROUTINE SISVD(A, B, M, N, X, SIGMA, U, V, R)

```

CC
C ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ СЛАР У ВИПАДКУ
C ПЕРЕВИЗНАЧЕНОЇ СИСТЕМИ
C ВХІДНІ ПАРАМЕТРИ:

```

```

C A - ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ (M, N), ЯКИЙ
C МІСТИТЬ ПРЯМОКУТНУ МАТРИЦЮ СЛАР;
C B - ВЕКТОР РОЗМІРНІСТЮ M, ЯКИЙ МІСТИТЬ ВЕКТОР
C ПРАВИХ ЧАСТИН СЛАР;
C M - КІЛЬКІСТЬ РЯДКІВ МАТРИЦІ A;
C N - КІЛЬКІСТЬ СТОВПЦІВ МАТРИЦІ A.

```

ВИХІДНІ ПАРАМЕТРИ:

```

C X - ВЕКТОР РОЗМІРНІСТЮ N, ЯКИЙ МІСТИТЬ РОЗВ'ЯЗОК
C СЛАР;
C SIGMA - ВЕКТОР РОЗМІРНІСТЮ N, ЯКИЙ МІСТИТЬ
C СИНГУЛЯРНІ ЧИСЛА МАТРИЦІ A;
C U - ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ (M, N), ЯКИЙ
C МІСТИТЬ ЛІВУ МАТРИЦЮ ОБЕРТАННЯ U;
C U - ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ (M, N), ЯКИЙ
C МІСТИТЬ ПРАВУ МАТРИЦЮ ОБЕРТАННЯ V;
C R - МІСТИТЬ НЕВ'ЯЗКУ !!A*X-B!!

```

```

CC

```

```

REAL A(M, N), B(M), X(N), SIGMA(N), U(M, N), V(M, N),

```

```

* WORK(30), D(30), Z(30), R1(31)

```

```

LOGICAL L1, L2

```

```

NM=M

```

```

L1=.TRUE.

```

```

L2=.TRUE.

```

```

CALL SVD(NM, M, N, A, SIGMA, L1, U, L2, V, IERR, WORK)

```

```

C ЗНАХОДЖЕННЯ ПЕРШИХ N КОМПОНЕНТ ВЕКТОРА ПРАВИХ

```

```

C ЧАСТИН СЛАР SZ=D

```

```

DO 10 J=1, N

```

```

D(J)=0.

```

```

DO 10 I=1, M

```

```

D(J)=D(J)+U(I, J)*V(I)

```

```

CONTINUE

```

10

```

C ЗНАХОДЖЕННЯ НАИБІЛЬШОГО СИНГУЛЯРНОГО ЧИСЛА

```

```

SIGMA1=0.

```

```

DO 20 J=1, N

```

```

IF(SIGMA(J).GT.SIGMA1) SIGMA1=SIGMA(J)

```

```

X(J)=0.

```

```

CONTINUE

```

20

```

TAU=1.E-04

```

```

C РОЗВ'ЯЗОК СИСТЕМИ SZ=D, S-ДІАГОН.МАТР., ЕЛЕМЕНТИ

```



```

INTEGER G
REAL M1, M2, MM2, M11
DIMENSION Y(1), AL(401), BE(401), X(1), H(1)
COMMON /GR/ AL1, S1, M1, AL2, S2, M2, /GG/G
EPS=0.00001
С ОБЧИСЛЕННЯ КОЕФІЦІЄНТІВ РІЗНИЦЕВОЇ СХЕМИ
С ТА КОЕФІЦІЄНТІВ ПРОГОНКИ AL(I), BE(I)
 XX=X(1)+0.5*H(2)
 IF ((G.NE.0).AND.(X(1).EQ.0.)) GO TO 30
 IF (G.EQ.0) GO TO 10
 AAK=AK(XX)*XX**G
 X11=X(1)**G
 GO TO 20
10 AAK=AK(XX)
 X11=1.
20 CONTINUE
 S11=S1+AL1*H(2)/2.*Q(X(1))*X11
 S12=H(2)*S11+AL1*AAK
 AL(1)=AL1*AAK/S12
 BE(1)=H(2)*(M1+AL1*H(2)*F(X(1))*X11/2.)/S12
 GO TO 40
30 AKK=AK(XX)+H(2)**2/(2.*G)*Q(0.)
 AL(1)=AK(XX)/AKK
 BE(1)=H(2)**2*F(0.)/AKK/(2.*G)
40 CONTINUE
С КІНЕЦЬ ОБЧИСЛЕННЯ КОЕФІЦІЄНТІВ
 N1=N-1
 DO 90 I=1, N1
 X1=X(I)
 X2=X(I+1)
 H1=H(I+1)
 H2=H(I+2)
 IF (G.EQ.0) GO TO 50
 A=AK(X1+0.5*H1)*(X1+0.5*H1)**G/H1
 B=AK(X2+0.5*H2)*(X2+0.5*H2)**G/H2
 GO TO 60
50 A=AK(X1+0.5*H1)/H1
 B=AK(X2+0.5*H2)/H2
60 CONTINUE
 DP=(H1*Q(X2-EPS)+H2*Q(X2+EPS))/2.
 IF (G.E.0) GO TO 70
 C=A+B+DP*X2**G
 FF=X2**G*(H1*F(X2-EPS)+H2*F(X2+EPS))/2
 GO TO 80
70 C=A+B+DP
 FF=(H1*F(X2-EPS)+H2*F(X2+EPS))/2.
80 CONTINUE
 CC=C-AL(I)*A
 AL(I+1)=B/CC
90 BE(I+1)=(A*BE(I)+FF)/CC
 H2=H(N+1)
 X2=X(N+1)

```

```

X3=X2-0.5*H2
AKX2=AK(X3)
IF(G.EQ.0) GO TO 100
XG=X3**G
MM2=M2+AL2*H2/2.*F(X2)*X2**G
SS2=S2+AL2*H2/2.*Q(X2)*X2**G
GO TO 110
100 XG=1.
MM2=M2+AL2*H2/2.*F(X2)
SS2=S2+AL2*H2/2.*Q(X2)
110 CONTINUE
ZS2=H2*SS2+AL2*AKX2*XG
SSM2=AL2*AKX2*XG/ZS2
SM2=H2*MM2/ZS2
C ЗВОРОТНИЙ ХІД МЕТОДУ ПРОГОНКИ
C ОБЧИСЛЕННЯ МАСИВУ Y
Y(N+1)=(SM2+SSM2*BE(N))/(1.-AL(N)*SSM2)
DO 120 K1=1,N
J=N-K1+1
120 Y(J)=AL(J)*Y(J+1)+BE(J)
RETURN
END

```

SUBROUTINE GRID(N,K,X,H,RX,IR,ID)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C ПІДПРОГРАМА ПОБУДОВИ СІТКИ НА ВІДРІЗКУ C  
C ВХІДНІ ПАРАМЕТРИ: C

C ----- C  
C N - КІЛЬКІСТЬ ІНТЕРВАЛІВ СІТКИ; C  
C K - РОЗМІРНІСТЬ МАСИВУ RX; C  
C RX(I) I=1,K МІСТЯТЬ МНОЖИНУ ФІКСОВАНИХ ВУЗЛІВ C  
C СІТКИ, RX(1)<RX(2)<...<RX(K-1)<RX(K) C  
C RX(1),RX(K) МОЖУТЬ МІСТИТИ ЛІВИЙ І ПРАВИЙ C  
C КІНЦІ ВІДРІЗКА, C  
C RX(2),...,RX(K-1) - МОЖЛИВІ ТОЧКИ РОЗРИВУ C  
C КОЕФІЦІЄНТІВ ДИФЕРЕНЦІАЛЬНОГО РІВНЯННЯ; C  
C IR, ID - КЕРУЮЧІ ПАРАМЕТРИ: C  
C ПРИ IR=0 В ПІДПРОГРАМІ ВИКОРИСТОВУЮТЬСЯ ДВА C  
C ВХІДНИХ ПАРАМЕТРИ N ; C  
C X(N+1) - МАСИВ ВУЗЛІВ СІТКИ. C  
C ПІДПРОГРАМА ОБЧИСЛЮЄ МАСИВ КРОКІВ H(N); C  
C ПРИ IR=0 ПІДПРОГРАМА ЗА ВХІДНИМИ ПАРАМЕТРАМИ C  
C N,K,RX,IR,ID ОБЧИСЛЮЄ МАСИВИ ВУЗЛІВ X(N+1) C  
C І КРОКІВ H(N). C  
C ПРИ ПЕРШОМУ ЗВЕРНЕННІ ДО ПІДПРОГРАМИ GRID C  
C НЕОБХІДНО ПОЖАСТИ ID=1. C  
C ПРИ ДРУГОМУ ЗВЕРНЕННІ МОЖЛИВА ПОБУДОВА БІЛЬШ C  
C ЩІЛЬНОЇ СІТКИ З ДОДАТКОВОЮ МНОЖИНОЮ ВУЗЛІВ, ЯКІ C  
C ЛЕЖАТЬ ПОСЕРЕДИНІ МІЖ ВУЗЛАМИ X(I),X(I+1). C  
C ПРИ ЦЬОМУ ДОСИТЬ ВДРУГЕ ЗВЕРНУТИСЯ ДО C  
C ПІДПРОГРАМИ GRID ПРИ ID=2, ЗБЕРІГАЮЧИ C  
C НЕЗМІННИМИ ЗНАЧЕННЯ ІНШИХ ПАРАМЕТРІВ. C

ВИХІДНІ ПАРАМЕТРИ:

X(I) I=1,N+1 МІСТЯТЬ МНОЖИНУ ВУЗЛІВ СІТКИ;  
H(I) I=2,N+1 МІСТЯТЬ МНОЖИНУ КРОКІВ СІТКИ.

```

DIMENSION X(1),H(1),RX(1)
IF (ID.EQ.2) GO TO 70
X(1)=RX(1)
IF (IR.NE.0) GO TO 20
DO 10 I=1,N
 H(I+1)=X(I+1)-X(I)
GO TO 110
20 CONTINUE
IF (K.NE.2) GO TO 40
HH=(RX(K)-RX(1))/N
DO 30 I=1,N
 H(I+1)=HH
 X(I+1)=X(1)+I*HH
GO TO 110
40 CONTINUE
K1=K-1
SN=0.5
NN=0
IO=1

```

C ПІДПРОГРАМА ДІЛИТЬ ВІДРІЗОК [RX(I),RX(I+1)] НА NK  
C ІНТЕРВАЛІВ ТАК, ЩОБ ЗАБЕЗПЕЧИТИ МІНІМАЛЬНУ  
C НЕРІВНОМІРНІСТЬ СІТКИ

```

DO 60 I=1,K1
 AL=(RX(I+1)-RX(I))/(RX(K)-RX(I))
 SN=AL*(N-NN)+0.5
 NK=SN
 NN=NN+NK
 IF (NN.GT.N) NN=N
 HH=(RX(I+1)-RX(I))/NK
 NN1=NN-1
 DO 50 J=IO,NN1
 X(J+1)=X(J)+HH
 H(J+1)=HH
 X(NN+1)=RX(I+1)
 H(NN+1)=HH
 IO=NN+1
60 CONTINUE
GO TO 110
70 CONTINUE
N2=2*N
N1=N+1
DO 80 I=1,N1
 X(2*(N1+1-I)-1)=X(N1+1-I)
DO 90 I=1,N
 X(2*I)=(X(2*I-1)+X(2*I+1))/2
DO 100 I=1,N2
 H(I+1)=X(I+1)-X(I)

```

## SUBROUTINE MFE(F)

```

CC
С ПІДПРОГРАМА РОЗВ'ЯЗУВАННЯ МСЕ ЗАДАЧІ ДІРІХЛЕ ДЛЯ С
С РІВНЯННЯ ПУАССОНА В ОПУКЛОМУ МНОГОКУТНИКУ С
С ВХІДНІ ПАРАМЕТРИ : С
С ----- С
С F - ІМ'Я ПІДПРОГРАМИ-ФУНКЦІЇ, ЯКА ОБЧИСЛЮЄ ПРАВУ С
С ЧАСТИНУ РІВНЯННЯ ПУАССОНА. С
С ПАРАМЕТРИ, ЯКІ ВЗАЄМОДІЮТЬ ЧЕРЕЗ НЕПОІМЕНОВАНИЙ С
С COMMON-БЛОК : С
С INPUT - НОМЕР ПРИСТРОЮ ВВЕДЕННЯ; С
С KONSOL - НОМЕР ПРИСТРОЮ ВИВЕДЕННЯ; С
С IERR - ОЗНАКА ПОМИЛКИ: IERR=0, ЯКЩО ВСЕ С
С ПРАВИЛЬНО; С
С NVE - ЧИСЛО ВУЗЛОВИХ ЗМІННИХ ЕЛЕМЕНТА(NVE=3); С
С MAXNOD - МАКСИМАЛЬНИЙ НОМЕР ВУЗЛА; С
С MAXELM - МАКСИМАЛЬНО ДОПУСТИМЕ ЧИСЛО ЕЛЕМЕНТІВ; С
С NELMTS - ЧИСЛО ЕЛЕМЕНТІВ; С
С X,Y - МАСИВИ, ЯКІ МІСТЯТЬ КООРДИНАТИ ВУЗЛІВ; С
С NVTX - СПИСОК ВУЗЛІВ ДЛЯ КОЖНОГО ЕЛЕМЕНТА; С
С SOURCE - ЗНАЧЕННЯ ПРАВОЇ ЧАСТИНИ РІВНЯННЯ С
С ПУАССОНА В ЦЕНТРІ ВАГИ ТРИКУТНИКА; С
С CONSTR = .TRUE. ДЛЯ ФІКСОВАНИХ ЗНАЧЕНЬ РОЗВ'ЯЗКУ С
С (УМОВА ДІРІХЛЕ); С
С NODES - ЧИСЛО ВУЗЛІВ У ЗАДАЧІ; С
С POTENT - МАСИВ ЗНАЧЕНЬ РОЗВ'ЯЗКУ; С
С RTHDSD - ПРАВА ЧАСТИНА СЛАР; С
С SELM = S ДЛЯ ОДНОГО ЕЛЕМЕНТА; С
С TELM = T ДЛЯ ОДНОГО ЕЛЕМЕНТА; С
С S - ГЛОБАЛЬНА МАТРИЦЯ ЖОРСТКОСТІ; С
С INTG - РОБОЧИЙ МАСИВ. С
С ПІДПРОГРАМИ, ЩО ВИКОРИСТОВУЮТЬСЯ: С
С ----- С
С MESHINE - ЗДІЙСНУЄ ЧИТАННЯ ВХІДНОГО ПОТОКУ ДАНИХ; С
С SZERO - ОБНУЛЕННЯ ГЛОБАЛЬНОЇ МАТРИЦІ S І ПРАВОЇ С
С ЧАСТИНИ RTHDSD; С
С ELMATR - ОБЧИСЛЮЄ МАТРИЧНЕ ПОДАННЯ С
С ТРИКУТНОГО ЕЛЕМЕНТА; С
С ELEMVD - ВКЛЮЧАЄ ОДИНОЧНІ МАТРИЧНІ ЕЛЕМЕНТИ В С
С ГЛОБАЛЬНУ МАТРИЦЮ S І ВЕКТОР RTHDSD; С
С OUTPUT - ЗДІЙСНУЄ ВИВЕДЕННЯ РЕЗУЛЬТАТІВ; С
С FACTSQ, RESQ - РЕАЛІЗУЮТЬ МЕТОД КВАДРАТНОГО КОРЕНЯ. С
С ПРИМІТКА: С
С ----- С
С ПІДПРОГРАМА MFE С ПЕРЕРОБЛЕНИМ ВАРІАНТОМ С
С ПІДПРОГРАМИ, ЯКУ НАВЕДЕНО В [17, С.32]. С
CC
LOGICAL CONSTR
С РОЗМІРНОСТІ НАСТУПНИХ МАСИВІВ ДОРІВНЮЮТЬ MAXNOD

```



```

 DIMENSION X(50), Y(50), S(50, 50), D(50), WM(50),
 * CONSTR(50), RTHDSD(50), POTENT(50)
C РОЗМІРНОСТІ НАСТУПНИХ МАСИВІВ ДОРІВНЮЮТЬ
C MAXELM I/АБО NVE
 DIMENSION NVTX(3, 75), SOURCE(75), SELM(3, 3),
 * TELM(3, 3), INTG(3)
C РОЗМІРНІСТЬ МАСИВУ VS ДОРІВНЮЄ MAXNOD*(MAXNOD+1)/2
 DIMENSION VS(1275)
 COMMON INPUT, KONSOL, IERR, NVE, MAXNOD, MAXELM,
 * NELMTS, X, Y, NVTX, SOURCE, CONSTR, NODES,
 * POTENT, RTHDSD, SELM, TELM, S, INTG
C
 MAXNOD=50
 MAXELM=75
 NVE=3
C ВВЕДЕННЯ ДАНИХ
 CALL MESHIN(F)
 IF(IERR.NE.0) GO TO 100
C ОБНУЛЕННЯ ГЛОБАЛЬНОЇ МАТРИЦІ S
C І ПРАВОЇ ЧАСТИНИ RTHDSD
 CALL SZERO
C ФОРМУВАННЯ МАТРИЦІ ЕЛЕМЕНТІВ, ПОЕЛЕМЕНТНА ОБРОБКА
 DO 10 I=1, NELMTS
C ФОРМУВАННЯ МАТРИЦЬ S I T ЕЛЕМЕНТІВ
 IE=I
 CALL ELMATR(IE)
C ВСТАВКА МАТРИЦЬ В S; ДОДАВАННЯ ДО ПРАВОЇ ЧАСТИНИ
 CALL ELEMBD(IE)
10 CONTINUE
C ЗМЕНШЕННЯ РОЗМІРНОСТІ СИСТЕМИ S*POTENT=RTHDSD
C З УРАХУВАННЯМ УМОВ ДІРІХЛЕ
 NODES1=NODES
 DO 50 II=1, NODES
 I=II
 IF(.NOT.CONSTR(I)) GO TO 50
 NODES1=NODES1-1
 IF(I.EQ.NODES) GO TO 50
 I=I-NODES+NODES1+1
 DO 40 J=I, NODES1
 RTHDSD(J)=RTHDSD(J+1)
20 DO 20 K=1, NODES1
 S(J, K)=S(J+1, K)
30 DO 30 L=1, NODES1
 S(L, J)=S(L, J+1)
40 CONTINUE
 I=I+NODES-NODES1-1
50 CONTINUE
 NODES1=NODES-NODES1
 NODES=NODES-NODES1
 NODES1=NODES+NODES1
 K=0
 DO 60 J=1, NODES

```

```

DO 60 I=1, J
 K=K+1
60 VS(K)=S(I, J)
C РОЗВ'ЯЗУВАННЯ ЗНАЙДЕНОЇ СИСТЕМИ РІВНЯНЬ
 CALL FACTSQ(NODES, VS, D, COND, WM)
 CALL RESQ(NODES, VS, D, RTHDSD)
C ЗАПИС ЗНАЧЕНЬ РОЗВ'ЯЗКУ НА ВІДПОВІДНІ МІСЦЯ
DO 70 I=1, NODES1
70 WM(I)=0.
 K=1
DO 80 I=1, NODES1
 IF(CONSTR(I)) GO TO 80
 WM(I)=RTHDSD(K)
 K=K+1
80 CONTINUE
DO 90 I=1, NODES1
90 POTENT(I)=WM(I)
 NODES=NODES1
C ВИВЕДЕННЯ ЗНАЧЕНЬ РОЗВ'ЯЗКУ
 CALL OUTPUT
 STOP
100 TYPE *, ' ПРОЦЕС НЕ МОЖЕ БУТИ ПРОДОВЖЕНИЙ
 STOP
 END

```

SUBROUTINE MESHIN(RIGHT)

```

CC
C ВВЕДЕННЯ ПОТОКУ ДАНИХ, ЯКИЙ СКЛАДАЄТЬСЯ З ТРЬОХ
C ЧАСТИН: ВУЗЛІВ, ЕЛЕМЕНТІВ, ФІКСОВАНИХ ЗНАЧЕНЬ
C РОЗВ'ЯЗКУ, ОЗНАКОЮ КІНЦЯ КОЖНОЇ ЧАСТИНИ Є
C НУЛЬОВИЙ РЯДОК
C ВХІДНІ ПАРАМЕТРИ:
C -----
C RIGHT - ІМ'Я ПІДПРОГРАМИ-ФУНКЦІЇ, ЯКА ОБЧИСЛЮЄ
C ПРАВУ ЧАСТИНУ РІВНЯННЯ ПУАССОНА.
C ПРИМІТКА: ПІДПРОГРАМА ВИКОРИСТОВУЄ НЕПОІМЕНОВАНИЙ
C ----- COMMON - БЛОК, ЯКИЙ ОПИСАНИЙ В ПІДПРО-
C ГРАМІ MFE
C -----
CC

```

EXTERNAL RIGHT

LOGICAL CONSTR

DIMENSION X(50), Y(50), S(50, 50),

\* CONSTR(50), RTHDSD(50), POTENT(50),

\* NVTX(3, 75), SOURCE(75), SELM(3, 3),

\* TELM(3, 3), INTG(3)

COMMON INPUT, KONSOL, IERR, NVE, MAXNOD, MAXELM,

\* NELMTS, X, Y, NVTX, SOURCE, CONSTR, NODES,

\* POTENT, RTHDSD, SELM, TELM, S, INTG

C ЗЧИТУВАННЯ СПИСКУ ВУЗЛІВ

NODES=0

10 READ(INPUT, \*) I, XI, YI

C ЯКЩО НЕМАЄ ОЗНАКИ КІНЦЯ ДЛЯ ВУЗЛІВ - ВВЕДЕННЯ ВУЗЛА

```

20 IF(I.EQ.0) GO TO 40
 NODES=MAXO(NODES I)
 X(I)=XI
 Y(I)=YI
 GO TO 10
C ЯКИЙ НАЙБІЛЬШИЙ ДОПУСТИМИЙ НОМЕР ВУЗЛА?
30 IF(NODES.LE.MAXNOD) GO TO 40
 WRITE(KONSOL,200) NODES,MAXNOD
 GO TO 130
C ЗЧИТУВАННЯ СПИСКУ ЕЛЕМЕНТІВ ТА ЇХ ПІДРАХУНОК
40 NELMTS=0
50 READ(INPUT,*) INTG
C ПЕРЕВІРКА КІНЦЯ СПИСКУ
 IF(INTG(1).EQ.0) GO TO 80
C ОБЧИСЛЕННЯ ЗНАЧЕННЯ ФУНКЦІЇ F(X,Y) - ПРАВОЇ ЧАСТИНИ
C ДИФЕРЕНЦІАЛЬНОГО РІВНЯННЯ В ЦЕНТРІ ВАГИ ТРИКУТНИКА
 I=INTG(1)
 J=INTG(2)
 K=INTG(3)
 XO=(X(I)+X(J)+X(K))/3.
 YO=(Y(I)+Y(J)+Y(K))/3.
 SOURCI=RIGHT(XO,YO)
C ЧИ Є ДОПУСТИМОЮ КІЛЬКІСТЬ ЕЛЕМЕНТІВ?
 NELMTS=NELMTS+1
 IF(NELMTS.LE.MAXELM) GO TO 60
 WRITE(KONSOL,201) NELMTS,MAXELM
 GO TO 130
C ВКЛЮЧЕННЯ НОВОГО ЕЛЕМЕНТА В СПИСОК; ПЕРЕХІД ДО
C ЗНАХОДЖЕННЯ НАСТУПНОГО
60 SOURCE(NELMTS)=SOURCI
 DO 70 M=1,NVE
70 NVTX(M,NELMTS)=INTG(M)
 GO TO 50
C ЗЧИТУВАННЯ СПИСКУ ВУЗЛІВ, В ЯКИХ ЗАДАНО УМОВУ ДІРІХЛЕ
80 DO 90 M=1,NODES
 POTENT(M)=0.
90 CONSTR(M)=.FALSE.
100 READ(INPUT,*) I
C ПЕРЕВІРКА ОЗНАКИ КІНЦЯ (НУЛЬОВИЙ РЯДОК)
 IF(I.LE.0) GO TO 120
C ПЕРЕВІРКА ДОПУСТИМОСТІ НОМЕРА
 IF(I.LE.NODES) GO TO 110
 WRITE(KONSOL,200) I,NODES
 GO TO 130
C ЗАПИС ВЕЛИЧИН ТА ПЕРЕХІД ДО НАСТУПНОЇ
110 CONSTR(I)=.TRUE.
 POTENT(I)=0.0
 GO TO 100
C ПОВЕРНЕННЯ ДО ВИЗИВНОЇ ПРОГРАМИ
120 IERR=0
 GO TO 140
130 IERR=1

```

```

140 RETURN
200 FORMAT(1X, 'НОМЕР ВУЗЛА', I3, ' MAXIMUM=', I3)
201 FORMAT(1X, 'НОМЕР ЕЛЕМЕНТА', I4, ' MAXIMUM=', I3)
END

```

SUBROUTINE SZERO

```

CC
C ПІДПРОГРАМА ОБНУЛЕННЯ ГЛОБАЛЬНОЇ МАТРИЦІ S I C
C ПРАВОЇ ЧАСТИНИ RTHDSD C
C ПРИМІТКА: ПІДПРОГРАМА ВИКОРИСТОВУЄ НЕПОІМЕНОВАНИЙ C
C ----- COMMON - БЛОК, ЯКИЙ ОПИСАНИЙ В ПІДПРО- C
C ГРАМІ MFE . C
CC
LOGICAL CONSTR
DIMENSION X(50), Y(50), S(50, 50),
* CONSTR(50), RTHDSD(50), POTENT(50),
* NVTX(3, 75), SOURCE(75), SELM(3, 3),
* TELM(3, 3), INTG(3)
COMMON INPUT, KONSOL, IERR, NVE, MAXNOD, MAXELM,
* NELMTS, X, Y, NVTX, SOURCE, CONSTR, NODES,
* POTENT, RTHDSD, SELM, TELM, S, INTG
C ОБНУЛЕННЯ КОЖНОГО ЕЛЕМЕНТА МАТРИЦІ S
DO 20 I=1, NODES
DO 10 J=1, NODES
S(I, J)=0.
10 CONTINUE
20 CONTINUE
C ОБНУЛЕННЯ ПРАВОЇ ЧАСТИНИ
DO 30 I=1, NODES
RTHDSD(I)=0.
30 CONTINUE
RETURN
END

```

SUBROUTINE ELMATR(IE)

```

CC
C ПІДПРОГРАМА ПОБУДОВИ МАТРИЧНИХ ЕЛЕМЕНТІВ S I T C
C ДЛЯ ОДНОГО ТРИКУТНОГО ЕЛЕМЕНТА C
C ВХІДНІ ПАРАМЕТРИ: C
C ----- C
C IE - НОМЕР ЕЛЕМЕНТА. C
C ПРИМІТКА: ПІДПРОГРАМА ВИКОРИСТОВУЄ НЕПОІМЕНОВАНИЙ C
C ----- COMMON - БЛОК, ЯКИЙ ОПИСАНИЙ В ПІДПРО- C
C ГРАМІ MFE . C
CC
LOGICAL CONSTR
DIMENSION X(50), Y(50), S(50, 50),
* CONSTR(50), RTHDSD(50), POTENT(50),
* NVTX(3, 75), SOURCE(75), SELM(3, 3),
* TELM(3, 3), INTG(3)
COMMON INPUT, KONSOL, IERR, NVE, MAXNOD, MAXELM,
* NELMTS, X, Y, NVTX, SOURCE, CONSTR, NODES,

```

POTENT, RTHDSD, SELM, TELM, S, INTG  
C УСТАНОВЛЕННЯ ІНДЕКСІВ ДЛЯ ТРИКУТНИКА

I=NVTX(1, IE)  
J=NVTX(2, IE)  
K=NVTX(3, IE)

C ОБЧИСЛЕННЯ Т-МАТРИЦІ ЕЛЕМЕНТА

AREA=ABS((X(J)-X(I))\*(Y(K)-Y(I))-  
\*(X(K)-X(I))\*(Y(J)-Y(I)))\*.5

DO 20 L=1, NVE  
DO 10 M=1, NVE

10 TELM(L, M)=AREA/12.  
20 TELM(L, L)=2.\*TELM(L, L)

C ОБЧИСЛЕННЯ S-МАТРИЦІ ЕЛЕМЕНТА

I1=1  
I2=2  
I3=3  
DO 30 L=1, NVE  
DO 30 M=1, NVE

30 SELM(L, M)=0.  
DO 40 NVRTEX=1, 3

\* C=(X(J)-X(I))\*(X(K)-X(I))+  
(Y(J)-Y(I))\*(Y(K)-Y(I))

C4A=C/(4.\*AREA)  
SELM(I2, I2)=SELM(I2, I2)+C4A  
SELM(I2, I3)=SELM(I2, I3)-C4A  
SELM(I3, I2)=SELM(I3, I2)-C4A  
SELM(I3, I3)=SELM(I3, I3)+C4A

C ПЕРЕСТАНОВКА ІНДЕКСІВ РЯДКІВ І СТОВПЦІВ

I4=I1  
I1=I2  
I2=I3  
I3=I4  
L=I  
I=J  
J=K  
K=L

40 CONTINUE  
RETURN  
END

SUBROUTINE ELEMBD(IE)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC  
C ПІДПРОГРАМА ВКЛЮЧЕННЯ ОДИНОЧНИХ МАТРИЧНИХ ЕЛЕМЕНТІВ  
C ТИП S I T (З ПОЛІВ SELM I TELM У СПІЛЬНОМУ COMMON-С  
C БЛОЦІ) В ГЛОБАЛЬНУ МАТРИЦЮ S І ВЕКТОР RTHDSD C  
C ВХІДНІ ПАРАМЕТРИ: C  
C ----- C

C IE - НОМЕР ЕЛЕМЕНТА.

C ПРИМІТКА: ПІДПРОГРАМА ВИКОРИСТОВУЄ НЕПОІМЕНОВАНИЙ  
C ----- COMMON - БЛОК, ЯКИЙ ОПИСАНИЙ В ПІДПРО- C  
C ГРАМІ MFE . C

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

LOGICAL CONSTR
DIMENSION X(50), Y(50), S(50, 50),
* CONSTR(50), RTHDSD(50), POTENT(50),
* NVTX(3, 75), SOURCE(75), SELM(3, 3),
* TELM(3, 3), INTG(3)
COMMON INPUT, KONSOL, IERR, NVE, MAXNOD, MAXELM,
* NELMTS, X, Y, NVTX, SOURCE, CONSTR, NODES,
* POTENT, RTHDSD, SELM, TELM, S, INTG
C ПЕРЕБІР МАТРИЧНИХ ЕЛЕМЕНТІВ S I T (SELM I, TELM) I
C КОРЕКТУВАННЯ ГЛОБАЛЬНОЇ МАТРИЦІ S I ПРАВОЇ ЧАСТИНИ
DO 40 I=1, NVE
 IROW=NVTX(I, IE)
C ЧИ ВІДПОВІДАЄ РЯДОК ФІКСОВАНОМУ ЗНАЧЕННЮ РОЗВ'ЯЗКУ ?
IF (CONSTR(IROW)) GO TO 30
C НІ - ПЕРЕГЛЯД NVE-СТОВПЦІВ
DO 20 J=1, NVE
 ICOL=NVTX(J, IE)
C ЧИ ВІДПОВІДАЄ СТОВПЕЦЬ ФІКСОВАНОМУ ЗНАЧЕННЮ
C РОЗВ'ЯЗКУ ?
IF (CONSTR(ICOL)) GO TO 10
C НІ, ТОДІ ЗБІЛЬШЕННЯ S I RTHDSD
S(IROW, ICOL)=S(IROW, ICOL)+SELM(I, J)
RTHDSD(IROW)=RTHDSD(IROW)+TELM(I, J)*
* SOURCE(IE)
GO TO 20
C ТАК, ТОДІ ЗБІЛЬШЕННЯ ТІЛЬКИ ПРАВОЇ ЧАСТИНИ
10 CONTINUE
RTHDSD(IROW)=RTHDSD(IROW)+TELM(I, J)*
* SOURCE(IE)-SELM(I, J)*
* POTENT(ICOL)
20 CONTINUE
GO TO 40
C УСТАНОВЛЕННЯ ГЛОБАЛЬНОЇ S-МАТРИЦІ I RTHDSD
C ДЛЯ ЗАДАНОГО ВУЗЛА
30 CONTINUE
S(IROW, IROW)=1.
RTHDSD(IROW)=POTENT(IROW)
40 CONTINUE
C ВСЕ ВИКОНАНО - ПОВЕРНЕННЯ У ВИЗИВНУ ПРОГРАМУ
RETURN
END

```

#### SUBROUTINE OUTPUT

```

CC
C ПІДПРОГРАМА ВИВЕДЕННЯ РЕЗУЛЬТАТІВ C
C ПРИМІТКА: ПІДПРОГРАМА ВИКОРИСТОВУЄ НЕПОІМЕНОВАНИЙ C
C ----- COMMON - БЛОК, ЯКИЙ ОПИСАНИЙ В ПІДПРО- C
C ГРАМІ MFE. C
CC

```

```

LOGICAL CONSTR
DIMENSION X(50), Y(50), S(50, 50),
* CONSTR(50), RTHDSD(50), POTENT(50),
* NVTX(3, 75), SOURCE(75), SELM(3, 3),

```

TELM(3,3),INTG(3)

COMMON INPUT,KONSOL,IERR,NVE,MAXNOD,MAXELM,  
NELMTS,X,Y,NVTX,SOURCE,CONSTR,NODES,  
POTENT,RTHDS,SELM,TELM,S,INTG

C ВИВЕДЕННЯ ВУЗЛІВ І ЗНАЧЕНЬ РОЗВ'ЯЗКУ

WRITE (KONSOL,100) (I,X(I),Y(I),POTENT(I),  
I=1,NODES)

RETURN

100 FORMAT(1X////15X,'РОЗВ'ЯЗОК ЗАДАЧІ'/  
/2X,'ВУЗЛІВ',8X,'X',9X,'Y',11X,'U',/  
/(1X,I3,2X,F10.5,F10.5,3X,F10.5))  
END

SUBROUTINE STEP1(T,TAU,Y,N,FUN,Q,RK,RMU1,  
IERR,WORK)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C ПІДПРОГРАМА STEP1 РЕАЛІЗУЄ ОДИН КРОК (ПО ЗМІННИЙС  
C ЧАСУ) РІЗНИЦЕВОЇ СХЕМИ З ВАГОВИМИ КОЕФІЦІЄНТАМИ  
C РОЗВ'ЯЗУВАННЯ ОДНОВИМІРНОГО РІВНЯННЯ ТЕПЛОПРОВІД-  
C НОСТІ В ДЕКАРТОВИХ, ЦИЛІНДРИЧНИХ АБО СФЕРИЧНИХ  
C КООРДИНАТАХ

C ЗАГАЛЬНИЙ ВИГЛЯД РІВНЯННЯ :

C  $D/DT(U) = 1/(X**M)*D/DX(X**M*K(X,T)*D/DX(U)) -$  (1)  
C  $Q(X,T)*U+F(X,T), X=(A,B)$

C  $ALFA1*K(X,T)*D/DX(U) - BT1*U = -RMU1(T),$  ПРИ  $X=A,$  (2)  
C  $ALFA2*K(X,T)*D/DX(U) + BT2*U = RMU2(T),$  ПРИ  $X=B,$  (3)

C  $U(X,0) = UO(X),$  (4)  
C ДЕ  $M=0,1,2, B > A.$

C ВХІДНІ ПАРАМЕТРИ :  
C -----

C T - ЗНАЧЕННЯ ЗМІННОЇ ЧАСУ, ЩО ВІДПОВІДАЄ  
C J -МУ ЯРУСУ, ТОБТО  $T=J*TAU;$

C TAU - ЗНАЧЕННЯ КРОКУ ПО ЗМІННІЙ ЧАСУ;

C Y - ОДНОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ N, ЯКИЙ  
C МІСТИТЬ РОЗВ'ЯЗОК НА J -МУ ЯРУСІ;

C N - КІЛЬКІСТЬ ВУЗЛІВ СІТКИ В НАПРЯМКУ X;

C FUN,Q,RK - ПІДПРОГРАМИ-ФУНКЦІЇ FUN(X,T),Q(X,T),  
C RK(X,T), ЯКІ ОБЧИСЛЮЮТЬ ВІДПОВІДНО ПРАВУ  
C ЧАСТИНУ РІВНЯННЯ, КОЕФІЦІЄНТ ТЕПЛОВІДДАЧІ  
C ТА КОЕФІЦІЄНТ ТЕПЛОПРОВІДНОСТІ;

C RMU1 - ПІДПРОГРАМА-ФУНКЦІЯ RMU1(T,RMU2), ЩО ОБЧИС-  
C ЛЮЄ ПРАВУ ЧАСТИНУ КРАЙОВИХ УМОВ, ПРИЧОМУ  
C ІМЕНІ ФУНКЦІЇ RMU1 НАДАЄТЬСЯ ЗНАЧЕННЯ ПРАВОЇС  
C ЧАСТИНИ КРАЙОВИХ УМОВ НА ЛІВІЙ МЕЖІ, А  
C ПАРАМЕТРУ RMU2 - НА ПРАВІЙ МЕЖІ;

C WORK - РОБОЧИЙ МАСИВ РОЗМІРНІСТЮ  $5*N-1$ ; ПРИ ПЕР-  
C ШОМУ ЗВЕРНЕННІ ДО ПІДПРОГРАМИ STEP1  
C ПЕРШИМ ВОСЬМИ ЕЛЕМЕНТАМ МАСИВУ НАДАЮТЬСЯ  
C ТАКІ ЗНАЧЕННЯ :

C WORK(1)=A - ЛІВА МЕЖА ВІДРІЗКА,

C WORK(2)=B - ПРАВА МЕЖА ВІДРІЗКА,

C WORK(3)=SIGMA - ЗНАЧЕННЯ ВАГОВОГО КОЕФІЦІЄНТА  
C РІЗНИЦЕВОЇ СХЕМИ,

C WORK(4)=ALFA1 - ЗНАЧЕННЯ КОЕФІЦІЄНТА З КРАЙОВОЇ  
 C УМОВИ (2),  
 C WORK(5)=ALFA2 - ЗНАЧЕННЯ КОЕФІЦІЄНТА З КРАЙОВОЇ  
 C УМОВИ (3),  
 C WORK(6)=BT1 - ЗНАЧЕННЯ КОЕФІЦІЄНТА ТЕПЛОВІДАЧІ  
 C НА ЛІВІЙ МЕЖІ,  
 C WORK(7)=BT2 - ЗНАЧЕННЯ КОЕФІЦІЄНТА ТЕПЛОВІДАЧІ  
 C НА ПРАВІЙ МЕЖІ,  
 C WORK(8)=M, ДЕ M=0,1,2 У ВИПАДКУ ДЕКАРТОВИХ,  
 C ЦИЛІНДРИЧНИХ АБО СФЕРИЧНИХ КООРДИНАТ  
 C ВІДПОВІДНО.  
 C ВИХІДНІ ПАРАМЕТРИ :

T - ЗНАЧЕННЯ ЗМІННОЇ ЧАСУ, ЯКЕ ВІДПОВІДАЄ  
 (J+1) -МУ ЯРУСУ, ТОБТО  $T=(J+1)*TAU$ ;  
 Y - МАСИВ, ЯКИЙ МІСТИТЬ РОЗВ'ЯЗОК НА (J+1) -МУ  
 ЯРУСІ;

IERR - ЯКЩО ПРИ ПЕРШОМУ ЗВЕРНЕННІ ДО ПІДПРОГРАМИ  
 IERR=-1, ТО ВХІДНІ ПАРАМЕТРИ ЗАДАНО НЕВІР-  
 НО, В ІНШИХ ВИПАДКАХ IERR=0.

ПІДПРОГРАМИ, ЩО ВИКОРИСТОВУЮТЬСЯ : STEP - ПІД-  
 ПРОГРАМА, ЯКАС

ФАКТИЧНО РЕАЛІЗУЄ ОДИН КРОК (ПО ЗМІННІЙ ЧАСУ)  
 РІЗНИЦЕВОЇ СХЕМИ РОЗВ'ЯЗУВАННЯ ОДНОВИМІРНОГО  
 РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ.

ЗАУВАЖЕННЯ: 1) ФАКТИЧНІ ПАРАМЕТРИ, ЩО ВІДПОВІДАЮТЬС  
 ФОРМАЛЬНИМ ПАРАМЕТРАМ FUN, Q, RK, RMU1,

МАЮТЬ БУТИ ОПИСАНІ В ОПЕРАТОРІ EXTERNAL  
 ПРОГРАМИ, ЩО ВИКЛИКАЄ ПІДПРОГРАМУ STEP;

2) ЗНАЧЕННЯ ФУНКЦІЇ Q, А ТАКОЖ ЗМІННИХ BT1, BT2  
 ПОВИННІ БУТИ НЕВІД'ЄМНИМИ, А ФУНКЦІЇ RK -  
 ДОДАТНИМИ;

3) ЗМІННІ ALFA1, ALFA2 ПОВИННІ НАБУВАТИ ЗНАЧЕННЯ  
 0 АБО 1;

4) ЯКЩО РОЗГЛЯДАЄТЬСЯ РІВНЯННЯ В КРИВОЛІНІЙНИХ  
 КООРДИНАТАХ ( M=1,2 ) І A=0, ТО ПРИ X=A НЕ  
 МОЖУТЬ БУТИ ЗАДАНІ КРАЙОВІ УМОВИ ПЕРШОГО РОДУ,  
 ТОБТО ЧЕТВЕРТОМУ ЕЛЕМЕНТУ МАСИВУ WORK ТРЕБА  
 НАДАТИ ЗНАЧЕННЯ 1 (ТОБТО WORK(4)=1);

5) ПІСЛЯ ЗАВЕРШЕННЯ РОБОТИ ПІДПРОГРАМИ ЕЛЕМЕНТИ  
 МАСИВУ WORK НЕ ЗБЕРІГАЮТЬСЯ.

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

REAL Y(N), WORK(1)  
 EXTERNAL FUN, Q, RK, RMU1  
 DATA KSH/O/  
 IF (KSH.NE.0) GO TO 10  
 IERR=0

C ПЕРЕВІРКА ВХІДНИХ ДАНИХ

IF (TAU.LE.0..OR.WORK(2).LE.WORK(1)..OR.T.LT.0.  
 \* .OR.WORK(3).LT.0..OR.WORK(3).GT.1..OR.  
 \* WORK(4).GT.1..OR.WORK(4).LT.0..OR.WORK(5)  
 \* .GT.1..OR.WORK(5).LT.0..OR.WORK(8).LT.0.



```

* . OR. WORK(8) . GT. 2. . OR. N. LT. 3. . OR. WORK(6)
* . LT. 0. . OR. WORK(7) . LT. 0.) IERR=-1
IF (WORK(1) . EQ. 0. . AND. WORK(8) . GE. 1. . AND
* WORK(4) . EQ. 0.)
* IERR=-1
RETURN
IF (IERR. NE. 0)
KSH=1

```

C РОЗПОДІЛ ЕЛЕМЕНТІВ РОБОЧОГО МАСИВУ WORK

```

N1=N-1
K1=N+1
K2=K1+N
K3=K2+N1
K4=K3+N

```

```

10 CALL STEP (T, TAU, Y, N, N1, FUN, Q, RK, RMU1, WORK(1),
* WORK(K1), WORK(K2), WORK(K3), WORK(K4))
RETURN
END

```

SUBROUTINE STEP (T, TAU, Y, N, N1, FUN, Q, RK, RMU1, V, X, XM, RM, RW)

```

CC
C ПІДПРОГРАМА STEP ВИКЛИКАЄТЬСЯ З ПРОГРАМИ STEP1 C
C І ФАКТИЧНО РЕАЛІЗУЄ ОДИН КРОК (ПО ЗМІННІЙ ЧАСУ) C
C РОЗВ'ЯЗУВАННЯ ОДНОВИМІРНОГО РІВНЯННЯ C
C ТЕПЛОПРОВІДНОСТІ C
C ВХІДНІ ПАРАМЕТРИ : C
C ----- C
C X - ОДНОВИМІРНИЙ МАСИВ, ЯКИЙ МІСТИТЬ ЗНАЧЕННЯ C
C ВУЗЛІВ СІТКИ ПО ПРОСТОРОВІЙ ЗМІННІЙ; C
C XM - ОДНОВИМІРНИЙ МАСИВ, ЯКИЙ МІСТИТЬ C
C КОЕФІЦІЄНТИ X**M; C
C V - ОДНОВИМІРНИЙ МАСИВ УСЕРЕДНЕНЬ МАСИВУ XM; C
C RM, RW - ОДНОВИМІРНІ РОБОЧІ МАСИВИ. C
C ІНШІ ВХІДНІ ТА ВХІДНІ ПАРАМЕТРИ МАЮТЬ ТОЙ САМИЙ C
C ЗМІСТ, ЩО І В ПІДПРОГРАМІ STEP1. C
C ЗАУВАЖЕННЯ : ЗНАЧЕННЯ ЕЛЕМЕНТІВ МАСИВІВ X, XM, V, C
C ----- RM, RW ЗБЕРІГАЮТЬСЯ ПРИ ДРУГОМУ І C
C НАСТУПНИХ ВИКЛИКАХ ПРОГРАМИ STEP. C
CC

```

```

REAL Y(N), X(N), V(N), XM(N1), RM(N), RW(N)
DATA KSH/0/

```

C ОБЧИСЛЕННЯ ВХІДНИХ ДАНИХ ПРИ ПЕРШОМУ ВИКЛИКУ ПІДПРОГРАМИ

```

IF (KSH. NE. 0) GO TO 60
A=V(1)
B=V(2)
SIGMA=V(3)
IALFA1=INT(V(4)+0.5)
IALFA2=INT(V(5)+0.5)
BT1=V(6)
BT2=V(7)
M=INT(V(8)+0.5)
H=(B-A)/FLOAT(N1)

```

```

H2=H*H
HD2=0.5*H
RKUR=TAU/H2
TAUH=TAU/H
SIGMA1=1.-SIGMA
N1N=2-IALFA1
N1N1=N1N+1
N1V=N1+IALFA2
N1V1=N1V-1
X(1)=A
DO 10 I=2, N
10 X(I)=X(I-1)+H
 IF(M.GE.1) GO TO 30
DO 20 I=1, N1
20 V(I)=1.
 XM(I)=1.
V(N)=1.
GO TO 50
30 DO 40 I=1, N1
40 V(I)=X(I)**M+(M-1)*H2/3
 XM(I)=(X(I)+HD2)**M
V(1)=X(1)**M+(M-1)*H*X(1)+H**M/(M+1)
V(N)=X(N)**M-(M-1)*H*X(N)+(1-2*MOD(M,2))*
* H**M/(M+1)
50 CONTINUE
 IF(M.EQ.0) X1=1
 IF(M.GE.1 AND.A.EQ.0.) X1=0
 IF(M.GE.1 AND.A.GT.0) X1=A**M
 XN=B**M
 KSH=1

```

C  
60

```

CONTINUE
TP=T+SIGMA*TAU
RM1=RMU1(TP, RM2)
IF(IALFA1.EQ.0) RM1=RM1/BT1
IF(IALFA2.EQ.0) RM2=RM2/BT2
RK1=RK(X(1)+HD2, TP)
RKP=RK(X(N1N)+HD2, TP)
RQ=Q(X(N1N), TP)

```

C ПРЯМИЙ ХІД МЕТОДУ ПРОГОНКИ

```

BI=XM(N1N)*SIGMA*RKUR*RKP
CI=-BI-(1-IALFA1)*SIGMA*RKUR*XM(1)*RK1-
* IALFA1*BT1*X1*SIGMA*TAUH-V(N1N)*
* (1.+RQ*SIGMA*TAU)/(1+IALFA1)
RM(N1N1)=-BI/CI
RW(N1N1)=(-SIGMA1*RKUR*(XM(N1N)*RKP*
* (Y(N1N1)-Y(N1N))-(1-IALFA1)*XM(1)*
* RK1*Y(2))+V(N1N)/(1+IALFA1))*(RQ*TAU*
* SIGMA1*Y(N1N)-Y(N1N)-TAU*
* FUN(X(N1N), TP))-RKUR*H**IALFA1*RM1*
* (X1+(1-IALFA1)*(XM(1)*RK1-X1))+
* IALFA1*BT1*X1*TAUH*SIGMA1*Y(1))/CI

```

BI=XM(N1N)\*RKUR\*RKP  
DO 70 I=N1N1, N1V1

DI=BI

BI=XM(I)\*RKUR\*RK(X(I)+HD2, TP)

RQ=Q(X(I), TP)

PI=-V(I)\*(1.+TAU\*SIGMA\*RQ)-((DI+BI)-

DI\*RM(I))\*SIGMA

RM(I+1)=-BI\*SIGMA/PI

RW(I+1)=(-V(I)\*Y(I)-SIGMA1\*(BI\*(Y(I+1)-

Y(I))-DI\*(Y(I)-Y(I-1)))+TAU\*

SIGMA1\*V(I)\*RQ\*Y(I)-TAU\*V(I)\*

FUN(X(I), TP)-DI\*SIGMA\*RW(I))/PI

DI=BI\*SIGMA

RKN=RK(X(N1)+HD2, TP)

RKP=RK(X(N1V1)+HD2, TP)

RQ=Q(X(N1V), TP)

CI=-DI-(1-IALFA2)\*SIGMA\*RKUR\*XM(N1)\*RKN-

IALFA2\*BT2\*XN\*SIGMA\*TAUH-V(N1V)/(1+IALFA2)\*

(1.+RQ\*SIGMA\*TAU)

Y(N1V)=(SIGMA1\*RKUR\*(XM(N1V1)\*RKP\*

(Y(N1V)-Y(N1V1))+(1-IALFA2)\*XM(N1)\*RKN\*

Y(N1))+V(N1V)/(1+IALFA2)\*(RQ\*TAU\*

SIGMA1\*Y(N1V)-Y(N1V)-TAU\*

FUN(X(N1V), TP))-RKUR\*H\*\*IALFA2\*RM2\*

(XN+(1-IALFA2)\*(XM(N1)\*RKN-XN))+IALFA2\*

BT2\*XN\*SIGMA1\*TAUH\*Y(N)-DI\*RW(N1V))/

(CI+DI\*RM(N1V))

C ЗВОРОТНИЙ ХІД МЕТОДУ ПРОГОНКИ

DO 80 I=N1N1, N1V

IR=N1V+N1N-I

IR1=IR+1

Y(IR)=RM(IR1)\*Y(IR1)+RW(IR1)

T=T+TAU

RM1=RMU1(T, RM2)

IF(IALFA1.EQ.0) Y(1)=RM1/BT1

IF(IALFA2.EQ.0) Y(N)=RM2/BT2

RETURN

END

SUBROUTINE PPS(T, TAU, NP1, NP2, N1, N2, Y, Y1, A, B)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C ПІДПРОГРАМА PPS РЕАЛІЗУЄ ОДИН КРОК (ПО ЗМІННІЙ C C

C ЧАСУ) АЛГОРИТМУ РОЗВ'ЯЗУВАННЯ ДВОВИМІРНОГО C C

C РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ В ПРЯМОКУТНИКУ C C

C [O, L1]X[O, L2] ЗА ДОПОМОГОЮ PPS C C

C ВХІДНІ ПАРАМЕТРИ : C C

C ----- C C

C T - ЗНАЧЕННЯ ЗМІННОЇ ЧАСУ, ЩО ВІДПОВІДАЄ J-МУ C C

C ЯРУСУ, ТОБТО T=J\*TAU; C C

C TAU - ЗНАЧЕННЯ КРОКУ ПО ЗМІННІЙ ЧАСУ; C C

C NP1 - КІЛЬКІСТЬ ВУЗЛІВ СІТКИ В НАПРЯМКУ X1; C C

C NP2 - КІЛЬКІСТЬ ВУЗЛІВ СІТКИ В НАПРЯМКУ X2; C C

C N1 - ЗНАЧЕННЯ КРОКУ СІТКИ В НАПРЯМКУ X1. C C

```

C H2 - ЗНАЧЕННЯ КРОКУ СІТКИ В НАПРЯМКУ X2;
C Y - ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ (NP1, NP2),
C ЯКИЙ МІСТИТЬ РОЗВ'ЯЗОК НА J -МУ ЯРУСІ;
C Y1 - РОБОЧИЙ ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ
C (NP1, NP2);
C A, B - РОБОЧИ ОДНОВИМІРНІ МАСИВИ, РОЗМІРНІСТЬ ЯКИХ
C ДОРІВНЮЄ МАКСИМАЛЬНОМУ З ДВОХ ЧИСЕЛ:
C (NP1-1) І (NP2-1).
C ВИХІДНІ ПАРАМЕТРИ :
C -----
C T - ЗНАЧЕННЯ ЗМІННОЇ ЧАСУ, ЩО ВІДПОВІДАЄ
C (J+1) -МУ ЯРУСУ, ТОБТО $T=(J+1)*\text{TAU}$;
C Y - МАСИВ, ЯКИЙ МІСТИТЬ РОЗВ'ЯЗОК НА (J+1) -МУ
C ЯРУСІ.
C ПІДПРОГРАМИ, ЩО ВИКОРИСТОВУЮТЬСЯ : G1(X2, T),
C ----- G2(X1, T),
C G3(X2, T), G4(X1, T), F(X1, X2, T) - ПІДПРОГРАМИ-ФУНКЦІЇ,
C ЯКІ ОБЧИСЛЮЮТЬ КРАЙОВІ УМОВИ ТА ПРАВУ ЧАСТИНУ
C РІВНЯННЯ.
CC
REAL Y(NP1, NP2), Y1(NP1, NP2), A(1) B(1)
N1=NP1-1
N2=NP2-1
T1=TAU/(N1*N1)
T2=TAU/(N2*N2)
T3=T+0.5*TAU
T4=T+TAU
X2=0.
X3=X2+H2
R11=G1(X2, T)
R12=G1(X2, T4)
R13=G1(X3, T)
R14=G1(X3, T4)
R31=G3(X2, T)
R32=G3(X2, T4)
R33=G3(X3, T)
R34=G3(X3, T4)
DO 30 J=2, N2
X2=X3
X3=X2+H2
A(1)=0.
R15=G1(X3, T)
R16=G1(X3, T4)
B(1)=0.5*(R13+R14)-0.25*T2*(R16-R15-
* 2.*(R14-R13)+R12-R11)
R11=R13
R12=R14
R13=R15
R14=R16
X1=0.
DO 10 I=2, N1
X1=X1+H1

```

```

 S=2. *(T1+1.)-A(I-1)*T1
 A(I)=T1/S
 FF=2. *Y(I, J)+T2*(Y(I+1, J)-2. *Y(I, J)+
10 * Y(I-1, J))+TAU*F(X1, X2, T3)
 B(I)=(T1*B(I-1)+FF)/S
 R35=G3(X3, T)
 R36=G3(X3, T4)
 Y1(NP1, J)=0. 5*(R33+R34)-0. 25*T2*(R36-R35-
 * 2. *(R34-R33)+R32-R31)
 DO 20 II=1, N1
20 I=NP1-II
 Y1(I, J)=A(I)*Y1(I+1, J)+B(I)
 R31=R33
 R32=R34
 R33=R35
 R34=R36
30 CONTINUE
 T=T4
 X1=0.
 DO 60 I=2, N1
 X1=X1+H1
 A(1)=0.
 B(1)=G2(X1, T)
 X2=0.
 DO 40 J=2, N2
 X2=X2+H2
 S=2. *(T2+1.)-A(J-1)*T2
 A(J)=T2/S
 FF=2*Y1(I, J)+T1*(Y1(I+1, J)-2. *Y1(I, J)+
40 * Y1(I-1, J))+TAU*F(X1, X2, T3)
 B(J)=(T2*B(J-1)+FF)/S
 Y(I, NP2)=G4(X1, T)
 DO 50 JJ=1, N2
 J=NP2-JJ
50 Y(I, J)=A(J)*Y(I, J+1)+B(J)
60 CONTINUE
 X2=0.
 DO 70 J=1, NP1
 Y(1, J)=G1(X2, T)
 Y(NP1, J)=G3(X2, T)
70 X2=X2+H2
 RETURN
 END

```

```

 SUBROUTINE LOC(T, TAU, NP1, NP2, H1, H2, Y, Y1, A, B)
CC
C ПІДПРОГРАМА LOC РЕАЛІЗУЄ ОДИН КРОК (ПО ЗМІННІЙ C
C ЧАСУ) АЛГОРИТМУ РОЗВ'ЯЗУВАННЯ ДВОВИМІРНОГО C
C РІВНЯННЯ ТЕПЛОПРОВІДНОСТІ В ПРЯМОКУТНИКУ C
C [0, L1]X[0, L2] ЗА ДОПОМОГОЮ LOC C
C ВХІДНІ ПАРАМЕТРИ : C
C ----- C
C T - ЗНАЧЕННЯ ЗМІННОЇ ЧАСУ, ЩО ВІДПОВІДАЄ J-МУ C

```

```

C ЯРУСУ, ТОБТО T=J*TAU;
C TAU - ЗНАЧЕННЯ КРОКУ ПО ЗМІННІЙ ЧАСУ;
C NP1 - КІЛЬКІСТЬ ВУЗЛІВ СІТКИ В НАПРЯМКУ X1;
C NP2 - КІЛЬКІСТЬ ВУЗЛІВ СІТКИ В НАПРЯМКУ X2;
C H1 - ЗНАЧЕННЯ КРОКУ СІТКИ В НАПРЯМКУ X1;
C H2 - ЗНАЧЕННЯ КРОКУ СІТКИ В НАПРЯМКУ X2;
C Y - ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ (NP1, NP2),
C ЯКИЙ МІСТИТЬ РОЗВ'ЯЗОК НА J -МУ ЯРУСІ;
C Y1 - РОБОЧИЙ ДВОВИМІРНИЙ МАСИВ РОЗМІРНІСТЮ
C (NP1, NP2);
C A, B - РОБОЧИ ОДНОВИМІРНІ МАСИВИ, РОЗМІРНІСТЬ ЯКИХ
C ДОРІВНЮЄ МАКСИМАЛЬНОМУ З ДВОХ ЧИСЕЛ:
C (NP1-1) І (NP2-1).
C ВИХІДНІ ПАРАМЕТРИ :
C -----
C T - ЗНАЧЕННЯ ЗМІННОЇ ЧАСУ, ЩО ВІДПОВІДАЄ
C (J+1) -МУ ЯРУСУ, ТОБТО T=(J+1)*TAU;
C Y - МАСИВ, ЯКИЙ МІСТИТЬ РОЗВ'ЯЗОК НА (J+1) -МУ
C ЯРУСІ.
C ПІДПРОГРАМИ, ЩО ВИКОРИСТОВУЮТЬСЯ : G1(X2, T),
C G2(X1, T),
C G3(X2, T), G4(X1, T), F(X1, X2, T) - ПІДПРОГРАМИ-ФУНКЦІЇ,
C ЯКІ ОБЧИСЛЮЮТЬ КРАЙОВІ УМОВИ ТА ПРАВУ ЧАСТИНУ
C РІВНЯННЯ.
C CCC
 REAL Y(NP1, NP2), Y1(NP1, NP2), A(1), B(1)
 N1=NP1-1
 N2=NP2-1
 T1=TAU/(H1*H1)
 T2=TAU/(H2*H2)
 T3=T+0.5*TAU
 X2=0.
 DO 30 J=2, N2
 X2=X2+H2
 A(1)=0.
 B(1)=G1(X2, T3)
 X1=0.
 DO 10 I=2, N1
 X1=X1+H1
 S=2.*T1+1.-A(I-1)*T1
 A(I)=T1/S
 B(I)=(T1*B(I-1)+Y(I, J)+0.5*TAU*
 F(X1, X2, T3))/S
 Y1(NP1, J)=G3(X2, T3)
 DO 20 II=1, N1
 I=NP1-II
 Y1(I, J)=A(I)*Y1(I+1, J)+B(I)
 CONTINUE
 T=T+TAU
 X1=0.
 DO 60 I=2, N1

```



```

C U1(X) - П/П-ФУНКЦІЯ ДЛЯ ДРУГОЇ ПОЧАТКОВОЇ УМОВИ; C
C G1(T) - П/П-ФУНКЦІЯ ДЛЯ ЛІВОЇ КРАЙОВОЇ УМОВИ; C
C G2(T) - П/П-ФУНКЦІЯ ДЛЯ ПРАВОЇ КРАЙОВОЇ УМОВИ; C
C AK(X), Q(X), F(X,T) - П/П-ФУНКЦІЇ ДЛЯ КОЕФІЦІЕН- C
C ТІВ РІВНЯННЯ. C
CC

```

```

DIMENSION YO(101), Y1(101)
COMMON /C1/ N, TAU, H
COMMON /C2/ M, A, B, AL1, AL2, BE1, BE2
IF(T.NE.O.) GO TO 10
NP1=N+1
HM=H*FLOAT(M+1)
T2=TAU*TAU
G=(TAU/H)**2

```

```

10 X=A
 XP=A+.5*H
 XM=A
 X1=1.
 X2=1.
 IO=1
 I1=NP1

```

```

C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ДЛЯ КРАЙОВИХ УМОВ ПЕРШОГО РОДУ

```

```

IF(AL1.NE.O.) GO TO 20
YO(1)=G1(T+TAU)/BE1
IO=2

```

```

X=A+H
XM=XP
XP=XM+H
20 IF(AL2.NE.O.) GO TO 30
YO(NP1)=G2(T+TAU)/BE2
I1=N

```

```

30 IF(M.GT.O) X1=XM**M
 R=AK(XP)

```

```

C ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ У ВНУТРІШНІХ ТОЧКАХ ТА НА
C КІНЦЯХ ДЛЯ КРАЙОВИХ УМОВ ТРЕТЬОГО РОДУ
DO 90 I=IO, I1

```

```

IF(I.EQ.NP1) XP=B
IF(M.GT.O) X2=XP**M
S=(X2*XP-X1*XM)/HM
IF(I.NE.1) GO TO 40
V=X2*R*(Y1(2)-Y1(1))+X1*H/AL1*
* (G1(T)-BE1*Y1(1))

```

```

V=G*V/S
GO TO 60
40 IF(I.EQ.NP1) GO TO 50
C=AK(XP)

```

```

V=G*(X2*C*(Y1(I+1)-Y1(I))-X1*R*
* (Y1(I)-Y1(I-1)))/S
GO TO 60

```

```

50 V=X1*R*(Y1(N)-Y1(NP1))+X2*H/AL2*
* (G2(T)-BE2*Y1(NP1))
V=G*V/S

```



```

60 V=V+T2*(F(X,T)-Q(X)*Y1(I))
 IF(T.EQ.0.) GO TO 70
С ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ ДЛЯ ШАРІВ, ПОЧИНАЮЧИ З ДРУГОГО
 Y0(I)=2.*Y1(I)-Y0(I)+V
 GO TO 80
С ОБЧИСЛЕННЯ РОЗВ'ЯЗКУ НА ПЕРШОМУ ШАРІ
70 Y0(I)=Y1(I)+TAU*U1(X)+.5*V
80 R=C
 X1=X2
 XM=XP
 XP=XM+H
 X=X+H
90 X=X+H
С ПЕРЕПРИЗНАЧЕННЯ МАСИВІВ РОЗВ'ЯЗКУ
 DO 100 I=1, NP1
 V=Y0(I)
 Y0(I)=Y1(I)
 Y1(I)=V
100 T=T+TAU
 RETURN
 END

```

## ПОКАЖЧИК РОЗМІЩЕННЯ ПІДПРОГРАМ

|              |                                                                                                                                   |     |
|--------------|-----------------------------------------------------------------------------------------------------------------------------------|-----|
| BISECT       | Підпрограма розв'язування нелінійного рівняння методом ділення навпіл                                                             | 199 |
| COAD         | Підпрограма визначення коефіцієнтів для неявних формул Адамса                                                                     | 258 |
| COSET        | Підпрограма встановлення значень коефіцієнтів інтегратора STIFF                                                                   | 256 |
| DD           | Підпрограма обчислення діагональних елементів таблиці розділених різниць                                                          | 216 |
| DEC          | Підпрограма декомпозиції вхідної матриці за допомогою методу виключень Гауса                                                      | 245 |
| DECOMP       | Підпрограма трикутного розкладання довільної матриці                                                                              | 203 |
| DRIVE        | Пакет підпрограм GEARZ розв'язування задачі Коші для звичайних диференціальних рівнянь                                            | 237 |
| EIGEN        | Підпрограма знаходження власних значень і власних векторів дійсної симетричної матриці                                            | 211 |
| ELEMBD       | Підпрограма включення одиночних матричних елементів у глобальну матрицю                                                           | 284 |
| ELMATR       | Підпрограма побудови матричних елементів для одного трикутного елемента                                                           | 283 |
| FACTSQ       | Підпрограма трикутного розкладання симетричної матриці                                                                            | 206 |
| FUNCTION FI  | Підпрограма-функція, яка обчислює поліномну координатну систему                                                                   | 272 |
| FUNCTION FII | Підпрограма-функція, яка обчислює значення підінтегральної функції                                                                | 273 |
| FUNCTION FIP | Підпрограма-функція, яка обчислює похідні від поліномної координатної системи                                                     | 273 |
| GALOR        | Підпрограма формування систем лінійних алгебраїчних рівнянь                                                                       | 270 |
| GAUSS        | Підпрограма обчислення визначеного інтеграла функції за квадратурною формулою Гауса                                               | 223 |
| GDD          | Підпрограма обчислення діагональних елементів таблиці розділених різниць з кратними вузлами                                       | 216 |
| GP1          | Підпрограма розв'язування мішаної задачі для одновимірного хвильового рівняння в разі плоскої, циліндричної та сферичної симетрії | 294 |
| GOR          | Підпрограма обчислення значення інтерполяційного многочлена за схемою Горнера                                                     | 215 |
| GRAFIK       | Підпрограма побудови графіка функцій однієї змінної, заданих координатами точок на площині                                        | 195 |
| GRAPH        | Підпрограма побудови графіка кількох функцій однієї змінної                                                                       | 194 |
| GRID         | Підпрограма побудови сітки на відрізку                                                                                            | 277 |
| INTERP       | Підпрограма обчислення за допомогою масиву передісторії Нордсіка інтерпольованих значень залежної змінної                         | 260 |

|        |                                                                                                                                                    |            |
|--------|----------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| LINUR  | Підпрограма побудови ліній рівня функцій двох змінних, заданої значеннями у вузлах нерівномірної прямокутної сітки                                 | 197        |
| LOC    | Підпрограма реалізації одного кроку алгоритму розв'язування двовимірного рівняння теплопровідності в прямокутнику                                  | 292        |
| MESHIN | Підпрограма введення потоку даних, що складається з трьох частин, ознакою кінця кожної з яких є нульовий рядок                                     | 281        |
| MFE    | Підпрограма розв'язування задачі Діріхле для рівняння Пуассона в опуклому многокутнику                                                             | 279        |
| NEWTON | Підпрограма розв'язування нелінійного рівняння методом Ньютона                                                                                     | 200        |
| NEWTS  | Підпрограма розв'язування системи $N$ нелінійних рівнянь з $N$ невідомими методом Ньютона                                                          | 210        |
| ORT    | Підпрограма ортонормування системи векторів і обчислення матриці коефіцієнтів ортогоналізації                                                      | 269<br>285 |
| OUTPUT | Підпрограма виведення результатів                                                                                                                  | 285        |
| PER    | Підпрограма перетворення многочлена, записаного у формі Ньютона, в многочлен, записаний по степенях $X$                                            | 215        |
| PPS    | Підпрограма реалізації одного кроку алгоритму розв'язування двовимірного рівняння теплопровідності в прямокутнику                                  | 290        |
| PRPM   | Підпрограма перемноження прямокутних матриць                                                                                                       | 267        |
| PRTM   | Підпрограма перемноження трикутних матриць                                                                                                         | 268        |
| PSET   | Підпрограма обчислення і перетворення матриці ітерації                                                                                             | 261        |
| QUANC8 | Підпрограма обчислення визначеного інтеграла за квадратною формулою Ньютона — Котеса                                                               | 224        |
| RESQ   | Підпрограма розв'язування системи лінійних алгебраїчних рівнянь                                                                                    | 208        |
| RKF45  | Підпрограма розв'язування системи звичайних диференціальних рівнянь методом Рунге — Кутти з автоматичним вибором кроку інтегрування                | 232        |
| RSTM   | Підпрограма розв'язування системи лінійних алгебраїчних рівнянь                                                                                    | 268        |
| SEVAL  | Підпрограма-функція знаходження значення сплайна в точці                                                                                           | 219        |
| SGL    | Підпрограма обчислення коефіцієнтів кубічного згладжуючого сплайна                                                                                 | 220        |
| SHTNG  | Підпрограма розв'язування лінійної крайової задачі для системи звичайних диференціальних рівнянь із застосуванням різних варіантів методу стрільби | 262        |
| SIMP   | Підпрограма обчислення визначеного інтеграла за формулою Сімпсона з заданою точністю                                                               | 222        |
| SISVD  | Підпрограма розв'язування системи лінійних алгебраїчних рівнянь у випадку перевизначеної системи                                                   | 274        |
| SITER  | Підпрограма розв'язування нелінійного рівняння методом простої ітерації                                                                            | 200        |
| SITERS | Підпрограма розв'язування системи нелінійних рівнянь методом простої ітерації                                                                      | 209        |
| SOL    | Підпрограма розв'язування системи лінійних алгебраїчних рівнянь                                                                                    | 244        |
| SOLVE  | Те саме                                                                                                                                            | 205        |
| SPLINE | Підпрограма обчислення коефіцієнтів кубічного інтерполяційного сплайна                                                                             | 217        |
| STEP   | Підпрограма реалізації одного кроку розв'язування одновимірного рівняння теплопровідності                                                          | 288        |

|               |                                                                                                                                                                                          |            |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>STEP1</b>  | Підпрограма реалізації одного кроку різницевої схеми з ваговими коефіцієнтами розв'язування одновимірного рівняння теплопровідності в декартових, циліндричних або сферичних координатах | <b>286</b> |
| <b>STIFF</b>  | Підпрограма реалізації одного кроку інтегрування задачі Коші для системи звичайних диференціальних рівнянь                                                                               | <b>246</b> |
| <b>SVD</b>    | Підпрограма обчислення сингулярного розкладання дійсної прямокутної матриці                                                                                                              | <b>227</b> |
| <b>SVISZ</b>  | Підпрограма знаходження максимального власного значення і відповідного власного вектора дійсної симетричної матриці                                                                      | <b>213</b> |
| <b>SZERO</b>  | Підпрограма обнулення глобальної матриці                                                                                                                                                 | <b>283</b> |
| <b>ZEROIN</b> | Підпрограма розв'язування нелінійного рівняння методом, що поєднує ділення навпіл, обернену квадратичну та лінійну інтерполяції                                                          | <b>201</b> |
| <b>ZKR1</b>   | Підпрограма розв'язування методом сіток крайової задачі для звичайного диференціального рівняння другого порядку                                                                         | <b>275</b> |

## СПИСОК ВИКОРИСТАНОЇ ТА РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. *Бабенко К. И.* Основы численного анализа.— М.: Наука, 1986.— 744 с.
2. *Бахвалов Н. С., Жидков Н. П., Кобельков Г. М.* Численные методы.— М.: Наука, 1987.— 598 с.
3. *Бахвалов Н. С.* Численные методы.— М.: Наука, 1973.— 623 с.
4. *Баяловский Ю. М., Михайлова Т. Н., Галактионов В. А.* ГРАФОР. Графическое расширение Фортрана.— М.: Наука, 1985.— 288 с.
5. *Волков Е. А.* Численные методы.— М.: Наука, 1987.— 256 с.
6. *Завьялов Ю. С., Квасов Б. И., Мирошниченко В. Л.* Методы сплайн-функции.— М.: Наука, 1980.— 352 с.
7. *Захаров А. Ю., Турчанинов В. И.* STISS-программа для решения жестких систем ЗДР: Инструкция / Ин-т прикл. математики им. М. В. Келдыша АН СССР.— М., 1977.— 28 с.
8. *Золотухин С. В.* Линейные формулы со второй производной в многопараметричном методе прогноз-коррекции для жестких систем.— М., 1985.— 21 с.— Деп. в ВИНТИ 06.12.85, № 8388—385.
9. *Золотухин С. В.* Способ поднятия эффективности метода Гира для жестких систем.— К., 1982.— 10 с.— Деп. в ВИНТИ 11.01.83, № 151—83.
10. *Калиткин Н. Н.* Численные методы.— М.: Наука, 1978.— 512 с.
11. *Коллатц Л., Альбрехт Ю.* Задачи по прикладной математике.— М.: Мир, 1978.— 168 с.
12. *Ляшко И. И., Макаров В. Л., Скоробогатько А. А.* Методы вычислений.— К.: Вища шк. Головное изд-во, 1977.— 406 с.
13. *Марчук Г. И., Агошков В. И.* Введение в проекционно-сеточные методы.— М.: Наука, 1981.— 416 с.
14. *Марчук Г. И.* Методы вычислительной математики.— М.: Наука, 1980.— 536 с.
15. *Михлин С. Г.* Численная реализация вариационных методов.— М.: Наука, 1966.— 432 с.
16. *Норри Д., Дефриз Ж.* Введение в метод конечных элементов.— М.: Мир, 1981.— 304 с.
17. *Плис А. И., Сливина Н. Н.* Лабораторный практикум по высшей математике.— М.: Высш. шк., 1983.— 208 с.
18. *Ректорис К.* Вариационные методы в математической физике и технике.— М.: Мир, 1985.— 590 с.
19. *Самарский А. А.* Введение в численные методы.— М.: Наука, 1982.— 272 с.
20. *Самарский А. А., Гулин А. В.* Численные методы.— М.: Наука, 1989.— 432 с.
21. *Самарский А. А., Николаев Е. С.* Методы решения сеточных уравнений.— М.: Наука, 1978.— 590 с.
22. *Самарский А. А.* Теория разностных схем.— М.: Наука, 1983.— 616 с.
23. *Сборник научных программ на ФОРТРАНе: Матричная алгебра и линейная алгебра.*— М.: Статистика, 1974.— Вып. 2.— 224 с.
24. *Сильвестер П., Феррари Р.* Метод конечных элементов для радионинженеров и инженеров-электриков.— М.: Мир, 1986.— 229 с.
25. *Уилкинсон Дж., Райни К.* Справочник алгоритмов на языке АЛГОЛ. Линейная алгебра.— М.: Машиностроение, 1976.— 329 с.

26. Форсайт Дж., Малькольм М., Моулер К. Машинные методы математических вычислений.— М. : Мир, 1980.— 280 с.
27. Gear C. W. Numerical initial value problems in ordinary differential equations.— New Jersey: Prentice Hall, 1, nc., Englewood Cl.— 1971.— 253 p.
28. Hindmarch A. C. LSODE and LSODE, two new initial value ODE solvers // ACM.— SIGNUM Newsletter.— 1980.— Vol. 15, N 4.— P. 10—11.
29. Hindmarch A. C. EPISODE: an experimental package for the integration of system of O.D.E // Tech Report. UCID—30112. Ren. LLL.— 1977.

- Вектори власні** 40  
 — — матриці 40  
 — вузлових сил елемента 163  
 — сингулярні ліві 95  
 — — праві 95  
**Вектор навантаження** 163  
 — — глобальний 164  
**Відхилення середньоквадратичне** много-  
 члена 93  
 — — функції 92  
**Вузли сітки** 6  
 — — внутрішні 159  
 — — граничні 159  
**Добуток скалярний** 144  
**Задача Діріхле** 158  
 — добре обумовлена 84  
 — інтерполювання многочленами 48  
 — кінематичних зв'язків 163  
 — Коші 101  
 — крайова 118  
 — мішана для двовимірного рівняння  
 теплопровідності 180  
 — погано обумовлена 80  
 — початково-крайова 187  
**Збіжність методу Ньютона** 35  
 — — простої ітерації 34  
 — рівномірна 170  
**Значення власні** 40  
**Інтерполювання обернене** 56  
**Коефіцієнти Кристофеля** 83  
**Корені рівняння** 7  
**Кратність вузла** 49  
**Масив вказівок** 59  
**Матриця Гільберта** 31  
 — жорсткості 162  
 — — глобальна 164  
 — маси елемента 163  
 — Нордсіка 113  
 — обернена 25  
 — трикутна верхня 26  
 — нижня 26  
**Метод Бубнова — Гальоркіна** 142  
 — виключення Гаусса 23  
 — — — з вибором головного елемента  
 в стовпчику 24  
 — дихотомії 15  
 — ділення навпіл 15  
 — дотичних 16  
 — інтегро-інтерполяційний 150, 170, 188  
 — ітерації єдиного власного значення  
 вектора 42  
 — квадратного кореня 26  
 — лінеаризації 16  
 — Ньютона 16, 35  
 — обертання Якобі 41  
 — прогонки 67, 152, 171, 182  
 — простої ітерації 15, 34  
 — Рітца 142  
 — Рунге 142  
 — Рунге — Кутти 100  
 — сингулярного розкладу (SVD) 94  
 — скінченних елементів 158  
 — — різниць 150  
 — стрільби 125  
 — — з відновленням розв'язку 126  
 — — — ортогоналізацією 127  
 — — повторним розв'язуванням задач  
 Коші 126  
**Методи економічні** 180  
**Многочлен інтерполяційний** 49  
 — — Ерміта 50  
 — — з кратними вузлами 50  
 — — у формі Лагранжа 49  
 — — — — Ньютона 49  
 — Лежандра 93  
 — матриці характеристичний 41  
 — найкращого середньоквадратичного  
 наближення 93  
 — узагальнений 49, 92  
 — Чебишова першого роду 94  
**Нерівність Лебега** 53  
**Норма вектора** 27  
 — матриці 27  
**Носій неперервної функції** 159  
**Обумовленість системи лінійних алгеб-  
 раїчних рівнянь** 27  
 — — нелінійних рівнянь 36  
**Оцінка числа обумовленості матриці** 27

Похибка інтерполювання 53  
— методу на кроці 102  
— — Рунге — Кутти 102  
Правило Рунге 85  
Проблема власних значень повна 41  
— — — часткова 42  
Простір гільбертовий 144  
Процес ортогоналізації Шмідта 128  
  
Радіус спектральний матриці Якобі 111  
Рівняння гіперболічного типу 187  
— теплопровідності 169  
Різниці розділені 50  
— — з кратними вузлами 51  
Розв'язок операторного рівняння узагальнений 143, 145  
Розклад матриці сингулярний 95  
  
Система звичайних диференціальних рівнянь жорстка 111  
— лінійних алгебраїчних рівнянь нормальна 93  
— — — — погано обумовлена 27  
— поліномна 145  
Сітка нерівномірна 150  
— рівномірна 7, 180  
Слайн-апроксимація 64  
— інтерполяційний 64  
— — кубічний 64  
— — параметричний 70  
— — природний 65  
Стала Лебега 54  
Схема Горнера 52  
— Пісмена — Рекфорда 181  
— різницева 170  
— — економічна 181  
— — з ваговими коефіцієнтами 180  
— — неявна 181  
— — — абсолютно стійка 181  
— — — повністю 170  
— — симетрична Кранка — Ніколсона 170

— — стійка 170  
— — — умовно 181  
— — явна 170, 181, 187  
Таблиця розділених різниць 50  
Триангуляція області 158  
Умови крайові роду другого 172  
— — — першого 172

Формула квадратурна 79  
— — Гаусса 82  
— — — ускладнена 84  
— — інтерполяційного типу 79  
— — Ньютона — Котеса 80  
— — парабол 80  
— — прямокутників 80  
— — — ускладнена 81  
— — Сімпсона 80  
— — — ускладнена 82  
— — трапецій 80  
— — — ускладнена 81  
Функції фінітні 158  
Функціонал 66, 68  
— енергії 142, 161

Частина головна похибки формули прямокутників 84  
Числа сингулярні матриці 95  
Число обумовленості 17  
— — задачі обчислення квадратурної суми 80  
— — кореня 17  
— — матриці 27  
— — спектральне системи звичайних диференціальних рівнянь 111  
Член залишковий інтерполяційного многочлена 53  
— — — — з кратними вузлами 53  
— — формули Гаусса 83  
— — — прямокутників ускладненої 81  
— — — Сімпсона 82  
— — — трапецій 82



## Навчальне видання

*Бурківська Валентина Лаврентіївна*  
*Войцехівський Сергій Олександрович*  
*Гаврилюк Іван Петрович*  
*Греков Леонід Дмитрович*  
*Клуннік Олексій Олексійович*  
*Копистира Микола Петрович*  
*Кузьмін Анатолій Володимирович*  
*Макаров Володимир Леонідович*  
*Москальков Михайло Миколайович*  
*Риженко Андрій Іванович*

## МЕТОДИ ОБЧИСЛЕНЬ Практикум на ЕОМ

Оправа і титул художника *В. Г. Самсонова*  
Художній редактор *С. В. Анненков*  
Технічний редактор *А. І. Омоховська*  
Коректор *Л. К. Дмитришина*

Здано до набору 28.02.95. Підписано до друку 26.10.95. Формат 60×84<sup>1</sup>/<sub>16</sub>. Папір друк. № 2. Гарнітура літературна. Високий друк. Умовн.-друк. арк. 17,67. Умовн. фарбовідб. 17,90. Обл.-вид. арк. 20,80. Вид. № 9855. Тираж 3500. Замовлення 5—555.

Видавництво «Вища школа», 252054, Київ-54,  
вул. Гоголівська, 7.

Головне підприємство республіканського виробничого об'єднання «Поліграфкнига», 252057, Київ-57,  
вул. Довженка, 3.