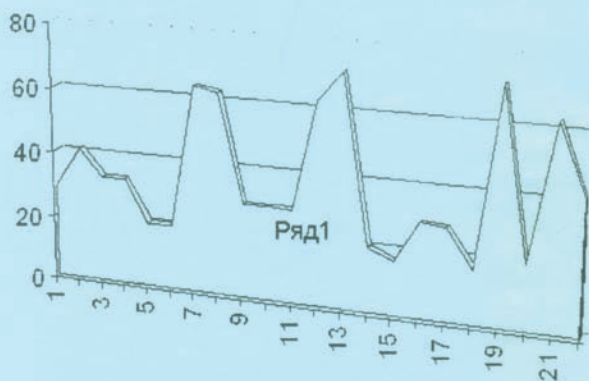


В.Ю.КОЦЮБИНСЬКИЙ

ПРИКЛАДНІ ПРОГРАМНІ СИСТЕМИ



Міністерство освіти та науки України
Вінницький державний технічний університет

В.Ю.Коцюбинський

ПРИКЛАДНІ ПРОГРАМНІ СИСТЕМИ

Затверджено Ученою радою Вінницького державного технічного університету як конспект лекцій для студентів бакалаврських напрямків 7.091401 – 01 - "Системи управління і автоматики", денної та заочної форм навчання. Протокол №13 від 4 липня 2002 р.

Вінниця ВДТУ 2003

Рецензенти:

Азаров О.Д., доктор технічних наук, професор

Поджаренко В.О., доктор технічних наук, професор

Хаїмзон І.Й., доктор технічних наук, професор

Рекомендовано до видання Ученою радою Вінницького державного технічного університету Міністерства освіти і науки України

Коцюбинський В.Ю.

К75 Прикладні програмні системи. Конспект лекцій. – Вінниця
ВДГУ, 2003. – 77 с.

В конспекті лекцій розглянуті основні положення, що стосуються сучасних пакетів прикладних програм під час вивчення теоретичних основ та виконання лабораторного практикуму. Конспект лекцій розроблений у відповідності з планом кафедри до дисципліни та призначений для студентів спеціальності 7.091401 „Системи управління і автоматики”, а також студентам суміжних спеціальностей.

ЗМІСТ

ВСТУП.....	5
1.Текстовий процесор „Word”.....	6
1.1.Форматування текстових документів.....	6
1.2.Дії з фрагментами тексту.....	8
1.3.Побудова таблиць і діаграм.....	9
1.4.Word. Дії з об'єктами.Об'єктиWordArt.....	10
1.5.Поля.Злиття документів.Макроси.....	12
Контрольні запитання.....	14
2.Електронна таблиця „Excel”.....	15
2.1.Структура ET Excel 97 і основні типи даних, що можуть бути використані в ній.....	15
2.2.Копіювання формул у таблиці.....	16
2.3.Загальні відомості про форматування листа.....	18
2.4.Швидке введення даних (автозаповнення).....	19
2.5.Редагування листа.....	21
2.6.Функції в Excel.....	23
2.7.Автоматичне форматування аркушів.....	24
2.8.Побудова графіків і діаграм в Excel 97.....	26
2.9.Організація структур розгалуження та ітерацій в Excel 97.....	27
Контрольні запитання.....	28
3. Система керування базами даних (СКБД) „Access”.....	29
3.1.Загальні відомості про бази даних. Типи даних, які зберігаються в них.....	29
3.2.Створення структури бази даних. Об'єкти, що використовуються в Access 97.....	30
3.3.Створення таблиці бази даних за допомогою конструктора таблиць.....	32
3.4.Робота з формами в Access 97.....	33
3.5.Робота з фільтрами і запитами.....	34
3.6.Побудова запитів у режимі конструктора.....	36
3.7.Реляційні таблиці, запити, форми.....	37
3.8.Оформлення звітів в Access 97.....	38
Контрольні запитання.....	39
4. Пакети прикладних програм для автоматизації математичних обчислень.....	41
4.1.Основи інтерфейсу користувача MathCAD 8.0.....	41
4.2.Вхідна мова системи MathCAD 8.0.....	45
4.3.Збереження і використання даних.....	47
4.4.Виконання арифметичних операцій.....	48
4.5. Обчислення та використання функцій.....	49
4.6.Операції з векторами і матрицями.....	50
4.7. Статистична обробка даних.....	52
4.8. Розв'язання нелінійних рівнянь і систем.....	53
4.9. Виконання прямого і оберненого перетворення Фур'є.....	54

4.10. Розв'язання диференціальних рівнянь.....	55
4.11. Програмування в середовищі MathCad.....	57
5. Математична система Maple V 4.....	60
5.1. Структура пакета Maple V 4 і основи роботи з ним.....	60
5.2. Вхідна мова програми Maple V 4.....	60
5.3. Типи об'єктів у Maple. Аналіз структури об'єктів.....	61
5.4. Функції в Maple.....	63
5.5. Оператори в Maple.....	63
5.6. Команди перетворення виразів, типів, оцінювання.....	64
5.7. Команди в спеціалізованих пакетах.....	65
5.8. Розв'язання рівнянь і нерівностей.....	66
5.9. Команди реалізації операцій математичного аналізу.....	67
5.10. Диференціювання та інтегрування в Maple.....	69
5.11. Команди для розв'язання задач лінійної алгебри.....	69
5.12. Основні матричні і векторні операції.....	70
5.13. Команди для розв'язання диференціальних рівнянь.....	71
5.14. Графіка в MAPLE.....	72
Контрольні запитання	75
Література.....	76

Поширення ЕОМ в сучасному світі відбувається настільки швидко, що комп'ютери давно вийшли за межі наукових лабораторій. Комп'ютер став настільним помічником вдома, а коло практичних задач, що можуть бути вирішені за допомогою сучасних комп'ютерів, також постійно розширюється.

Таке впровадження комп'ютера в наше життя супроводжується зростанням кількості програмного забезпечення, що дозволяє розширити коло користувачів ЕОМ від бухгалтера, що працює з такими обчислювальними додатками, як електронні таблиці, статистичних службовців, що працюють з сучасними системами керування баз даних, до студентів, чи наукових співробітників, які використовують сучасні потужні обчислювальні пакети для оптимізації математичних розрахунків.

Незважаючи на зростаюче коло таких прикладних програмних систем, їх сучасний розвиток відбувається шляхом удосконалення інтерфейсу користувача, покращення наочності та створення багатофункціональних довідкових систем. Все це дозволяє сучасним користувачам комп'ютера використовувати мінімум знань та навичок для початку ефективної праці за ЕОМ та швидко отримати бажані результати.

В конспекті лекцій розглянуті питання ефективного застосування різних пакетів прикладних програм, знання яких дозволяє створювати різноманітні сучасні документи різного функціонального призначення. Розглянуто програми, що входять до сучасного програмного комплексу Microsoft Office. Серед цих програм необхідно виділити текстовий редактор Word, програму для проведення економічних, статистичних та інших розрахунків Excel та систему керування базами даних Access. Інша частина конспекту присвячена розгляду найпотужніших сучасних обчислювальних математичних пакетів Maple V 4 та MathCad.

Матеріал, викладений в Конспекті лекцій, може бути використаний для вивчення курсу „Прикладні програмні системи”, інших курсів, що пов'язані з вивченням питань, викладених в цьому посібнику, а також буде корисний широкому колу фахівців, що займаються проведенням розрахунків у власних дослідженнях та оформленні документів у відповідності з сучасними стандартами.

1.ТЕКСТОВИЙ ПРОЦЕСОР “Word”

Текстовий процесор Word є найбільш відомим програмним додатком, що входить до складу сімейства програм Microsoft Office та створений для роботи в операційному середовищі Windows. Сучасні функціональні можливості текстового процесору Word вийшли за межі звичайного редагування та друку документів. На даний час Word представляє собою потужне програмне середовище, що дозволяє працювати одночасно з декількома документами, здійснювати обмін даними між додатками тощо. Особливо потрібно відзначити можливість застосування гіпертекстових посилань з метою інтеграції створених документів при роботі в мережі, використання засобів автоматизації введення створення нових документів, їх редагування та перевірки правопису.

Вікно програми Word має загальний схожий інтерфейс Windows-програм та складається з:

- рядка, в якому є назва програми і документа та кнопки керування вікном програми;
- рядка меню з кнопками керування вікном документа;
- панелей інструментів;
- робочого поля документа зі смугами прокручування і кнопками задання вигляду документа;
- рядка статусу.

Найважливіші дії, які можна виконати над текстовим документом за допомогою команд Файл або кнопок стандартної панелі інструментів є:

- створити новий документ (Створити, Ctrl+N);
- відкрити існуючий (Відкрити, Ctrl+O);
- зберегти на диску (Зберегти, Ctrl+S);
- зберегти з іншою назвою (Зберегти як, F12);
- закрити документ (Закрити);
- надрукувати (Друк, Ctrl+P);
- задати параметри сторінки (Параметри сторінки).

1.1. Форматування текстових документів

Форматування полягає у наданні документу вигляду, який є необхідним для користувача. Форматування здійснюють за допомогою команд головного меню або кнопок панелі форматування, що дозволяє задати такі параметри:

- стиль документа (комбінація різних форматів, що має своє ім'я і може бути застосована до виділеного фрагмента);
- шрифт (стиль символу визначає зовнішній вигляд і розміри);
- розмір символів у пунктах (28 пунктів = 1 см);

- стиль шрифту:
 - жирний, що задається комбінацією клавіш **Ctrl+B**;
 - курсив (**Ctrl+I**);
 - підкреслений (**Ctrl+U**);
- колір шрифту;
- вирівнювання:
 - по лівому краю (**Ctrl+L**);
 - по центру (**Ctrl+E**);
 - по правому краю (**Ctrl+R**);
 - двостороннє (**Ctrl+J**);
- нумерований список;
- маркований список;
- зсув фрагмента вправо;
- текст у рамці.

Перш ніж форматувати текст, необхідно виділити ту частину, яку необхідно змінювати:

- слово (подвійним клацанням миші на слові);
- рядок (клацання мишею зліва від рядка);
- абзац (потрійне клацання);
- весь документ (**Ctrl+A**).

До виділеного фрагмента можна застосовувати ефекти:

- перетворення шрифтів (формат → шрифти):
 - створення верхніх або нижніх індексів;
 - закреслювання і підкреслення;
 - контурів і тіней;
 - піднятий і утоплений;
 - використання схованих виділень;
- задання параметрів інтервалу і зсуву (формат → абзац):
 - відступи та інтервали;
 - розміщення на сторінці;
 - заборона висячих рядків;
 - заборона розриву абзацу;
 - заборона нумерації рядків і автоматичного переносу;
 - параметри табуляції;
- анімації:
 - феєрверк;
 - миготливий фон;
 - мурахи й ін.

Під стилем документа розуміють таку сукупність параметрів:

- стиль шрифту;
- розмір символів;
- вказівка на наявність зсуву;
- значення інтервалів;
- розміщення на сторінці;

- вид тексту.

Для зручності роботи з текстовим документом існують такі режими перегляду документу:

- звичайний - представляє найбільший екран і відображає форматування символів і абзаців;
- режим електронного документа - зручний при роботі з великим текстом, який містить велику кількість заголовків, праворуч – текст, ліворуч – заголовки;
- режим розмітки документа - документ виглядає так як після друку, у цьому режимі відображені всі об'єкти документа, і показані місця розташування розривів сторінок;
- режим структури - в цьому режимі можна побачити структуру документа в розгорнутому і згорнутому вигляді; створюється документ у згорнутому вигляді з використанням заголовків з бібліотеки вбудованих заголовків стилів, а потім вводиться зміст після кожного з заголовків та застосовується розгортання структури документа;
- режим попереднього перегляду;
- повноекранний режим перегляду ховає панель інструментів, рядок стану і рядок меню і звільнює місце для самого документа
- задання власного режиму перегляду (вид – масштаб).

Іншою зручною можливістю Word є створення нумерованих і маркованих списків.

Для створення списку вводиться така послідовність:

- TAB 1.
- TAB (елемент списку) ENTER, ENTER

Інший спосіб полягає в тому, що на панелі інструментів перед введенням нумерованого списку необхідно просто натиснути кнопку Нумерація.

Чудова здатність Word 97 полягає в автоматичній зміні номерів списку при видаленні з списку або вставки нових елементів.

1.2. Дії з фрагментами тексту

Елементами тексту є символ, слово, речення, абзац, які називаються фрагментами. Над фрагментом тексту визначені такі основні дії:

- копіювання в буфер обміну; виправлення - копіювання, Ctrl - C;
- вирізання в буфер обміну; виправлення - вирізати Ctrl - X;
- видалення з тексту, Del;
- вирівнювання;
- перетворення символів: збільшення, зменшення, задання стилю написання.

Крім фрагмента тексту в текст можна вставити рисунок, використовуючи панель інструментів **Рисуння**: лінія, прямокутник, еліпс.

Фігури можна переміщувати, зафарбовувати, розтягувати, стискати, змінювати товщину лінії, накладати одну фігуру на іншу, розміщати в них текст різного кольору, створювати об'ємні ефекти або ефекти затінення, повертати.

Ці дії виконують за допомогою кнопок панелі рисуння:

вибір об'єктів; повертати; автофігури; кнопки з зображенням геометричних фігур; кадр; WordArt; колір заливання та ін.

За допомогою меню готових фігур (Автофігури) можна будувати блок схеми. Видалення невдалих фігур здійснюється Del після попереднього виділення.

1.3. Побудова таблиць і діаграм

Таблиці призначені для наочного представлення інформації. Елементами таблиці є:

- комірки;
- рядки;
- стовпці;
- рамки і дані, що знаходяться в клітинках.

Існує три способи створення таблиць за допомогою пункту **Таблиця** основного меню:

- вставити таблицю;
- нарисувати таблицю;
- перетворити текст у таблицю.

Всі ці дії виконуються за допомогою команд рядка меню і представлені нижче:

1. Таблиця → вставити Таблицю → кількість стовпців і рядків → дані.
2. Таблиця → нарисувати Таблицю → П.І. таблиці і границі → курсор набуде зображення олівця.
3. Ввести текст із роздільниками → перетворити текст у таблицю.

Над таблицями і основними елементами допустимі такі дії:

1. Перед введенням інформації необхідна комірка виділяється за допомогою одиночного клацання миші;
2. Для переміщення між комірками використовуються комбінації клавіш TAB, (Shift-TAB)
3. Розміри комірок можуть бути змінені перетягуванням обмежувальних ліній комірок за допомогою миші;
4. Клітинки можна з'єднувати, попередньо виділивши;
5. Дані в клітинках можна форматувати у межах клітинки аналогічно.

Вигляд таблиці можна змінити засобами команди **Границі і заливання** пункту **Формат**: закладки границі, сторінка і заливання дозволяють:

- вибрати тип рамки, тип, ширину і колір ліній, забрати окремі лінії;
- сторінку з таблицею взяти в рамку - рисунок;
- клітинки затінити, залити кольором або прикрасити.

Діаграми є наочним способом подання числової інформації. Для того, щоб на основі даних таблиці створити діаграму необхідно виконати такі дії:

1. Таблицю потрібно виділити;
2. Вибрати такі команди рядка меню, як **Вставка** → **Рисунок** → **Діаграма**;
3. Ввести у випадку необхідності дані;
4. За допомогою кнопки **інші види діаграм** вибрати іншу;
5. Закрити вікно таблиці і клацнути в необхідному місці документа;
6. Активізовану діаграму можна перетягувати або змінювати розміри.

1.4. Word. Дії з об'єктами. Об'єкти WordArt

Крім звичайного тексту, рисунків, автофігур, таблиць і діаграм текстовий документ може містити в собі й інші об'єкти: кадри, картинки, фотографії, гіперпосилання, текстові ефекти, графічні і мультимедійні ефекти.

Розглянемо детальніше поняття об'єкта. Об'єкт займає один рядок у документі. Якщо він не широкий, то поруч варто розміщати інший об'єкт чи вводити текст. В такому випадку текст повинен обтікати об'єкт як це робиться в журналах, книгах і т.д.

Створені в документі різні тексти і таблиці є елементами документа, а не об'єктами. Переміщати їх можна тільки через буфер обміну. Для того, щоб елемент текстового документа став об'єктом, його необхідно помістити в кадр. Кадр - це прямокутник (об'єкт), в який розміщують різні елементи документа й інші об'єкти: таблиці, діаграми, окремі слова, рисунки, фотографії. Таблиця стає об'єктом тільки після того, як вона вставиться в кадр. Кадр має рамку. Її можна зробити кольоровою або невидимою.

Таким чином об'єкт - це автономний елемент документа. Його можна переміщати, змінювати розміри (мишею), обтікати текстом. Існує кілька стилів обтікання об'єкта текстом. Їх вибирають з меню: **Формат** → **Конкретний об'єкт (Кадр)** → **Закладка** (обтікання).

Задаються такі стилі: навколо рамки, по контуру, наскрізне, немає, зверху і знизу, а також стилі розташування тексту навколо об'єкта: навколо, ліворуч, праворуч, по більшій стороні.

За допомогою інших закладок задають кольори ліній, заливання робочого поля.

Після вставки об'єкта текстовий редактор автоматично переходить у режим розмітки сторінки. У звичайному режимі об'єкти будуть невидимими.

Командою **Вставка** в документ можна також вставляти:

- номери сторінок;
- інший документ із деякого файлу (файл);
- поточну дату і час;
- примітки;
- закладки;
- художні букви (символ), символи і спеціальні символи (автозаміна і клавіші);
- інші об'єкти доступні для процесора Word у цій операційній системі;
- гіперпосилання.

Розглянемо приклад створення об'єкта, вставки його в документ на прикладі об'єкта, який містить картинку зображення на екрані:

- натискання клавіші Print Screen приводить до запису вигляду екрана в буфер обміну;
- запускаємо графічний редактор Paint;
- копіюємо вміст буфера в графічний документ;
- допрацьовуємо зображення і зберігаємо його у файлі з розширенням bmp.
- повертаємося до текстового документа і вставляємо графічний файл у заздалегідь створений кадр.

Виготовлення текстових спецефектів аналогічних тим, які зустрічаються в журналах, газетах, книгах, можливо за допомогою редактора текстових спецефектів WordArt.

Редактор спецефектів WordArt працює в двох режимах:

- Створення ефекта вручну засобами меню і панелі інструментів WordArt;
 - Використання еталонних зразків ефектів.
- Вручну за допомогою меню чи кнопок панелі інструментів можна підібрати наступні параметри:
- шрифт, його розмір, стиль написання;
 - тип ліній;
 - спосіб затінення, кольору букв і візерунків;
 - тип тіней;
 - повороти і форму накреслення тексту.

Іншим поширеним об'єктом є гіперпосилання як спосіб для встановлення зв'язку між документами. Якщо у файл вставити гіперпосилання на інші файли, то можна мати доступ відразу до двох чи більше файлів, не вставляючи їх один в один.

Гіперпосилання – це виокремлений іншим кольором об'єкт (підкреслений текст чи картинка), що містить адресу іншого файлу. Клацнувши раз на гіперпосиланні можна відкрити необхідний файл. Для створення гіперпосилання необхідно:

а) ввести або вибрати шлях до документа, на який буде вказувати посилання. У цій якості може виступати адреса Інтернету чи документ на мережному або іншому диску;

б) вказати ім'я об'єкта в документі, щоб перейти до конкретного об'єкта в документі (закладці, об'єкта бази даних чи слайду);

в) встановити підтвердження використання для гіперпосилання відносного шляху.

1.5. Поля. Злиття документів. Макроси

Поля призначені для вставки в основний документ текстів, рисунків або інших об'єктів, які будуть обновлятися відповідно до конкретних правил, а також для організації обчислень.

Поля використовують для підготовки типової документації із змінною адресацією, наприклад, ділових листів багатьом адресатам, заадресовування конвертів.

Розрізняють звичайні поля редактора Word і поля злиття. Звичайні поля вставляються командою **Вставити** → **Поле**.

Кожне поле характеризується кодом і значенням.

Код – це деяка послідовність символів (команда). Код і значення поля переключуються комбінацією клавіш Shift+F9.

Прикладами звичайних полів є поля для вставки поточної дати, нумерації сторінок документа, даних про автора документа, формул для обчислень, вибору альтернативних фрагментів тексту. Значення цих полів обновляються натисканням клавіші F9.

Поле як і всі об'єкти, що застосовуються в Word 97, має своє контекстне меню, яке використовується для:

- поновлення змісту полів;
- вибору відображення в документі кодів полів або їхнього значення;
- дій з форматування змісту полів.

Крім звичайних полів, розрізняють поля злиття, які застосовуються для створення копій типових документів методом злиття документів: основного і джерела даних.

У цьому випадку основний документ містить незмінну частину текст і поля злиття для даних, які будуть змінюватися. Дані будуть вставлятися в основний документ із джерела даних.

Коди полів злиття – назви полів, які дає користувач, наприклад, поштовий індекс, місто, вулиця.

Кількість полів і їх коди необхідно визначити для кожної задачі заздалегідь згідно з умовою задачі. Поля злиття користувач вставляє спеціальною командою, а їх назви система автоматично виділяє подвійними фігурними кутовими дужками.

Щоб у документі були відображені значення полів, а не коди, необхідно зняти режим відображення кодів полів: сервіс → параметри → вид → коди полів.

Розглянемо структуру джерела даних – файл, який містить дані про акціонерів (роздільником є пропуск) якщо не таблиця.

Прізвище	Ім'я	Індекс	Місто	Вулиця	Сума	Закінчення
Іванов	Сергій	21000	Вінниця	Хм. Шосе, 5	50	-ий
Петрова	Ірина	21000	Вінниця	Пирогова, 111	100	-а

Джерелом даних може бути текстовий файл, де дані набрані введеному вище вигляді (придатному для перетворення в таблицю), а також деяка таблиця даних, файл бази даних.

Ця задача розв'язується із застосуванням вікна Злиття (меню сервіс) і панелі інструментів Злиття.

1. Призначають створений чи поточний документ документом злиття на бланку і створюють або задають для нього джерело даних.

Сервіс → Злиття → Створити → Документи на бланку → активне вікно.

Сервіс → Злиття → Одержати дані → Створити джерело даних.

Видалити з запропонованого списку непотрібні або всі поля.

Додати свої назви, змінити порядок назв у списку полів (якщо це необхідно) → Ок.

Ввести назви файлу для джерела даних Зберегти → редагувати джерело даних → Ввести необхідні дані → Ок і повертаються в основний документ.

2. Створюють основний документ - вводять основний текст, вставляють поля злиття, використовуючи панель інструментів Злиття або команду Злиття.

3. Виконують злиття основного документа з джерелом даних у новий документ (або злиття на принтер) таким чином:

Сервіс → Злиття → З'єднати → Призначення → Новий документ → Об'єднати → зберегти результат - форму з назвою `формал.doc` або відправити на друк.

Макрос – це записана в пам'ять редактора послідовність натискань на клавіші і клацань мишею. Макросу ставлять у відповідність

(призначають) комбінацію клавіш (або кнопку панелі). Натискання на ці клавіші призводить до автоматичного виконання макросу.

Макроси призначені для автоматизації роботи користувача, якщо деякі дії приходиться часто повторювати. Макроси необхідно спочатку записати, а потім можна використовувати.

Сервіс → Макрос → Почати запис: Дають ім'я макросу і призначають макрос клавішам або кнопкам панелі інструментів.

У діалоговому вікні Настроювання клацають у рядку Нова комбінація клавіш і натискають на цю комбінацію клавіш → Призначити → Закрити – почався запис макросу.

Тепер усі дії користувача (крім переміщення курсору миші) будуть записані. На екрані буде маленьке віконце Зупинити запис із кнопками Зупинити (ліворуч) і Пауза (праворуч). Натискають на кнопку Зупинити і закривають вікно. На цьому створення макросу закінчене. Для виконання макросу натискають відповідну комбінацію клавіш.

Контрольні запитання

- 1 З яких елементів складається вікно програми “Word”?
- 2 Які режими функціонування програми задає користувач?
- 3 Як увімкнути панелі інструментів?
- 4 Як можна створити, зберегти та надрукувати документ?
- 5 Як задати параметри сторінки та розмір шрифту?
- 6 Що таке форматування тексту?
- 7 Які функції кнопок на панелі інструментів?
- 8 Які є види форматування фрагменту тексту?
- 9 Які функції панелі інструментів форматування?
- 10 Які функції буфера обміну?
- 11 Як побудувати геометричну фігуру та змінити її розміри?
- 12 Як створити автотекст?
- 13 Як створити у документі математичну формулу?
- 14 Як переглянути розташування тексту на сторінках?
- 15 Які об'єкти можливо вставляти в текст документа?
- 16 Які є способи створення таблиці?
- 17 Як можна змінити розмір таблиці, кількість рядків або стовпців у ній?
- 18 Які є типи діаграм?
- 19 Що таке об'єкт “WordArt”?
- 20 Яка функція гіперпосилання?
- 21 Які є види полів?
- 22 Які функції макросів?

2. ЕЛЕКТРОННА ТАБЛИЦЯ „Excel”

2.1. Структура ET Excel 97 і основні типи даних, що можуть бути використані в ній

Електронна таблиця – програма, призначена для обробки даних бухгалтерського, економічного або статистичного типу, а також для автоматизації математичних обчислень.

Робота з ET Excel 97 - це створення, редагування і використання книг. Книга складається з одного чи декількох аркушів (електронні або просто таблиці). Лист складається з рядків і стовпців, у яких зберігаються дані.

Користувач може відкрити одночасно кілька книг і переміщатися між ними Ctrl + F6.

Нові книги в Excel 97 складаються за замовчуванням із трьох аркушів (Лист1, Лист2, Лист3). Клацнувши на ярлику Листа можна перейти до роботи з цим листом, що буде відображений на екрані.

Кожен стовпець Листа має заголовок, що позначається буквами А, В, АА, IV. Кожен рядок має заголовок, що позначається числами 1, 2.....65536. Перетин стовпців і рядків, утворює комірку, що має унікальну адресу (в межах листа А1, У3) інші аркуші Лист! А1.

Хоча Excel 97 призначена насамперед для роботи з числовими даними, у її комірках можуть міститися дані таких типів:

- тексти – текстові значення (прізвища, адреси);
- числа – числові значення;
- формули – вирази для проведення обчислень над числами (хоча деякі формули можуть працювати і з текстовими значеннями);
- спеціальні формати даних (значення дати, часу).

Крім того Excel 97 може оперувати з даними з інших програм Office 97, а також імпортувати дані з інших програм електронних таблиць, наприклад Lotus 1-2-3.

Для того щоб ввести в комірку дані чи виконати над нею якісь дії, її необхідно виділити - зробити активною (застосовується спеціальне виділення у вигляді рамки: комірки, заголовків).

Для переміщення в програмі Excel за допомогою клавіатури застосовуються такі комбінації:

- клавіші зі стрілками – у напрямку на одну комірку;
- Ctrl↑, Ctrl↓ - перехід до самого нижнього чи самого верхнього рядка даних у листі;
- Ctrl←, Ctrl→ - перехід до самого лівого чи самого правого стовпця даних у листі;

- Page UP, Page Down – попередній чи наступний екран листа;
- Ctrl + Home – верхній лівий куток листа (комірка A1);
- End, клавіша зі стрілкою – остання порожня комірка у напрямку стрілки;
- Ctrl+Page UP, Ctrl+Page Down – перехід у робочу область іншого листа.

Коли три аркуші у книзі не задовольняють користувача, можна ввести максимальну кількість аркушів, що включаються в книгу **Сервіс – Параметри – Загальне – Аркушів у новій книзі**.

Часто в практичних задачах користувач обмежується одним листом у книзі.

Правий щиглик на ярличку листа, що видаляється, (видалити).

Для додавання листа (правий щиголь Додати). Додати можна також командою **Лист з меню Вставка**.

Перейдемо до введення даних

Над таблицею є рядок для введення даних (називається рядок формул), у якій відображаються дані, що вводяться. Під час введення дані можна редагувати. Щоб дане потрапило в пам'ять, необхідно натиснути клавішу Enter, клавішу переміщення чи курсору ∇ у рядку формули. Скасування введення в комірку - ESC чи Скасування X.

Якщо у виділеній комірці є дане яке необхідно відредагувати, то це робиться так:

- 1) подвійний щиглик мишею.
- 2) клавіша F2.
- 3) очищають комірку командами з меню:

Виправити → Очистити → Усе – вводиться нове дане.

Введений в комірку текст (до 255 символів) автоматично вирівнюється уздовж лівого краю, а числа уздовж правого.

Текстові дані використовуються для оформлення назв таблиць і назв рядків і стовпців даних. Числа в комірку вводяться звичайним чином, однак на екрані вони можуть бути відображені незвичайно:

- Число може бути округлено;
- Із символом грошової одиниці (\$, грн.);
- 3 комами, що відокремлюють тріади цифр.

2.2. Копіювання формул у таблиці

Відображення даного залежить від формату його представлення. Виділяти можна не тільки одну, але і кілька комірок. Формати чисел в обраних комірках задаються командою:

Формат → Комірка → Закладка (Число) → Формат числовий.

Іншим типом даних, що використовується у Excel є формули. Формули призначені для виконання дій над вмістом комірок (над даними) згідно з умовою конкретної задачі. Формули мають такий вигляд: =B1*C3. Після введення формули в комірку негайно відображається результат, а формула показана лише в рядку формул.

Щоб побачити усі формули в таблиці, необхідно задати режим відображення формул в комірках: Сервіс → Параметри → Закладка → Вид → Формула → ОК.

Щоб знову побачити результати обчислень необхідно зняти режим відображення формул.

При написанні формул можна комбінувати кілька стандартних арифметичних операторів з дотриманням природного пріоритету їхнього виконання.

Якщо замість результатів в комірці відображене #####, то це означає, що велике число в комірці не вміщується, тому стовпець необхідно зробити ширшим.

Обчислення в таблиці здійснюється зліва-направо і зверху-вниз (але цей порядок можна змінити автоматично.) Це означає, що зміна будь-якого вхідного даного відразу призводить до перерахунку всієї таблиці (якщо задане Автоматично на закладці обчислення вікна Параметри).

Режим Автоматично можна виключити і скористатися ручним керуванням за допомогою клавіш.

Адреси комірок виду B3 і C3 називаються відносними. Відносна адреса - це адреса, що посилається на комірку, грунтуючись на її поточному місці розташування.

У ET Excel 97 є можливість копіювати однотипні формули (а не вводити їх у кожену комірку окремо), що прискорює процес розв'язання задачі.

Під час копіювання формул відбуваються такі дві дії:

- Формула вводиться в інші комірки;
- Формула автоматично модифікується тобто змінюються відносні адреси, на які є посилання у формулі.

Наприклад:

Нехай в комірці A1 міститься формула =A2+A3. Якщо скопіювати формулу в комірку B5 то в ній буде зберігатися формула: =B6+B7. Крім відносних адрес в Excel можуть використовуватися абсолютна адресація і частково абсолютна адресація.

Абсолютна адреса - це адреса, що не змінюється при копіюванні формул. (приклад \$B\$5).

Частково абсолютна адреса - адреса в якій змінним може бути або адреса рядка або стовпця (\$B5)

Копіювання формул і автоматичне перерахування - це два основних способи автоматизації обчислень у ET Excel.

2.3. Загальні відомості про форматування листа

Для надання аркушам Excel професійного вигляду застосовують такі прийоми форматування:

1. Вирівнювання

Як уже згадувалося ET за замовчуванням вирівнює числа по правому краю, а текстові дані по лівому. При бажанні виділена комірка чи діапазон комірок можуть бути відформатовані за допомогою кнопок панелі інструментів:

- по лівому краю;
- по центру;
- по правому краю.

2. Зміна висоти рядків і ширини стовпців: Ширина стовпців і висота рядків змінюється шляхом перетягування обмежувальних ліній. (подібно до Word).

Якщо необхідно змінити ширину стовпця таким чином, щоб вона відповідала найбільшому значенню даних у стовпці, то необхідно виділити стовпець і двічі клацнути мишею біля правої границі назви цього стовпця.

3. Зміна шрифту

Для даних листа Excel використовує при форматуванні за замовчуванням формат загальний. За бажанням користувача можна встановити такі типи форматів:

ЗАГАЛЬНИЙ	Числові дані не мають спеціального форматування і відображаються точно в такому вигляді, у якому вводяться
ЧИСЛОВИЙ	Можливе вказання числа десяткових знаків, що Excel буде відобразити у всіх числових значеннях
ГРОШОВИЙ	Відображає знак валюти, відводячи під нього два розряди
ФІНАНСОВИЙ	Вирівнює розташовані в стовпці грошові величини по роздільнику цілої і дробової частини
ДАТА	Відображає дату і час як значення, формати яких можна змінити
ЧАС	Відображає тільки час зі значення дати і часу
ПРОЦЕНТНИЙ	Поділяє значення комірки на 100 і виводить на екран із символом %
ДРОБОВИЙ	Виводить числа у вигляді тремтінь
ЕКСПОНЕНТНИЙ	Використовує експонентне представлення числових значень
ТЕКСТОВИЙ	Форматує усі дані як текст

ДОДАТКОВИЙ	Форматує поштові індекси, індекси, телефонні номери і табельні номери
УСІ ФОРМАТИ	Дозволяє задати власний формат комірки, можна самостійно установити виводити «+» чи «-», і можна змінювати число знаків після коми

2.4. Швидке введення даних (автозаповнення)

Якщо користувач вводить дані, які мають деяку закономірність, то Excel використовує заповнення даними, що дозволяє копіювати дані чи робити збільшення значень, введених в одну комірку, записуючи їх у кілька суміжних комірок. Нехай потрібно створити перспективну послідовність, що містить рядок Рік 5 разів, а наступний рядок буде містити коди 1997-2001.

1. У крайню ліву комірку ввести слово Рік. Після цього не натискати ні клавішу Enter, ні клавіші зі стрілками.
2. Знайти маркер заповнення комірки. Маркер заповнення – маленький чорний квадрат, розташований у нижньому правому кутку активної комірки.
3. Перетягнути маркер заповнення вправо через чотири наступних стовпці. При перетяганні маркера заповнення Excel 97 відобразить підказку Рік, що впливає, показуючи значення нових комірок.
4. Відпустити кнопку миші. Excel 97 заповнить усі п'ять комірок вставивши в них написи.

Якщо перетягнути маркер заповнення вниз, то Excel 97 скопіює написи вниз. Аналогічним чином можна заповнити і прямокутну область. Ті ж дії можна виконати за допомогою команди **Виправлення/Заповнити**.

Крім звичайного заповнення приведеного вище Excel 97 дозволяє виконувати більш складну процедуру – **автозаповнення**. Автозаповнення – здатність ET перевіряти і робити збільшення введених даних відповідно до визначених закономірностей.

Повернемося до вище розглянутої задачі:

- 1) Введемо рік першим заголовком Рік число 1997, а під другим заголовком Рік число 1998.
- 2) Виділимо комірку з числами, використовуючи техніку перетягування, а потім перемістимо маркер заповнення на комірки, що залишилися.

Якщо необхідно використовувати автозаповнення для збільшення значень комірок на одну одиницю, то можна не виділяти перші дві комірки. Якщо виділити будь-яку комірку з числом і натиснути Ctrl і перетягнути маркер заповнення, Excel 97 додасть по одиниці при перенесенні даних у кожен наступну комірку, через яку перетягується маркер.

Оскільки Excel 97 може працювати не тільки з числовими, але і з текстовими значеннями, то засіб автозаповнення може помітити

закономірності, що найбільш часто використовуються для текстових значень, до яких відносяться:

- Назви днів тижня;
- Скорочені назви днів тижня (наприклад пн., вт., і т.д.);
- Назви місяців;
- Скорочені назви місяців (наприклад січн., гр. і т.д.);

Автозаповнення

Крім назв (скорочених) місяців і днів тижня Excel 97 може заповнювати будь-який список значень. Користувач може “навчити” Excel 97 складати власні списки автозаповнення. Після того як ви вкажете в Excel 97 новий список, щоразу при введенні першого значення і перетягування в будь-якому напрямку маркера заповнення комірки, у якій введено те значення, Excel 97 автоматично заповнить інші комірки елементами списку.

Для того щоб додати до наявних списків автозаповнення власний список, необхідно:

1. Вибрати команду Сервіс/Параметри.
2. Клацнути вкладку Списки, щоб відобразити наявні в Excel списки автозаміни.
3. Клацнути кнопку Додати.
4. Ввести свій список у поле Елементи Списку і натиснути ОК.

Такий список можна створити з даних, що знаходяться в суміжних комірках робочого листа. Для цього:

1. Виділяється список за допомогою перетягування миші.
2. **Сервіс/Параметри/Списки.**
3. Клацнути кнопку Імпорт для додавання виділеного діапазону до списків.

Автозаповнення звичайно зручно застосовувати у випадку використання набору типових заголовків і рядка з кількістю елементів не більше дванадцяти. Для створення ряду з великою кількістю значень, необхідно виконати такі дії:

1. Введіть в першу комірку перше значення рядка.
2. Введіть дані в цю комірку і у всі комірки, в яких будуть введені інші значення рядка.
3. Виділіть команду **Виправлення/Заповнити/Прогресія**, щоб відобразити на екрані діалогове вікно **Прогресія**.
4. Встановіть перемикач у положення по рядках (якщо виділені комірки в рядку) чи в положення по стовпцях.
5. Виберіть тип створюваного рядка (прогресія)

	Введіть	Excel відобразить
• Арифметична	-50.0, 50.100	150, 200, 250
• Геометрична	2, 4	8, 16, 32, 64

- Дати 1-січн., 1-квіт., 1-черв, 1-вер.,
- Автозаповнення Рік '96 Рік '97, Рік '98

Додавання приміток

Як і в Word 97, ви можете вставити примітку в комірку ET Excel 97. Примітка - це замітка, що відноситься до комірки, але не відображається як частина даних, що зберігаються в цій комірці. Примітка не відображається в комірці, Excel 97 тільки показує, що в комірці є примітка, додаючи для цього у верхній правий куток комірки червоний трикутник. Вміст примітки можна побачити тільки встановивши покажчик миші на таку комірку.

Щоб додати до виділеної комірки примітку, виберіть: **Вставка/Примітка**. Excel відобразить прямокутну область в яку можна ввести примітку.

Примітку можна також додавати до групи комірок.

2.5. Редагування листа

Для успішної і швидкої роботи в ET необхідно опанувати прийоми редагування. Розглянемо основні інструменти редагування Excel 97.

Перевірка орфографії

До Excel ні в одній програмі ET не було засобу перевірки орфографії. Хоча аркуші ET призначені, насамперед, для розміщення, форматування, і обчислення. Однак рідко зустрічаються таблиці без заголовків. Тому засіб перевірки орфографії є дуже корисним.

Нижче перераховані способи проведення перевірки орфографії в аркушах:

- Клацніть мишею на панелі інструментів кнопки **орфографія**;
- **Меню Сервіс/орфографія**;
- Клавіша F7, що є швидкою клавішею перевірки орфографії.

На відміну від Word 97 в Excel відсутні засоби перевірки граматики.

Засіб перевірки орфографії в Excel перевіряє тільки поточний лист. Якщо Excel знаходить помилку то її можна пропустити, виправити або додати в словник.

Автозаміна

Засіб автозаміни використовується при введенні елементів Excel так само як і однойменний засіб у Word 97. Взагалі Word 97, Excel 97 і інші продукти, що входять до складу Office 97, використовують той самий словник для автозаміни і перевірки правопису. Тому, якщо здійснюються зміни в словниках Word і Excel, інші продукти розпізнають ці зміни.

Для додавання в список елементів автозаміни:

1. Виберіть команду **Сервіс/Автозаміна**
2. У поле **замінити** введіть позначення для елемента автозаміни.

3. Натисніть TAB і в поле введіть текст, на який буде замінюватися елемент списку автозаміни.

4. OK, щоб повернутися в область редагування листа.

Що стосується команд, які працюють із фрагментами листа, такими як копіювання (Ctrl+C), вставка (Ctrl+V), вирізання (Ctrl+X) то їхня дія аналогічна дії відповідних команд Word, що не стосується команди **видалення**. Від видалених даних можуть сильно залежати інші дані на листі, тому перш ніж вдатися до видалення необхідно вибрати спосіб очищення, який ви хочете застосувати.

- Очищення виділених комірок і відправлення їх вмісту в буфер обміну (звична дія);
- Очищення виділених комірок з окремим видаленням форматів, приміток і вмісту;
- Повне очищення вмісту виділених комірок, усіх форматів і приміток;
- Видалення виділених комірок і їхніх позицій, у результаті чого інші комірки в рядку змістяться вліво, або ж комірки, розташовані нижче, змістяться вгору.

В комірках міститься набагато більше, ніж просто числа і текст, що зображений на екрані. В комірках можуть міститися не тільки формули і примітки, у них часто знаходяться посилання на інші комірки.

Якщо ви хочете видалити дані виділених комірок, то натисніть клавішу Del. Excel 97 збереже форматування і примітки, що використовувалися до видалення даних.

Для того, щоб очистити комірку вибірково, виберіть команду **Виправлення/Очистити** і клацніть на одному з варіантів очищення:

- усе - видаляє з виділених комірок усе: формати, вміст, примітки.
- формати - видаляє тільки формат виділених комірок, скасовується спеціальне форматування без зміни вмісту комірок.
- вміст - видаляє вміст комірок, не має форматування і примітки.
- Примітка - видаляє примітки, що відносяться до виділених комірок.

Щоб відновити те, що ви видалили помилково, потрібно використати команду **Виправлення /Скасувати** або Ctrl+Z.

Більш швидко очистити вміст комірок можна в такий спосіб: необхідно виділити комірку і перетягнути маркер заповнення вгору і вліво. Якщо після перетягування маркера відпустити кнопки, то Excel виділить сірим кольором комірки, що будуть стерті.

Щоб разом з вмістом видалити і самі виділені комірки, а потім заповнити простір, що звільнився при цьому, виберіть **Правка/Видалити**, щоб вивести діалогове вікно **Видалення комірок**. Встановіть перемикач у положення комірок зі зміщенням вліво, або комірку зі зміщенням вгору, щоб вказати Excel спосіб видалення.

Крім видалення комірок існує можливість вставки комірок:

1. Виділити комірку;
2. Вставка/комірки;

Встановити перемикач у положення комірки зі зміщенням вправо, або в положення зі зміщенням вниз.

Аналогічним чином можливе здійснення процедур вставки і видалення цілих рядків і стовпців: виправлення/вставка видалити/рядок стовпець

Діапазони комірок листа

Під діапазоном розуміється група комірок, що утворює прямокутник. Діапазоном може бути одна комірка, рядок, стовпець чи кілька суміжних рядків і стовпців. Діапазон завжди складається із суміжних комірок. Описати діапазон можна, вказавши адреси його лівої верхньої комірки (показчика прив'язки) і правої нижньої комірки. У такий спосіб діапазон, що починається з комірки B3 і що закінчується F4 записується B3 : F4.

Однак якщо дати діапазону ім'я, то процес звертання до нього можна сильно полегшити, крім того діапазони, що мають імена легше запам'ятовуються користувачем.

Щоб дати діапазону ім'я виконайте такі дії:

1. Виділіть комірку, яку хочете включити в діапазон з ім'ям;
2. Клацніть поле Ім'я, розташоване ліворуч від рядка формул (це поле введення тексту, у якому відображаються адреси комірок);
3. Введіть ім'я діапазону і натисніть Enter.

У випадку, коли буде необхідно перемістити чи скопіювати один з цих діапазонів чи використати його у формулі, просто вкажіть ім'я діапазону, а Excel 97 розпізнає його сам.

2.6. Функції Excel 97

Для спрощення виконання ряду обчислень у ET Excel 97 включена велика кількість вбудованих функцій. Вбудовані функції мають зарезервовані імена, що завжди закінчуються круглими дужками (СУММ (...)). Аргументами різних функцій можуть бути як числа так і текстові значення. Для функції СУММ аргументами можуть бути перелік комірок, числа, адреса діапазону або його ім'я.

При роботі дуже часто виникає потреба в підсумовуванні даних. Excel може запропонувати застосування функції СУММ() автоматично, щоб скористатися цією можливістю потрібно виконати такі дії:

1. Виділити діапазон, значення якого потрібно додати;
2. На панелі інструментів клацнути кнопку **Автоподаток**, і Excel "здогадається", що функція СУММ застосовується до виділеного

- діапазону і вставити функцію СУММ() у вільну комірку праворуч або знизу від останньої комірки діапазону;
3. При необхідності можна ввести зміни у виділеному діапазоні, усі зміни відібноються на значенні функції;
 4. При необхідності скопіювати комірку з функцією в сусідні комірки.

Перелік функцій, що найчастіше зустрічаються в Excel A

ABS() Обчислює абсолютну величину свого аргументу (зручно застосовувати для обчислення різниці зростання і відстаней);

ДИСП() Обчислює дисперсію по вибірці, вибіркою є аргументи зі списку;

КОРЕНЬ() Обчислює квадратний корінь аргументу;

МАКС() Повертає найбільше значення зі списку аргументів;

МИН() Повертає найменше значення зі списку аргументів;

ПИ () Обчислює значення константи (аргументи вводити не вимагає);

ПРОИЗВЕД () Обчислює добуток значень;

РИМСКОЕ () Перетворить значення комірки в число, записане римськими цифрами;

СРЗНАЧ () Обчислює середнє арифметичне своїх аргументів;

СЕГОДНЯ () Повернення поточної дати (аргумент вводити не потрібно);

СТАНДОТКЛОН () Обчислює стандартне відхилення списку аргументів;

СУММ () Обчислює суму аргументів;

СЧЁТ () Підраховує кількість числових аргументів у списку;

СЧЕТЗ () Підраховує загальну кількість аргументів у списку;

СЧИТАТЬПУСТОТЫ () Підраховує кількість порожніх комірок (якщо вони є) в інтервалах, заданих аргументами.

2.7. Автоматичне форматування аркушів

Крім засобів форматування аркушів, розглянутих раніше, Excel 97 дозволяє використовувати засоби автоматичного форматування, що застосовуються вже до створеного листа для надання йому професійного вигляду:

Для використання автоформату, необхідно виділити весь лист і вибрати команду **Формат-автоформат**. Після чого буде запропонований список форматів, які можна буде застосувати до виділеного фрагменту.

Для використання визначених параметрів автоформату, натисніть кнопку параметри, після чого в додатковому вікні можна буде заборонити автоформату змінювати: шрифт, формати чисел, вирівнювання та інші параметри.

Часто користувач при створенні робочих аркушів має справу з визначеним набором стилів: стиль шрифту, стиль вирівнювання і т.д. У такому випадку доцільно змінити використовуваний за замовчуванням стиль. Для цього виконуються такі дії:

1. Команда **Формат/ стиль**.
2. У вікні, що розкривається, **Стиль** у списку **Ім'я стилю** виберіть стиль **Звичайний**.
3. Клацніть кнопку **Змінити** для появи вікна **Формат комірок**, у якому можна змінювати формати всіх стилів, у яких є імена.
4. **Закрийте** вікна **Формат комірок** і **Стиль** введенням **ОК**, після чого до виділеного Фрагмента застосуйте створений стиль форматування.

Багато засобів форматування Excel 97 аналогічні засобам форматування Word 97. Це полуторне і курсивне накреслення, підкреслення, колір шрифту та інші, однак в Excel існують засоби форматування, які мають аналоги у Word 97.

В Excel 97 можна використовувати вирівнювання не тільки по лівому, правому краю, центру, але і вирівнювати щодо виділених комірок, а також можна розташовувати дані в комірках вертикально або під будь-яким потрібним кутом - **Формат/ Формат комірок/ Вирівнювання**.

Якщо необхідно ввести в одну комірку велику кількість тексту таким чином, щоб Excel не обрізав вміст комірки під її наявний розмір, можна вибрати кілька варіантів розміщення тексту в комірці.

Якщо на вкладці вирівнювання обрано прапорець:

- ❖ **Переносити по словах** (Excel буде переносити текст із рядка на рядок, не виходячи за розміри комірки, лише збільшуючи її висоту і висоту рядка);
- ❖ **Автомідбір ширини** (Excel зменшить розмір шрифту комірки (наскільки можливо), щоб відобразити весь вміст комірки не змінивши її розмірів);
- ❖ **Об'єднання комірок** (Excel об'єднує злиті комірки в одну широку комірку).

За допомогою вкладки границя команди **формат/комірки** можна застосовувати до виділених комірок чи рамки границі; також додавати границю до будь-якої сторони виділених комірок і навіть провести в комірці діагональні лінії.

Своєрідними властивостями форматування Excel 97 є блокування і приховування вмісту комірок.

Заблокована комірка – комірка, вміст якої не можна змінити, якщо лист захищений.

Прихована комірка – комірка, вміст якої не можна побачити, якщо лист захищений.

Заблокувати або приховати комірки можна лише встановивши попередньо захист листа - **Сервіс/ Захист**.

Блокування комірок корисне для захисту від випадкових змін. А для додаткової схоронності на блокування комірок можна накласти ще і пароль.

2.8. Побудова графіків і діаграм в Excel 97

Діаграма призначена для графічного представлення даних у звітах, на презентаційних чи рекламних сторінках. Діаграми, які можна побудувати в Excel 97, поділяються на стандартні (найбільш розповсюджені) і нестандартні (застосовуються рідко). Стандартні діаграми бувають круговими, точковими, гістограмами і т.д.

Кругова діаграма відображає один виділений стовпець чи рядок числових даних таблиці у вигляді кола із секторами. Така діаграма відображає співвідношення частин і цілого, де ціле відповідає 100%.

Точкова діаграма (X-Y діаграма, графік) призначена для побудови традиційних математичних графіків. Перший виділений стовпець у таблиці інтерпретується як вісь X, інші – як значення однієї чи декількох функцій уздовж Y. Кількість рядків у таблиці повинна перевищувати кількість стовпців (тобто для однієї функції – три, і т.д.)

Гістограма показує числові дані з виділених стовпців таблиці у вигляді прямокутних стовпців. Такий тип діаграм в основному застосовують для ілюстрації змін у часі чи просторі.

Усі діаграми (крім кругової) мають дві осі: горизонтальну вісь категорій, вертикальну – вісь значень. Об'ємні діаграми мають третю вісь – вісь рядів.

Елементи діаграми є об'єктами, над якими визначені дії переміщення і дії з контекстного меню. За допомогою контекстного меню частіше використовують команду формат елемента.

Діаграми в Excel 97 будуються за допомогою програми Майстер діаграм, що запускається:

- ❖ Натисканням кнопки Майстер діаграм на панелі інструментів.
- ❖ Командою меню Вставка/ діаграма.

Рекомендується перед запуском майстра діаграм виділяти діапазон з даними, які необхідно відобразити графічно. Звичайно це суміжні стовпці і рядки (часто з назвами).

Побудова діаграми здійснюється в чотири етапи:

1. Вибирається тип діаграми (стандартна чи нестандартна), потім її вигляд; після чого за необхідності можливий перегляд зразка діаграми обраного типу.
2. Задаються дані (якщо не були задані раніше) встановленням перемикача рядів в положення в рядках чи стовпцях, щоб повідомити майстру діаграм як розташовані рядки даних. Для зміни даних в діапазоні можна ввести нову відносну адресу діапазону, його ім'я, або використовувати кнопку (праворуч) зміни виділеного діапазону.
3. Задаються параметри (заголовки) діаграми. Такі як: заголовок діаграми, заголовки осей, місце розташування легенди (опису осей і рядків даних), відображення вихідних даних у вигляді таблиці під діаграмою.

4. Вказується місце, де потрібно помістити побудовану діаграму, при необхідності Excel 97 створить новий лист. Діаграма поміщається на лист у вигляді об'єкта і при необхідності за допомогою маркерів зміни розміру чи переміщається, чи змінюється в розмірах.

Маркери зміни розміру – це вісім чорних квадратиків, розташованих навколо об'єкта, перетягуючи які можна змінювати розмір об'єкта.

Для переходу до наступного кроку натискається кнопка далі, до попереднього – назад. Для припинення роботи майстра діаграм (часто побудованої) і побудови діаграми використовується кнопка Готово.

Зміни в уже створеній діаграмі можна здійснити за допомогою контекстного меню елементів, команди Діаграма, чи панелі інструментів з назвою Діаграма.

2.9. Організація структур розгалуження та ітерацій в Excel 97

Організація структур розгалуження в ЕТ реалізується за допомогою логічної функції **ЯКЩО**, що використовується у формулах і має вигляд:

ЯКЩО (< умова >, < вираз 1 >, < вираз 2 >).

Якщо умова вірна, тоді функція набуває значення першого виразу, інакше виразу два. Вирази 1, 2 можуть також бути логічними функціями ЯКЩО. У такий спосіб утворюються вкладені розгалуження. Excel 97 допускає застосування рівнів вкладення не більше семи. Умови записуються як в алгоритмічних мовах – за допомогою операцій порівняння =, >, <, <=>=> <>, визначеними над виразами. Часто виразом є тільки адреса комірки, що містить деяке значення.

Організацію ітерацій у ЕТ Excel 97 розглянемо на прикладі розв'язання нелінійного рівняння

$$2x - 1 = \sin x$$

Це нелінійне рівняння за допомогою ЕТ можна розв'язати чотирма способами.

Спосіб 1. Метод простих ітерацій. Щоб рівняння можна було розв'язати методом простих ітерацій, його приводять до вигляду:

$$x_{i+1} = (\sin x_i - 1) / 2, \text{ де } x_i - \text{будь-яке початкове наближення.}$$

Розв'язання нелінійного рівняння $2x-1=\sin x$ методом простої ітерації

A	B	C
Попереднє	наступне	похибка
2.0000.	$=(\sin(A6)+1)/2$	$=ABC(B6-A6)$
B6		
17.0.887862212	0.887862212	

Інший спосіб полягає у використанні властивості ЕТ Excel 97 автоматичного багаторазового перерахування, якщо включений режим

ітерацій у діалоговому вікні сервіс Параметри. У цьому випадку для розв'язання цієї задачі достатньо двох комірок.

1. 2.000
2. $= (\sin (A17)+1)/2 \rightarrow B17$
3. $= (\sin (A17)+1)/2 \rightarrow A17$

Також нелінійне рівняння можна розв'язати методом підбору параметра. Ще один спосіб полягає у використанні можливостей програми Solver, що додається до Excel 97.

Контрольні запитання

1. Що таке електронна таблиця та її призначення?
2. З чого складається робоча книга?
3. Які типи даних є у ЕТ "Ексел"?
4. Яке призначення робочої комірки?
5. Що таке копіювання формул?
6. Що таке автоматичне переобчислення?
7. Наведіть приклади статистичних функцій?
8. Які математичні функції є в ЕТ?
9. Як організуються розгалуження?
10. Як організуються ітерації?
11. Яке призначення і які є види діаграм?
12. Як створити діаграму і зробити в ній зміни?
13. Як змінити тип діаграми?

3. Система керування базами даних (СКБД) "Access"

3.1 Загальні відомості про бази даних. Типи даних, які зберігаються в них

Під базою даних (БД) розуміється велика кількість однорідних даних, організованих певним чином, що відносяться до деякої предметної області і зберігаються на комп'ютерних носіях.

БД створюються за необхідністю обробки великих об'єктів однорідних даних. Таких як: списки абітурієнтів, співробітників підприємств (анкетні дані), пропозиція товарів на ринку, облік товарів на складах і т.д.

Робота в БД містить у собі такі етапи:

1. Створення структури БД;
2. Введення даних;
3. Редагування структури і даних;
4. Пошук інформації в БД;
5. Оформлення звітів.

Для виконання перерахованих вище дій існують спеціальні програми, що називаються СКБД Access, FoxPro, Dbase і т.д.

У Access БД – файл, що містить дані у вигляді однієї чи декількох таблиць. Таблиця в БД складається з рядків (записів) і стовпців (полів).

Запис містить інформацію про один елемент БД: людину, книгу, одиницю продукції і т.д. і складається з полів, що формують структуру запису.

Поле – це мінімальна (найважливіша) порція інформації в записі, над якою визначені операції введення, виведення, перетворення і т.д. Кожне поле має своє значення, ім'я, характеризується типом. Розглянемо типи даних, які можна зберігати в таблицях Access 97.

❖ Текстовий. Текстові дані складаються з букв, цифр і спеціальних символів. Їх можна тільки записати, але з ним не можна робити ніякі дії. (Приклади: імена, адреси, назви міст, назви виробів).

❖ МЕМО. У полі МЕМО можуть зберігатися тексти великого розміру. Поля МЕМО займають багато місця, тому їх вводити в таблицю рекомендується тільки за необхідності.

❖ Числовий – зберігаються звичайні числа. Ці поля використовуються для проведення обчислень.

❖ Поля дата / час – зберігаються значення дат і часу (у форматі, аналогічному формату Excel 97. Access 97 дозволяє використовувати встановлені в Windows стандарти і вводити дані у форматі необхідної країни.)

❖ Грошовий – зберігаються грошові значення, округлені і вивірні по десятковому розділювачу. Розпізнаються національні стандарти і використовується національна грошова одиниця.

❖ Поле лічильника – зберігаються наступні номери, по одному номеру для кожного запису в таблиці. (При додаванні записів у таблицю Access автоматично поновлює значення цього поля.)

❖ Логічне поле ТАК/Ні використовується, щоб вказати наявність чи відсутність елемента, або дати відповідь на питання.

❖ Об'єкти OLE – це впроваджені об'єкти типу діаграм, створених Excel 97. У базах даних Access 97 можуть зберігатися всі типи OLE-сумісних впроваджених об'єктів. (Object Linking & Embedding – зв'язування і впровадження об'єктів з інших програм, які можна вставити в таблицю Access 97.

❖ Гіперпосилання. Наприклад адреса вузла Web сторінки в Internet.

У кожній таблиці БД Access має задаватися первинне ключове поле. Первинне ключове поле - це поле в якому не міститься однакових записів. Таке поле має бути унікальним для кожного запису. Користувач може сам призначити ключове поле, а може доручити Access автоматично додати ключове поле в таблицю. У цьому випадку до ключового поля буде застосована автоматична нумерація, щоб у кожному ключовому полі містилися різні значення. Поняття ключового поля використовується в Access для швидкого й однозначного пошуку необхідних значень.

3.2. Створення структури БД. Об'єкти, що використовуються в Access 97

Існує кілька способів створення структури БД у Access 97.

❖ Використання майстра БД:

❖ Використання конструктора таблиці.

Існує цілий набір майстрів БД, що дозволяють будувати БД по наборах існуючих шаблонів. Такі майстри можуть побудувати цілий спектр БД таких як:

❖ Адресна книга

❖ Бібліотека

❖ Основні фонди

❖ Тренування і т.д.

Однак у більшості випадків майстри не можуть створити точно такий тип бази даних, який потрібний користувачу і її буде необхідно змінювати після закінчення роботи майстра.

Змінити існуючу БД не маючи досвіду роботи з Access досить складно тому варто створювати свою БД вручну.

Запустивши Access і вибравши параметр Нова база даних – ОК, необхідно ввести назву нової БД і натиснути кнопку Створити, щоб вивести на екран вікно бази даних. У вікні БД перелічуються назви об'єктів вашої майбутньої БД і відображається центральна панель керування для роботи з Access.

Об'єкт БД - це частина БД Access. Наприклад таблиця Access – об'єкт, звіт БД у якому надруковані дані це теж об'єкт. Але самі дані не є об'єктами.

В якості об'єктів Access 97 можуть використовуватися:

1. Таблиці. (Таблиці – це зв'язані дані в базі)
2. Запити. (Запити – записані в пам'ять інструкції з вибору даних для складання звітів, аналізу і керування даними).
3. Форми - це екранний варіант паперових форм чи бланків, що використовується для введення даних у таблицю.
4. Звіти – надруковані списки вибрані з бази даних.
5. Макроси – списки команд, що зберігаються в пам'яті, які повинні виконати для вас програма Access .
6. Модулі – програми, написані мовою Visual Basic для додатків, за допомогою яких можна автоматизувати будь-яку процедуру, що виконується в БД.

Вкладка таблиця знаходиться першою у вікні БД, і не випадково, оскільки створення таблиці полягає у введенні даних, які надалі використовують інші об'єкти БД.

1. У вікні БД вибрати вкладку Таблиці і клацнути кнопку Створити.

2. Вибрати опцію Конструктор, щоб відобразити структуру таблиці, що створюється з відносними полями і їх властивостями.

3. Ввести ім'я першого поля БД. Це ім'я не має ніякого відношення до типів даних, що можливо, зберігаються в даному полі таблиці, але дозволяє звертатися до поля при створенні таблиці. Тільки коли закінчиться створення рядків таблиці, потрібно буде ввести в таблицю дані.

4. Натиснути TAB для вибору типу даних поля із списку Тип Даних, що розкривається.

5. За необхідності натиснути TAB і ввести опис поля, що полегшить роботу згодом.

Після введення імені, типу даних і опису (не обов'язково) необхідно в більшості випадків задати ще і властивості полів. Для кожного типу даних можна встановлювати різні набори властивостей поля. При створенні таблиці можна установити такі значення властивостей поля:

❖ Розмір поля - обмежує кількість символів, що можуть вводитися в дане поле, забороняючи таким чином наступне введення даних у це поле.

❖ Формат поля – відображається список, що розкривається, з переліком форматів, що можуть застосовуватися для даних цього поля.

❖ Підпис – містить текст, що відображається, при введенні користувачем даних у це поле таблиці. Якщо не вказати підпис, то Access буде використовувати тип поля.

❖ Значення за замовчуванням - значення, що додається в поле за замовчуванням, яке з'являється при введенні користувачем даних у таблицю.

❖ **Обов'язкове поле** - містить "Так" чи "Ні", щоб вказати Access чи обов'язково вводити значення в це поле до того як користувач може зберегти запис даних у таблиці. Якщо розглянете поле визначене як ключове, тоді в рядку **обов'язкове** необхідно вказати **ТАК**.

❖ **Число десяткових знаків** - визначається кількість десяткових знаків установлених за замовчуванням.

Якщо ви закінчили додавання полів у таблицю, то не зможете закрити таблицю в режимі конструктора, не вказавши її ключового поля. Access може додати ключове поле використовуючи формат автоматичної нумерації. Вручну ключове поле можна задати, клацнувши ліворуч від імені поля, а потім клацнувши на панелі інструментів кнопку Ключове поле. Після задання ключового поля необхідно зберегти таблицю використовуючи кнопку на панелі інструментів Зберегти.

3.3. Створення таблиці БД за допомогою конструктора таблиць

Крім режиму конструктора, Access дозволяє працювати з таблицею у так званому режимі таблиці: Панель інструментів → кнопка Режим таблиці.

У режимі таблиці нічого не буде відбите в ній доти, поки в таблицю, яка відтворюється, не будуть введені дані. Після введення даних у таблицю, у режимі таблиця з'являється можливість побачити свою створену таблицю у вигляді рядків і стовпців, як на листі Excel.

У режимі таблиці також можливе створення структури таблиці і введення в неї даних, однак це досить трудомісткий і недоцільний процес.

Зміна структури таблиці

На відміну від інших СКБД Access дозволяє легко змінити структуру таблиць, навіть після введення в них даних.

Для змін структури таблиці необхідно переключитися в режим конструктора таблиць, вибравши у вікні бази даних ім'я таблиці і клацнути кнопку Конструктор.

Для додавання поля в кінець таблиці, просто клацніть перше порожнє поле, виберіть вкладку Ім'я поля і введіть відомості про поле, як при додаванні в таблицю вихідних даних. Для додавання поля між рядками слід вибрати рядок, клацнути правою кнопкою і вибрати команду додати рядок.

Для видалення поля, досить у режимі конструктора його виділити і натиснути DEL.

3.4. Робота з формами в Access 97

Раніше ми розглянули основні засоби створення таблиць для БД: створення таблиці в режимі конструктора і в режимі таблиць. Крім перерахованих способів створення таблиць і додавання в них даних Access надає такий засіб для додавання і візуалізації даних у таблиці як форми.

Форма за зовнішнім виглядом подібна до бланка або картки.

Прикладами форми можуть бути сторінка з паспорта, картка з досьє чи бібліографічна картка для бібліотечної книги.

Одна форма містить дані лише з одного запису. Перехід до форм крім гращої візуалізації даних, дає можливість не використовувати ряд додаткових можливостей. Access 97 дозволяє розміщати на формі зручним чином такі елементи:

- поля типу OLE з картинками, фотографіями;
- елементи керування: кнопки, перемикачі;
- надписи, заголовки форми, рубрик, а також розрисовувати форму чи задавати фоновий рисунок - заставку (хмари, глобус);
- поля для обчислень - для відображення результатів обчислень над вмістом інших полів;
- закладки - багатосторінкові форми, де поля групують у залежності від змісту на різних закладках.

Існує кілька способів створення форм:

- автоматично - за допомогою команди Автоформа;
- за допомогою майстра форм;
- вручну за допомогою конструктора форм;
- комбінованим способом.

Наприклад, виконавши команду Автоформа → у стовпчик одержуємо форму, у якій усі поля з таблиці будуть розташовані у стовпчик і вирівняні по лівому краю.

Досить простим і зручним способом створення форм є використання майстра форм. Щоб запустити майстер форм необхідно:

1. Відобразити вікно бази даних.
2. Клацнути вкладку Форми, щоб активізувати її.
3. Клацнути кнопку Створити.
4. У вікні Нова форма, що з'явилася, необхідно вибрати рядок

Майстер форм.

5. У нижній частині діалогового вікна обирається таблиця, що використовується, як джерело даних при створенні форми.

6. Якщо клацнути ОК, то з'явиться перше діалогове вікно в якому перераховані поля, обраної таблиці, що можуть бути включені у форму.

Щоб включити деяке поле у форму, його необхідно включити і натиснути кнопку >. Якщо необхідно включити всі поля натисніть >>.

Аналогічно для видалення вже обраного поля натисніть < , для видалення всіх <<.

Натисніть кнопку далі для переходу до наступного діалогового вікна, у якому можна буде вибрати різні види форм:

- рядкова і таблична більше схожі на таблицю;
- у стовпчик і вирівняний більше схожі на звичайні паперові бланки.

Перейшовши до наступного діалогового вікна можна вибрати стиль форми. Багато стилів мають спеціально розроблений дизайн, що дозволяє будь-якій непримітній за змістом форми привернути до себе увагу. В останньому діалоговому вікні необхідно ввести ім'я форми, що створюється. За замовчуванням, ім'я форми збігається з ім'ям обраної таблиці.

Що стосується режиму конструктора форм, то цей режим має досить великі можливості по створенню і редагуванню вже існуючих форм, однак вимагає певних навичок роботи і досвіду.

У режимі конструктора робоче вікно розділене на три робочі області:

- заголовок форми;
- область даних;
- примітка форми.

Для конструювання форми використовують можливості панелі інструментів, команд контекстного меню і панелі елементів керування з кнопками.

Щоб вставити у форму елемент керування, необхідно натиснути кнопку запуск майстра, вибрати елемент керування і клацнути на формі в місці, куди необхідно вставити елемент керування.

Звичайно у формах відображаються окремі записи. На відміну від режиму таблиці, форми є записами. У режимі таблиці ви часто можете бачити багато записів, але не всі поля, тому що поля рідко вміщуються на екрані в режимі таблиць. У формі відображається тільки один запис, але часто до неї можуть бути включені всі поля.

Для переміщення в режимі форми між полями одного запису використовуйте кнопки зі стрілками TAB, Shift-TAB, між записами Page-up чи Page-Down або кнопки керування. І нарешті, якщо в режимі форми вибрати команду Файл/попередній перегляд, то Access подасть до уваги лист і розмістить на ньому ту кількість форм, скільки може поміститися.

3.5. Робота з фільтрами і запитами

Під фільтром розуміється можливість створення тимчасової підмножини записів з таблиці. Access 97 підтримує три способи фільтрації:

- Фільтр по виділеному фрагменту. Працює з виділеними даними таблиці.

- Звичайний фільтр. Він дозволяє вибрати поля з даними, що Access буде використовувати для фільтрації.

- Розширений фільтр/сортування. Він керує параметрами розширеної фільтрації за допомогою меню Access.

Більш простими фільтрами, які використовуються досить часто, є фільтри по виділеному фрагменту і звичайний фільтр, які розглянуті нижче.

Фільтр по виділеному фрагменту.

Для створення такого фільтру необхідно виконати такі дії:

1. Відобразити таблицю з даними в режимі таблиці.
2. Знайти один запис з елементом, що цікавить. Якщо виділити тільки частину вмісту поля, то фільтрація буде здійснюватися тільки за цією ознакою.

3. На панелі інструментів клацнути кнопку Фільтр по виділеному. Вихідна таблиця перетворюється таким чином, що містить тільки записи, які мають поля, що задовольняють фільтр.

4. Для повернення в режим відображення повної таблиці клацніть на панелі інструментів, кнопку видалити фільтр. Таким чином використовується контекстне меню поля Фільтр.

Звичайний фільтр.

Звичайний фільтр дозволяє провести фільтрацію по декількох значеннях, а не тільки по одному. Звичайно фільтрацію можна зробити як у режимі таблиці, так і в режимі форми.

Щоб використовувати Звичайний фільтр, необхідно:

1. На панелі інструментів клацнути кнопку Змінити фільтр і Access відобразить один порожній запис.

2. Вибрати поле, по якому хочете провести фільтрацію і клацнути порожнє поле.

3. Клацнути, стрілку, що з'явилася в полі списку.

4. У списку всіх наявних значень, що розкривається, вибрати значення, по якому необхідно провести фільтрацію.

5. За необхідністю клацнути інше порожнє поле і вибрати зі списку, що розкрився, інше значення. Звичайний фільтр дозволяє зробити фільтрацію по декількох полях.

6. Клацнути кнопку Застосування фільтра, щоб відобразити відібрані записи.

7. Клацнути кнопку Видалити фільтр для повернення до звичайного режиму перегляду: до фільтрації та ж кнопка.

Access 97 дозволяє робити фільтрацію з використанням логічних операторів "чи" і "і". Для цього необхідно використовувати нижній рядок <знайти > <чи>.

Запит - об'єкт бази даних, що містить у собі збережений набір команд вибору записів БД. Як і в інших об'єктах у запитів є імена. Багато запитів є усього лише фільтрами. Але фільтри не зберігаються по закінченні роботи, а запит можна згодом викликати за іменем і використовувати повторно. Хоча не можна дати фільтрам імена, але можна перетворити фільтр в іменованій запит за допомогою кнопки панелі інструментів Зберегти як запит.

Що стосується створення запитів, то в Access 97 існують такі чотири майстри запитів:

- Майстер простого запиту - вибирає поле з однієї чи декількох таблиць і з інших запитів.

- Майстер перехресного запиту - створює запит у вигляді таблиці, у якій значення полів згруповані по горизонталі і вертикалі.

- Майстер запиту записів, що повторюються - створює підмножину даних з декількох таблиць чи запитів, що містять потрібні значення в одному чи декількох запитах.

- Майстер запиту записів без підлеглих - він створює підмножину даних з декількох таблиць чи запитів, що не містять однакових записів.

Найчастіше використовується майстер простих запитів, через його неспеціалізоване призначення.

Щоб побудувати простий запит за допомогою майстра необхідно:

1. У діалоговому вікні Простий запит вибрати рядок Простий запит. Access 97 запустить майстра і відобразить на екрані діалогове вікно Створення простих запитів.

2. Вибрати таблицю (чи існуючий запит) для даних, що будуть обрані. За допомогою запиту Access відобразить поля обраної таблиці чи запиту в списку доступні поля.

3. Вибрати вікно чи декілька полів і клацнути кнопку зі значком >. Access включить ці поля в підсумкову підмножину даних запиту.

4. При бажанні зі списку таблиці, що розкриваються, (Запити) можна вибрати іншу таблицю чи запит і занести додаткові поля до списку Обрані поля.

5. Клацнути кнопку Далі, щоб вибрати докладний чи короткий запит. У докладний запит входять усі поля всіх записів.

6. Далі вибирають ім'я запиту і після натискання Готово, Access97 побудує запит і відобразить на екрані обрані з запиту записи в режимі таблиці.

3.6. Побудова запитів у режимі конструктора

Перед розглядом особливостей побудови запитів у режимі конструктора відзначимо особливості запитів взагалі.

1. Access не зберігає результати запиту, і для перегляду підмножини даних у вікні бази даних необхідно обрати вкладку Запити і клацнути для відкриття. При цьому для виведення даних буде потрібно якийсь час, адже запит завжди буде оновлений, тобто в запиті будуть відбиті дані з врахуванням останніх змін у таблиці.

2. У разі потреби використання підмножини даних у Access 97 не потрібно зберігати дані двічі (один раз у вихідних таблицях, інший раз у запиті).

3. При зміні даних запиту (або в режимі конструктора, або - форми) Access поновлює дані у вихідних таблицях.

Режим конструктора.

Режим конструктора необхідно використовувати в двох випадках:

- коли необхідно створити складний запит, що неможливо створити за допомогою майстрів;
- коли необхідно змінити існуючий запит.

Створення запиту:

1. У вікні бази даних вибрати вкладку Запити і натиснути кнопку створити.

У діалоговому вікні Новий запит вибрати Конструктор.

2. У діалоговому вікні Додавання таблиці виберіть таблиці (у вікні таблиці) або запит (у вікні запити), у залежності від того що вибирається.

Таким чином, вікно конструктора набуває вигляд вікна, розділеного навпіл, де у верхньому вікні відбитий перелік вихідних таблиць і запитів, а в нижньому вікні відбиті умови відбору (команди відбору). Умови відбору утворюються шляхом введення необхідних умов у відповідні поля нижнього вікна.

3.7. Реляційні таблиці, запити, форми

Дві чи кілька таблиць, що мають хоча б одне загальне поле між якими існує зв'язок по загальному полю, називаються реляційними.

Реляційними також є форми, запити, звіти, побудовані на основі декількох реляційних таблиць.

Для реляційного зв'язку використовуються поля, у яких значення не повторюються, наприклад, числове поле типу лічильник, поле з персональними номерами співробітників і т.д., тобто ключові поля.

Здавши ключове поле хоча б в одній таблиці, можна встановлювати зв'язок між таблицями командою Сервіс - Схема даних. У вікні Схема даних вставляють необхідні таблиці, а зв'язок встановлюють методом перетягування і накладення назви поля з однієї таблиці на назву поля в іншій. Зв'язок являє собою лінію, що з'єднує загальні поля декількох

таблиць. Зв'язок можна анулювати клацнувши на лінії зв'язку і натиснувши DEL.

Для створення реляційних запитів, звітів і форм спочатку відкриваються їх конструктори, у вікно конструктора вводяться необхідні таблиці і тільки тоді формують зв'язки.

3.8 .Оформлення звітів в Access 97

Звіти призначені для оформлення необхідних даних із БД, відповідно до вимог стандартів чи замовника і представлення в друкованому вигляді.

Існує три способи створення звітів:

- автозвіт;
- використання майстрів звітів;
- конструктор звітів.

Як і у випадку форм оптимальним способом створення звітів є комбінований метод: звіт створюють за допомогою майстра звіту, а пізніше оформляють його вручну за допомогою конструктора.

Автозвіт - інструмент автоматичного складання звіту, що працює в режимі таблиці. Для створення автозвіту необхідно відобразити дані в режимі таблиці (якщо звіт на основі даних з декількох таблиць, тоді необхідно відобразити дані в режимі запиту) та на панелі інструментів натиснути кнопку ▼ - стрілка вниз (поруч із кнопкою новий об'єкт). В списку, що розкривається, вибрати Автозвіт. У режимі попереднього перегляду ознайомтеся зі звітом. Закрийте вікно попереднього перегляду автозвіту, потім вікно режиму конструктора звіту і введіть ім'я звіту.

Якщо структура чи зовнішній вигляд звіту не досить інформативні можна вдатися до послуг майстрів звітів.

У Access 97 існують такі види майстрів звітів, які можна побачити, клацнувши кнопку створити вкладки Звіту, що знаходяться у вікні БД.

• *Конструктор*. У цьому режимі створюється звіт "з нуля", в який вручну можна додати колонтитули, дані, підсумки, елементи керування.

• *Майстер звітів*. Допомагає створити звіт, надає можливість вибрати вихідні таблиці і запити, а також поля які ви хочете використовувати. Цей майстер дозволяє створити звіти найбільш універсального характеру.

• *Автозвіт у стовпець*. Створює звіт, що містить всі поля з таблиці чи запиту, на основі яких він побудований. Багато в чому схожий на Автоматичний, але створює більш шаблонний звіт, куди можуть бути включені заголовки, спеціальні шрифти, а також можуть бути виділені поля даних.

• *Автозвіт рядковий*. Створює звіт, у якому кожен запис з вихідної таблиці чи запиту розглянутий в окремому рядку, а розмір шрифту

змінений таким чином, щоб звіт міг поміститися на сторінці. Фактично в лінійних звітах створюються списки даних, що мають гарну наочність.

• *Майстер діаграм.* Буде діаграми на основі ваших даних. Діаграми Access схожі на діаграми Excel. Можливо змінити як формат, стиль діаграм, так і вихідну таблицю з даними.

• *Поштові наклейки.* Звіт загального списку розсилання, що створює роздруківку у вигляді поштових наклейок. Під час друку поштових наклейок використовується стандартна система нумерації (система нумерації Avery), одна з найвідоміших комп'ютерних компаній з випуску поштових наклейок.

Для запуску майстра звітів необхідно:

- відкрити вікно БД і відобразити вміст вкладки звіт;
- клацнути кнопку створити, щоб відобразити діалогове вікно Новий звіт і вибрати майстер звітів;

- відкрити список таблиць і звітів, що розкривається, і вибрати в ньому таблицю та кількість вихідних даних.

- у списку Доступні поля необхідно вибрати поля, по яких проводиться звіт, і перемістити їх у список Вибрані поля.

- у наступному діалоговому вікні необхідно задати Access спосіб групування даних і підведення підсумків. Справа в тому, що рідко готуються звіти, які не містять розрахунків, і тому в залежності від типу даних, обраних полем, майстер звітів може запропонувати відомості про суму, часткову суму та інше. Якщо в звіті містяться числові дані, то майстер звітів виведе на екран діалогове вікно підсумків, у якому над даними можна буде зробити такі обчислення:

- SUM - обчислює загальну суму значень в полі;

- Avg - обчислює середнє значення в полі;

- MIN - Access виділяє кольором мінімальні значення в полі;

- MAX- Access виділяє кольором максимальні значення в полі.

Звернемо увагу на загальну структуру звіту незалежно від способу створення останнього:

- заголовок звіту;

- верхній колонтитул (що повторюється на всіх сторінках);

- заголовок групи;

- область даних;

- нижній колонтитул;

- примітки;

- рисунки, діаграми й інші об'єкти.

Контрольні запитання

1. Що таке база даних?
2. Які функції СКБД "Access".

3. Що таке база даних в "Access".
4. Як створити структуру бази даних?
5. Для чого призначені майстри бази даних?
6. Яке призначення конструктора таблиці?
7. Яке призначення форм?
8. З чого складається форма?
9. Які переваги таблиць над формами і навпаки?
10. Яке призначення фільтрів?
11. Які є типи запитів?
12. Які є засоби створення запитів?
13. Що таке реляційна таблиця?
14. Що таке реляційна форма?

4. ПАКЕТИ ПРИКЛАДНИХ ПРОГРАМ ДЛЯ АВТОМАТИЗАЦІЇ МАТЕМАТИЧНИХ ОБЧИСЛЕНЬ

4.1. Основи інтерфейсу користувача MathCAD

Під інтерфейсом користувача мається на увазі сукупність засобів графічної оболонки, що забезпечують легке керування системою як із клавіатури, так і за допомогою миші. Під керуванням розуміється і просто набір необхідних символів, формул, текстових коментарів і т.д. та можливість повної підготовки в середовищі MathCAD документів і електронних книг, з наступним запуском їх у реальному часі. Інтерфейс користувача системи розроблений так, що користувач, який має елементарні навички роботи з Windows – додатками миг відразу почати роботу з MathCAD. Зовні інтерфейс системи схожий на інтерфейс текстових процесорів Word 97:

File – команда рядка меню служить для роботи з файлами документів, крім відомих позицій, існує група команд, які зустрічаються вперше і забезпечують:

Collaboratory – установку зв'язку з фірмою, розроблювачем системи для забезпечення спільної роботи над документами;

Internet Setup – установку модемного зв'язку з Internet;

Send – відправлення документа електронною поштою по мережі Internet;

Edit – служить для редагування файлів документів;

Links – задання зв'язку для редагування файлів документів;

Object – редагування вставленого в текст об'єкта (приклад);

View – команда для зміни засобів огляду;

Insert – включення/виключення елементів інтерфейсу;

Graph – здійснюється вставка шаблонів графіки з вибором їх з підменю.

Більшість параметрів графічного процесора, необхідних для побудови графіків задаються за замовчуванням автоматично, тому для початкової побудови необхідно задати тип графіка:

X Y Plot @ - створити шаблон двовимірного графіка в декартовій системі координат;

Polar Plot (Ctrl + 7) - створити шаблон графіка в полярних координатах;

Surface Plot (Ctrl + 2) – створити шаблон для побудови тривимірного графіка;

Contour Plot (Ctrl + 5) – створити шаблон для контурного графіка тривимірної поверхні;

3D scatter Plot – створити шаблон для графіка у вигляді точок (фігур) у тривимірному просторі;

Vector Field Plot – створити шаблон для графіка векторного поля на площині;

3D Bar Chart – створити шаблон для зображення у вигляді сукупності стовпців у тривимірному просторі;

Matrices Ctrl + M – забезпечує задання векторів чи матриць (з'являється діалогове вікно, у якому задається розмірність матриці і вводяться дані);

function (Ctrl + F) – пропонує вибрати функцію з наявних, пояснює її призначення і дозволяє коректно ввести шаблон з ім'ям функції в необхідне місце;

unit (Ctrl + U) – вставка одиниць вимірювань розмірних величин;

Picture (Ctrl + T) – вставка шаблону імпортованого рисунка. Дозволяє включати в документи MathCAD високоякісні графічні ілюстрації, що необхідні при створенні електронних книг (вставляє елементи графічних редакторів класу Paint Brush, системи AutoCAD, PCAD і ін.);

Math Region – вставка в текстову область шаблону математичної області для створення невиконаного коментарю у вигляді математичної формули;

Text Region – вставка текстової області (для створення текстових коментарів);

Hyperline – вставка гіперпосилання (рядка чи символів слів, підкреслених знизу й асоційованих з визначеним об'єктом);

Reference – вставка звертання до заданого файлу активізацією кнопки;

Componenet – вставка компонентів інших математичних систем і використання вхідних у них засобів і функцій (напр., MathCAD 4.0, Excel, Axum і ін.);

Object – вставка об'єкта з встановленням динамічного зв'язку з додатком, що його породжує. Ця можливість дозволяє легко встановлювати зв'язок з додатками систем класу Windows 98 (напр. Word, Access);

View – опція керування виглядом документа;

Toolbar – забирає чи вставляє панель із кнопками інструментів для швидкого керування системою;

Format bar – забирає чи відновлює панель форматування;

Math pallette – забирає чи виводить панель висновку палітр математичних символів;

Regions – дозволяє виділити всі області і забезпечує зафарбування проміжків між ними сірим кольором (самі блоки при цьому виділені білим тлом);

Zoom – виводить вікно зі списком різних масштабів і дозволяє змінити масштаб символів документа;

Refresh - усуває перекручування зображення на екрані дисплея, що можуть бути викликані взаємним накладенням блоків один на одного чи з інших причин;

Animation - дозволяє готувати анімаційні графіки;

Playback - виводить програвач анімаційних файлів;

Format - зміна формату (параметрів) об'єкта;

Number - установка формату чисел;

Equation - установка формату виразів;

Text - установка формату тексту;

Paragraph - установка формату параграфа;

Style - установка формату стилю;

Properties - установка властивостей;

Graph - установка формату графіків;

Color - установка кольорів об'єкта;

Separate region розділення областей для усунення їхнього перекриття;

Align region – задання розташування областей виведення символічних обчислень;

Lock region - створення закритих (недоступних для редагування) областей;

Header Footer - створення колонтитулів (написів на кожній сторінці, видимих при їх роздрукуванні).

Math - ця позиція забезпечує керування обчислювальним процесом системи MathCAD. Її операції :

Calculate (F9) - запускає обчислення при відключеному режимі автоматичних обчислень з місця, на якому знаходиться маркер миші;

Calculate Worksheet - запускає обчислення при відключеному режимі автоматичних обчислень, починаючи від початку до кінця документа;

Automatic calculation - встановлює режим автоматичних обчислень для переглянутого документа (цей режим включений за замовчуванням);

Optimize - включає оптимізацію обчислень (заміна складної функції чи математичного виразу їх аналітичним представленням (якщо воно є);

Options - задає опції обчислювальних процесів;

Команда **Options** виводить підменю для установки таких опцій:

Built - in- установка значень убудованих (системних) Variables змінних;

Unit system- установка систем вимірювання фізичних величин;

Dimension - установка розмірності фізичних величин;

Graphics - робота з графічним редактором.

Symbolic - містить операції, що відносяться до роботи символічного процесора. Символьні операції розбиті на 5 характерних груп.

1) *Операції з виділеними виразами:*

Evaluate - перетворити вираз з вибором виду перетворень з підменю;

Evaluate Symbolically (Shift+F9) - виконати символічне обчислення виразу;

Floating Point Evaluation - виконати арифметичні операції у виразі з результатом у формі числа з плаваючою точкою;

Complex Evaluation - виконати обчислення з представленням операцій у комплексному вигляді;

Simplify - спростити виділений вираз з виконанням таких операцій, як скорочення подібних що складаються, приведення до загального знаменника і т.д.;

Expand - розкрити вираз;

Factor - розкласти число чи вираз на множники;

Collect - згрупувати подібні до виділеного вирази, що може бути окремою функцією зі своїм аргументом, чи змінною (результатом буде вираз, поліноміальний щодо обраного виразу);

Polynomial - знайти коефіцієнти полінома за заданою змінною, що наближає вираз, у якому ця змінна використана.

2) *Операції з виділеними змінними:*

Solve - знайти значення виділеної змінної, при яких вирази, що її містять, дорівнюють нулю;

Substitute - замінити зазначену змінну вмістом буфера обміну;

Differentiate - диференціювати усі вирази, що містять виділену змінну, по цій змінній (інші змінні розглядаються як константи);

Integrate - інтегрувати усі вирази, що містять виділену змінну, по цієї змінній;

Expand to Series - знайти кілька членів розкладання виразу в ряд Тейлора щодо виділеної змінної;

Convert to Partial Fraction - розкласти на елементарні дроби вирази, що розглядаються як раціональні дроби, по виділеній змінній.

3) *Операції з виділеними матрицями, представлені підменю Matrix:*

Transpose - одержати транспоновану матрицю;

Invert - створити обернену матрицю;

Determinant - обчислити детермінант (визначник) матриці.

4) *Операції перетворення:*

Fourier Transform - виконати пряме перетворення Фур'є щодо виділеної змінної;

Laplace Transform - обчислити пряме перетворення Лапласа щодо виділеної змінної;

Inverse Laplace Transform - виконати зворотне перетворення Лапласа;

Z Transform - обчислити пряме Z перетворення виразу щодо виділеної змінної;

Inverse Z Transform - обчислити зворотне Z перетворення виразу щодо виділеної змінної.

5) *Стиль символічних перетворень:*

Evaluation Style – задати результат символної операції під основним виразом, поруч з ним чи замість нього;

Window – керування вікнами системи;

Books – робота з електронними книгами;

Help – робота з довідковою базою даних системи.

Усі зазначені операції можна виконувати двома способами:

- безпосередньо в командному режимі (використовуючи, описані вище операції в позиції Symbolic головного меню);

- за допомогою оператора символних операцій і операцій, поданих у палітрі символних обчислень.

Виконання символних операцій із застосуванням оператора \rightarrow є переважним у силу таких їхніх властивостей:

- можна поруч задавати операції з різних опцій і організувати деревоподібну структуру символних операцій;

- можна використовувати операції над виразами з функціями користувача;

- при зміні тих чи інших виразів по ланцюжку обчислень змінюються і результати обчислень;

- можливе застосування деяких функцій (наприклад границь, яких немає в командному режимі);

- із застосуванням оператора \rightarrow можна виконати як усі наведені вище операції в командному режимі символних обчислень, так і ряд інших операцій і опцій.

Серед них основними є:

- complex - перетворення в комплексній формі;

- assume - присвоювання змінним невизначеного значення, навіть якщо до цього були присвоєні значення, і накладання обмежень на значення змінних;

- float - перетворення у формат чисел із плаваючою точкою;

- literally - заборона символного перетворення для наступного виразу;

- rfac - розкладання на елементарні дробі;

- coeffs - повернення коефіцієнтів полінома;

- M^T - транспонування матриці;

- M^{-1} - інвертування матриці;

- M - обчислення детермінанта матриці.

4.2. Вхідна мова системи MathCAD 8.0

-Вхідна мова системи – це математично орієнтована мова описування задач, що є проміжною ланкою в спілкуванні користувача й ЕОМ.

Вхідна мова системи MathCAD 8.0. містить у собі:

1. **Alphabet**, що є набором символів з якого складаються об'єкти мови, що включає: малі і великі латинські і грецькі букви, букви кирилиці, арабські цифри 0.. 9, ідентифікатори системних змінних, спеціальні знаки, оператори, імена вбудованих функцій, спеціальні знаки.

2. **Типи даних**. До основних типів даних відносяться: константи; звичайні і системні змінні, масиви (вектори і матриці) і дані файлового типу. **Константи** – це об'єкти імена яких відомі системі. В якості констант можуть використовуватися: цілочисельні дійсні числа з мантисою і порядком, вісімкові числа із значеннями 0..8; шістнадцяткові числа з основою 16 від 0 до F, де H наприкінці числа; комплексні числа $Z = \text{Re}Z + I * \text{Im}Z$; системні константи (e), строкові константи («...»), одиниці вимірювання фізичних величин, а також типи фізичних величин такі, як time, tenth, mass, charge.

3. **Оператори** – елементи мови, за допомогою яких можна створювати математичні вирази. Оператори виконуються тільки після вказівки операндів (тобто аргументів відповідних операцій).

4. **Функції** – об'єкти, у відповідь на звертання до яких по імені з вказівкою аргументу, повертають значення (символьні, числові, векторні чи матричні). Поряд з вбудованими функціями можуть задаватися і функції користувача. При виконанні символічних операцій системні константи використовуються тільки в символічному вигляді і приймають числові значення тільки під час виведення результатів обчислень.

5. **Змінні** – об'єкти, яким можна привласнювати різні значення. Імена констант змінних та інших об'єктів називаються ідентифікаторами. У MathCAD тип змінної задається її значенням, тому змінні в MathCAD можуть мати будь-яку довжину. Малі і великі букви розрізняються. Ідентифікатори повинні бути унікальними, тобто не можуть збігатися з іменами вбудованих чи визначених користувачем функцій. Підозрілі імена можна перевіряти, порівнюючи зі списком існуючих імен, якщо активізувати піктограму f(x) панелі інструментів.

Змінна в MathCAD може бути доступна тільки після того як вона буде визначена. Розрізняють глобальне і локальне присвоювання. **Розмірні змінні** характеризуються не тільки значенням, але і фізичною величиною. Застосування розмірних змінних дозволяє використовувати при фізичних розрахунках фактично відповідні фізичні розмірності. Також в MathCAD існують ранжовані змінні, які застосовуються для задання змінних, що приймають цілий ряд значень. У найпростішому випадку задання ранжованої змінної здійснюється так:

Name:=N begin...N end, або

Name:=N begin, (N begin + Step)...N end.

Такі змінні широко застосовуються для представлення, значень функцій у вигляді таблиць, а також для побудови їхніх графіків.

Знак рівності після будь-якого виразу, що містить змінні, ініціює таблицю виведення, у якій представлена задана змінна, але в таблицю можна вставляти свої дані, коректувати інші і т.д. Окрім вищезгаданих типів змінних в MathCAD існують також системні змінні, що мають заздалегідь задані початкові значення і можуть бути перевизначені користувачем. Однак, щоб уникнути помилок двоякого тлумачення таких змінних, це робити не рекомендується. Деякі системні змінні TOL, ORIGIN =0,1 мають принципові значення.

6. **Масиви** - об'єкти, що містять певну кількість даних, і характеризуються доступом до будь-якого значення (на відміну від змінних). Найпоширеніші одновимірні і двовимірні масиви – вектори і матриці. Введення елементів матриці здійснюється як $M [1;2:]$ і т.д. Але простіше вводити матриці з використанням операції Matrices у рядку меню Math по шаблону, що представляється.

4.3. Збереження і використання даних

Для збереження результатів роботи на магнітних носіях система MathCAD має спеціальний тип даних – файлові дані. Це ті ж вектори і матриці, що можуть записуватися у вигляді простих текстових файлів. Існує сім файлових операцій. Файли, що створюються і використовуються ними, легко переглянути будь-яким текстовим редактором, що сприймає тексти у вигляді ASCII-кодів. Розглянемо ці операції:

1. **READ** («ім'я файлу») – зчитує дані з файлу із зазначеним ім'ям і повертає значення вектор. Ім'я файлу – це строкова змінна, що записується в лапках. Звичайно вона використовується для присвоєння значень векторам. Наприклад: $V:=READ(DATA)$ – тут елементи вектора V одержують значення, з файлу з ім'ям DATA. Для вказівки повного імені варто вживати загальні для MS DOS складні імена (D:\EXPER\DATA, напр.).

2. **WRITE** – операція записує дані у файл і привласнює йому зазначене ім'я. Дані можуть породжуватися математичним виразом, наприклад: $WRITE («ім'я файлу»):=Вираз$.

3. **APPEND** - подібна операції WRITE, але вона дописує дані в кінець вже існуючого файлу. Не можна використовувати інші функції для дописування даних – вони знищують колишні дані у файл і заносять нові.

4. **READPRN** – подібна операції READ, але зчитує дані у вигляді двовимірного масиву – матриці. Функція READPRN повертає матрицю, значення елементів якої однозначно зв'язані зі значеннями елементів файлу.

5. **WRITEPRN** – подібна **WRITE**, але застосовується для запису матричного виразу (чи матриці) у файл із зазначеним ім'ям. Структура файлу подібна структурі матриці.

6. **APPENDPRN** – дописує дані у вже існуючий матричний файл. При роботі з векторами і матрицями яка має комплексні елементи, ці операції використовують розширені вектори і матриці елементів яких дійсні числа.

7. **READRGB** – операція відноситься до системних операцій. Використовується разом із графічними файлами і повертає масив з 3-х підмасивів, що несе дані про розкладання кольорового зображення на 3 основні кольори – червоний, зелений, синій. Операція виконує функції кольороподілу, що використовується в поліграфії для кольорового друку графіків і рисунків.

4.4. Виконання арифметичних операцій

Найпоширенішою областю застосування математичних систем є арифметичні операції.

У загальному випадку обчислюються математичні вирази, до складу яких можуть входити операнди з операторами і функції з параметрами. Наприклад, у виразі $Y = 2 * \ln(x) + 1$, Y – змінна, 1 і 2 – числові константи, * і + – оператори, $\ln(x)$ – вбудована функція з аргументом x .

Арифметичні оператори призначені для виконання арифметичних дій над чисельними величинами (операндами) і конструювання математичних виразів. Найпоширенішими є оператори арифметичних дій (+, -, * і /), піднесення до степеня ^, знаходження кореня певного степеня $\sqrt{\quad}$ та ін. MathCad оперує як з дійсними, так і комплексними величинами. Для роботи з комплексними величинами потрібно ввести визначення уявної одиниці i чи j (у вигляді $\sqrt{-1}$).

MathCad містить ряд розширених арифметичних операторів: обчислення суми (\$) і добутку (#) ряду величин, обчислення похідної (?) і визначеного інтеграла (&). Введення даних у дужках символів задає виведення шаблону оператора відповідної операції. Шаблон містить місця, що підлягають заповненню числовими чи символічними значеннями. Зрозуміло, для виклику шаблону можна використовувати і відповідну палітру математичних символів.

Робота з операторами відношення (логічними операторами).

Дуже часто в математичних розрахунках необхідно з'ясувати рівність чи нерівність величин. Для цього використовуються оператори відношення:

Вирази	Введення	Призначення оператора
$X > Y$	$X > Y$	X більше Y
$X < Y$	$X < Y$	X менше Y
$X @ Y$	$X \text{Ctrl}(Y)$	X більше чи дорівнює Y
$X @ Y$	$X \text{Ctrl} Y$	X менше чи дорівнює Y
$X @ Y$	$X \text{Ctrl} \# Y$	X не Y
$X = Y$	$X \text{Ctrl} = Y$	X дорівнює Y

Слід не плутати оператор порівняння (знак рівності) зі схожим знаком введення значень змінних. У MathCad знак рівності як оператор відношення має великий розмір і більш жирне написання, чим звичайні знаки (=) – оператор введення. Всі оператори відношення можуть вводитися самостійно замість розташування курсору. У цьому випадку з обох боків від них з'являються маленькі чорні прямокутники. Вони є місцем введення виразів, що підлягають порівнянню. Наприклад, якщо ввести знак “менше”, то на екрані з'явиться блок типу: $\square <$. Знак виведення при цьому буде з'являтися з повідомленням про помилку *Missing operand* (пропущений операнд).

Вирази з логічними операторами повертають логічне значення, що відповідає виконанню чи невиконанню умови, заданої оператором. Ці значення в системі MathCad є логічною одиницею 1, якщо умова виконана, і логічним нулем 0, якщо вона не виконана. Математичні значення логічних 1 і 0 збігаються зі значеннями числових констант 1 і 0. Наприклад:

$2 > 1 = 1$ – умова виконана, результат одиниця;

$1 > 2 = 0$ – умова не виконана, результат нуль.

Зазначена властивість операторів дозволяє будувати не зовсім звичайні вирази: $2 * (5 > 0) = 2$, вираз $5 > 0$ повертає одиницю, тому результат загального виразу дає число 2.

4.5. Обчислення та використання функцій

MathCad містить розширений набір вбудованих елементарних функцій. Вони задаються своїм ім'ям і значенням аргументу в круглих дужках. У відповідь на звертання до них функції повертають обчислені значення. Аргумент і значення функцій можуть бути дійсними і комплексними числами.

Тригонометричні функції: $\sin(z)$, $\cos(z)$, $\text{tg}(z)$, $\text{sec}(z)$, $\text{csc}(z)$, $\text{cot}(z)$.

Гіперболічні функції: $\sinh(z)$, $\cosh(z)$, $\text{tgh}(z)$, $\text{sech}(z)$, $\text{csch}(z)$, $\text{coth}(z)$.

Зворотні тригонометричні функції: $\text{asin}(z)$, $\text{acos}(z)$, $\text{atg}(z)$.

Зворотні гіперболічні функції: $\text{asinh}(z)$, $\text{acosh}(z)$, $\text{atgh}(z)$.

Показникові і логарифмічні функції: $\exp(z)$, $\ln(z)$, $\log(z)$.

Функції комплексного аргументу: $\text{Re}(z)$ – виділення дійсної частини z , $\text{Im}(z)$ – виділення уявної частини z , $\text{arg}(z)$ – обчислення аргументу (фази).

Обчислення спеціальних математичних функцій.

Поряд з елементарними, у системі MathCad міститься ряд вбудованих спеціальних математичних функцій. Їхнє застосування розширює можливості системи в розв'язанні складних математичних задач.

Функції Бесселя. Найважливішими зі спеціальних функцій, що є розв'язками диференціального рівняння 2-го порядку є такі функції Бесселя:

$J_0(x)$ - функції Бесселя першого роду;

$I_0(x)$ - модифіковані функції Бесселя;

$N_0(x)$ - функції Бесселя другого роду;

$K_0(x)$ - модифіковані функції Ганкеля;

а також гамма функція - $\Gamma(z)$ - для дійсного і для комплексного аргументу.

Окрім стандартних функцій, що входять до складу системи MathCad, користувач може створювати свої власні функції і задавати їх в такому вигляді:

Ім'я функції (список параметрів) = вирази.

4.6. Операції з векторами і матрицями

Серед основних операцій з векторами і матрицями слід виділити такі операції, які мають аналогічний математичний зміст та вводяться за допомогою відповідного шаблону:

$\frac{V}{z}; \frac{M}{z}$ - частка від елементами вектора та скаляра, а також від елементами матриці та скаляра відповідно;

M^{-1} - обчислення оберненої матриці;

M^n - обчислення матриці, кожний елемент якої подається в n -ому степені;

модуля вектора та детермінанта матриці відповідно;

$V^T; M^T$ - транспонування вектора та матриці відповідно;

$V1 * V2$ - обчислення скалярного добутку векторів;

Операції векторизації дозволяють застосовувати векторні операції до скалярів, а скалярні – до векторів $\cos(V)$; $V1 * V2$.

До вбудованих **векторних функцій** слід віднести такі функції:

$\text{length}(V)$ – повертає число елементів вектора;

$\text{last}(V)$ – повертає індекс останнього елемента;

$\max(V)$, $\min(V)$ – повертає максимальний чи мінімальний за значенням елемент;

$Re(V)$, $Im(V)$ – повертає вектор дійсної і уявної частин вектора відповідно.

До вбудованих матричних функцій слід віднести такі функції:

$augment(M1, M2)$ – поєднує в одну двох матриць $M1$ і $M2$;

$identify(n)$ – створює одиничну квадратну матрицю розміром n ;

$stack(M1, M2)$ – поєднує дві матриці $M1$ і $M2$. $M1$ зверху;

$diag(V)$ – створює діагональну матрицю, в якій елементи головної діагоналі збігаються з елементами вектора;

$matrix(m, n, f)$ – створює матрицю, в якій (i, j) -й елемент дорівнює $f(i, j)$;

$Re(M)$, $Im(M)$ – повертає вектор дійсної і уявної частин вектора.

Серед функцій, що повертають спеціальні характеристики матриць слід виділити такі:

$cols(M)$, $rows(M)$ – повертає кількість стовпців і рядків матриці M відповідно;

$Rank(M)$ – повертає ранг матриці M ;

$tr(M)$ – повертає слід (суму діагональних елементів) квадратної матриці;

$mean(M)$ – середнє значення елементів масиву;

$norm1(M)$, $norm2(M)$, $norml(M)$ – повертає норми матриць;

$eigenvals(M)$ – повертає вектор, що містить власні значення матриці M ;

$eigenvec(M, z)$ – для зазначеної матриці і заданого власного значення z повертає приналежний цьому власному значенню вектор.

Розглянемо функції сортування для векторів і матриць:

$sort(V)$ – сортування елементів векторів у порядку зростання їхнього значення;

$revers(V)$ – перестановка елементів (після $Sort$) векторів у зворотному порядку;

$csort(N, n)$, $rsort(N, n)$ – перестановка рядків матриці M таким чином, щоб відсортованим виявився n -й стовпець, n -й рядок.

Розв'язання систем лінійних рівнянь за допомогою матричних операцій можна здійснити таким чином. Якщо матриця A - це матриця коефіцієнтів, а вектор B - вектор вільних членів тоді матричне рівняння можна представити у вигляді:

$$A \cdot x = B.$$

Якщо використати операцію знаходження оберненої матриці, тоді можна знайти вектор невідомих змінних:

$$x := A^{-1}B.$$

В розв'язанні систем лінійних рівнянь також використовується вбудована функція $lsolve(A, B)$, $x := lsolve(A, B)$.

4.7. Статистична обробка даних

При виконанні фізичних експериментів дані, що отримуються, звичайно подаються з тією чи іншою випадковою похибкою. Тому їхня обробка потребує відповідних статистичних методів.

rnd (x) – функція генерації випадкових чисел на інтервалі $[0, x]$ з рівномірним розподілом;

corr (VX, VY) – коефіцієнт кореляції двох векторів – VX, VY ;

cvar (x, y) – коефіцієнт коваріації x і y .

Статистичні функції для векторів.

Наступна група функцій відноситься до обчислення основних статистичних параметрів одновимірного масиву даних – вектора.

Mean (V) – повертає середнє значення елементів вектора V ;

var (V) – повертає дисперсію для елементів вектора V ;

side (V) – повертає середньоквадратичну похибку, тобто квадратний корінь з дисперсії;

sidev (V) – задає стандартне відхилення елементів вектора V ;

hist (int, t) – повертає вектор частот потрапляння даних V у задані інтервали int (служить для побудови гістограми).

Функції обчислення щільності розподілу ймовірності

Функції обчислення щільності ймовірності розподілу представлені таким переліком:

dbinom (k, n, p) – біноміальний розподіл (повертає значення ймовірності p $P(x=k)$, де n і k – цілі числа, причому $0 \leq k \leq n$ та $0 \leq p \leq 1$;

dexp (x, r) – експоненціальний розподіл ($r, z > 0$);

dnorm (x, μ, σ) – нормальний розподіл (μ – середнє значення, $\sigma > 0$ – середньоквадратичне відхилення);

dpois (k, λ) – розподіл Пуассона ($\lambda > 0, k$ – ціле ≥ 0);

dunif (x, a, b) – рівномірний розподіл (a і b – граничні точки інтервалу ($a \leq x \leq b$)).

Функції розподілу

Такі функції дають ймовірність того, що випадкова величина буде мати значення менше або те, що дорівнює визначеній величині:

pbinom (k, n, p) – значення функції розподілу біноміального закону для k -успіхів у серії n -іспитів;

pxp (x, r) – значення в точці x функції експоненціального розподілу;

pnorm (x, μ, σ) – значення в точці x функції нормального розподілу;

ppois (k, λ) – значення для k функції розподілу Пуассона;

punif (x, a, b) – значення в точці x функції рівномірного розподілу.

Що стосується створення векторів з елементами, що мають різні закони розподілу, то в середовищі *MathCad* для цього існують такі функції:

rbinom (m, n, p) – біноміальний розподіл;

rexp (m, r) – експоненціальний розподіл;

rnorm (m, μ, σ) – нормальний розподіл;

rpois (m, λ) – розподіл Пуассона;

runif (m, a, b) – рівномірний розподіл.

4.8. Розв'язання нелінійних рівнянь і систем

Багато рівнянь, наприклад, трансцендентні, і системи з них не мають аналітичних розв'язків. Однак вони можуть розв'язуватися чисельними методами з заданою похибкою ($< TOL$).

Для рівнянь виду $F(x) = 0$ розв'язок знаходиться за допомогою функції:

Root (вирази, ім'я змінної).

Ця функція повертає значення змінної, при якому вираз дає 0. Функція розв'язує рівняння ітераційним методом і допускає задання початкового наближення змінної. Наприклад:

$F(x) := 2x^3 + 3x^2 + 4x + 5$ - задання полінома;

$x := 0; x_1 := \text{root}(F(x), x) \quad x_1 = \dots$

Для обчислення можливих комплексних коренів доцільно використовувати такі початкові присвоєння:

$i := \sqrt{-1}; x := 1 + i; \quad x_2 := \text{root}\left(\frac{F(x)}{(x - x_1)}, x_2\right) \quad x_2 = \dots,$

$x_3 := \text{root}\left(\frac{F(x)}{(x - x_1)(x - x_2)}, x_3\right) \quad x_3 = \dots$

Іншим ефективним способом обчислення коренів є обчислення за допомогою функції *Polyroots*. Для пошуку коренів звичайного полінома $R(x)$ ступеня n *MathCad* містить функцію *Polyroots* (V), що повертає вектор коренів багаточлена ступеня n , коефіцієнти якого знаходяться у векторі V , що має довжину $n + 1$. А саме:

$V_0 := 5; V_1 := 4; V_2 := 3; V_3 := 2 \quad \text{polyroots}(V) = \begin{bmatrix} \dots \\ \dots \\ \dots \end{bmatrix}$

Цікавий приклад використання функції *root* у складі функції користувача:

$F(a, x) := \text{root}(\sin x - a \cos x, x)$

$a := 1 \dots 5, \quad x_0 := 0, \quad x_a := F(a, x_{a-1})$

Розв'язання систем нелінійних рівнянь з використанням директиви Given.

Для розв'язання систем нелінійних рівнянь використовується спеціальний обчислювальний блок, що відкривається спеціальною директивою Given, яку необхідно вказати перед подальшими діями.

При розв'язанні з використанням цього обчислювального блоку необхідно вказати:

- 1) вирази, для яких знаходиться розв'язок;
- 2) обмежувальні умови;
- 3) функцію, що буде використовуватися для розв'язання;
- 4) також рекомендується розв'язання доповнити перевіркою.

В якості функції, що буде використовуватися для розв'язання можна використати $Find(V_1, V_2, \dots, V_n)$, що повертає значення однієї чи ряду змінних для точного розв'язку або функцію $Minerr(V_1, V_2, \dots, V_n)$, що повертає значення однієї чи ряду змінних для наближеного розв'язку.

Між функціями, що розглядаються, існує принципова різниця. Функція $Find$ використовується, коли розв'язок реально існує (хоча і не є аналітичним). Функція $Minerr$ намагається знайти максимальне наближення навіть до неіснуючого розв'язку шляхом мінімізації середньоквадратичної похибки.

Застосування вказаних функцій можна продемонструвати на прикладі знаходження точки перетину графіків функцій:

$$y = x^2 \text{ та } y = 8 + 3 \cdot x \text{ при обмеженнях } x < 0$$

Given

$$y = x^2 \text{ и } y = 8 + 3 \cdot x \quad x < 0$$

$$\begin{bmatrix} x0 \\ y0 \end{bmatrix} := Find(x, y) \quad \begin{bmatrix} x0 \\ y0 \end{bmatrix} = \begin{bmatrix} -1.702 \\ 2.895 \end{bmatrix}$$

Перевірка розв'язку

$$x0^2 = 2.895 \quad 8 + 3 \cdot x0 = 2.895.$$

4.9. Виконання прямого і оберненого перетворення Фур'є

Для реалізації прямого перетворення Фур'є використовуються дві функції: функція $fft(V)$, та функція $cff(A)$.

Функція $fft(V)$ реалізує пряме швидке перетворення Фур'є для даних, представлених дійсними числами - значеннями вихідного вектора V . Однак при цьому вектор V повинен мати 2^n складових, де n - ціле число. Якщо число складових інше, функція відмовляється від швидкого перетворення і виконує перетворення за повним алгоритмом.

Функція $cff(A)$ реалізує пряме перетворення Фур'є для вектора A з комплексними елементами. Якщо A - матриця, то здійснюється двовимірне перетворення.

Для реалізації оберненого перетворення Фур'є використовуються дві функції: функція $ifft(V)$, та функція $cifft(V)$.

Функція $ifft(V)$ - реалізує обернене перетворення Фур'є для вектора V з комплексними елементами. Вектор V повинен мати 2^{m+1} елементів. Функція повертає вектор D з дійсними елементами.

Функція $cifft(V)$ - виконує обернене перетворення Фур'є за повним алгоритмом, при якому і вихідний і вислідний вектори матриці містять комплексні числа. Якщо R - матриця, то здійснюється двовимірне перетворення. Наприклад:

$$f(x) := 5x^2 + 3x \quad V := f(x) \quad x := 1.0 \quad H_i := if(((x-1)/0,1+1 \leq k, G, 0) \\ x := 1, 1.1 \dots 3 \quad G := cfft(V) \quad Y_i := ifft(H_i)$$

4.10. Розв'язання диференціальних рівнянь

MathCad не має засобів для аналітичного розв'язання тих диференціальних рівнянь, які, наприклад, має *Mathematica* і *Maple*, і тому дозволяє знаходити розв'язки диференціальних рівнянь тільки в числовому вигляді. В цьому випадку використовуються такі функції:

Rkadapt (y, x_1, x_2, n, F, k, s) - повертає матрицю, що містить таблицю значень розв'язання задачі Коші на інтервалі x_1, x_2 для системи звичайних диференціальних рівнянь (ЗДР), обчислену методом Рунге-Кутта із змінним кроком і нелінійне рівняння у векторі Y (праві частини системи записані у векторі F), n - число кроків, k - максимальне число проміжних точок розв'язку, s - мінімально припустимий інтервал між точками;

Rkadapt (y, x_1, x_2, n, F) - інша форма застосування попередньої функції;

Rkfixed (y, x_1, x_2, n, F) - застосування попередньої функції з фіксованим кроком ітерацій.

Функція *Rkadapt* завдяки автоматичній зміні кроку розв'язання дає більш точний результат, але за швидкістю обчислень поступається функції *rkfixed*, застосування якої доцільне при розв'язках, що повільно змінюються. Наприклад:

$$\mu := -1 \quad X := \begin{bmatrix} 0 \\ 1 \end{bmatrix} \\ D(t, x) := \begin{bmatrix} \mu \cdot x_0 - x_1 - [(x_0)^2 + (x_1)^2] \cdot x_0 \\ \mu \cdot x_1 + x_0 - [(x_0)^2 + (x_1)^2] \cdot x_1 \end{bmatrix} \\ z := rkfixed(x, 0, 20, 100, d) \quad n := 0 \dots 99$$

Функції для розв'язання жорстких систем диференціальних рівнянь.

Для розв'язання жорстких систем диференціальних рівнянь використовується функція *Bulstoer* ($y, x_1, x_2, acc, F, k, s$), що повертає матрицю розв'язання системи ЗДР, права частина яких записана в символічному векторі F з заданими початковими умовами (ПУ) у векторі Y , на інтервалі x_1, x_2 . В цьому випадку використовується метод Буллер-Штоера зі змінним кроком:

$$\begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \dots & \dots & \dots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ \dots \\ \Delta x_n \end{bmatrix} = \begin{bmatrix} -f_1 \\ \dots \\ -f_n \end{bmatrix}$$

матрична форма представлення системи диференціальних рівнянь в частинних похідних.

stiffb (y, x_1, x_2, n, F, J) – повертає матрицю розв'язків твердого диференціального рівняння, записаного в F і функції Якобіана J , Y – вектор ПУ на інтервалі $[x_1, x_2]$;

stiffb ($y, x_1, x_2, n, F, J, k, S$) – повертає матрицю розв'язків тільки в кінцевій точці диференціального рівняння;

stiffv (y, x_1, x_2, n, F, J) – використовується метод Розенброка;

stiffv ($y, x_1, x_2, acc, n, F, J, k, S$) – використовується метод Розенброка із змінним кроком.

Функції для розв'язання диференціальних рівнянь у частинних похідних (Пуассона, Лапласа).

bvalfit ($V1, V2, x_1, x_2, x, F, L1, L2, score$) – встановлює ПУ для крайової задачі, заданої у векторах $F, V1$ и $V2$ на інтервалі $[x_1, x_2]$, де розв'язок відомий в деякій проміжній точці x .

multigrd (m, n) – повертає матрицю розв'язку рівняння Пуассона, в якій розв'язок дорівнює нулю на границях.

Функції для розв'язання крайових задач.

Для розв'язання двоточкових крайових задач призначені функції:

sval ($V, x_1, x_2, D, load, score$) та *svalfit* ($V1, V2, x_1, x_2, xf, load1, load2, score$).

Вектори $V, V1, V2$ задають ПУ, x, x_1, x_2 – граничні точки інтервалу розв'язків, $D(x, y)$ – функція, що повертає n -компонентний вектор з першими похідними невідомих функцій, $load(x_1, V)$,

$load(x_2, V2)$ – векторозначні функції поворотного значення ПУ. У точках (x_1, x_2) , $score(xf, y)$ векторозначна функція, що повертає h -елементний вектор відповідності, вказує наскільки значення розв'язків, що

починаються з точок x_1 та x_2 повинні відповідати xf . Якщо потрібне збігання розв'язків, то $\text{score}(xf, y) := y$.

4.11. Програмування в середовищі MathCad

Програмні оператори вперше з'явилися в MathCad 6.0 і починаючи з версії 7.0 представлені в розширеному варіанті. Всі оператори зосереджені в панелі програмування елементів і доступні для введення в робочий документ після звертання до вищезгаданої панелі.

Оператор Addline виконує функції розширення програмного блоку. Розширення фіксується подовженням вертикальної риски програмних блоків чи їхнім деревоподібним розширенням. Завдяки цьому можна створювати як завгодно великі програми.

Оператор внутрішнього переривання \leftarrow - виконує функції внутрішнього локального присвоювання $x \leftarrow 10$. Локальний характер присвоювання означає, що значення змінна зберігає тільки в тілі програми. Поза програмою модуля значення змінної може бути невизначене, має локальне присвоювання = або глобальне присвоювання \equiv .

Оператор створення умовних виразів if. Задається у вигляді: *вираз if умова*.

Якщо умова виконується, то повертається значення виразу. Разом з цим оператором часто використовуються оператор переривання *break* і оператор іншого вибору *otherwise*.

Оператор for служить для організації циклів із заданим числом повторень. Задається у вигляді:

For Var \in Nmin...Nmax...

Таким чином, якщо змінна *Var* змінюється з кроком +1 у діапазоні *Nmin* до *Nmax*, то вирази у шаблоні, будуть виконуватися. Змінну лічильника *Var* можна використовувати у виразах програми.

Оператор While служить для організації циклів, що діють доти, поки виконується деяка умова.

Записується у вигляді:

While умова

Умова, що виконується записується на місці шаблону.

Оператор otherwise – інакше, використовується з оператором *if*.

Наприклад:

$f(x) := \begin{cases} 1 & \text{if } x > 0 \\ -1 & \text{otherwise} \end{cases}$ повертає 1 якщо $x > 0$, повертає -1 у всіх інших

випадках.

Оператор break – викликає переривання роботи програми всякий раз, коли зустрічається. Найчастіше використовується разом з оператором

умовного виразу *if* та операторами циклів *While* і *For*, забезпечуючи перехід у кінець тіла циклу.

Оператор *continue* – використовується для продовження роботи після переривання програми. Також використовується разом з операторами задання циклів *While* і *For*, забезпечуючи після переривання повернення в початок циклу.

Оператор функція повернення *return* перериває виконання програми і повертає значення свого операнда, що стоїть слідом за ним:

Return 0 if x < 0, буде повертатися 0 при $x < 0$.

Оператор *on error* і функція *error*

Оператор обробки помилок дозволяє створювати конструкції оброблювачів помилок. Здається у вигляді:

вираз *1 on error вираз* *2.*

Якщо при виконанні *виразу* *1* виникає помилка, то виконується *вираз* *2*. Для обробки помилок корисна також функція *error(s)*, що при застосуванні в програмному модулі повертає віконце з написом, що зберігається в символьній змінній *S* чи символьній константі. Нижче наведемо приклади програмування з використанням представлених конструкцій:

$$\begin{array}{l}
 \text{Fact}(n) := \left\{ \begin{array}{l} f \leftarrow 1 \\ \text{while } n \leftarrow n - 1 \\ f \leftarrow f(n-1) \\ f \end{array} \right. \\
 \text{Fact}(3) = 6
 \end{array}
 \qquad
 \begin{array}{l}
 F_n(n) := \left\{ \begin{array}{l} f \leftarrow n \\ \text{while } 1 \\ f \leftarrow f(n-1) \\ n \leftarrow n - 1 \\ \text{break if } n = 1 \\ f \end{array} \right.
 \end{array}$$

$$f(x) := x^2 + 9 \quad x := 1 \quad R(f, x) := \text{root}(f(x), x) \quad R(f, x) = \text{помилка}$$

$$RF(f, x) := \left\{ \begin{array}{l} a \leftarrow x\sqrt{-1} \\ R(f, a) \text{ on error } R(f, x) \end{array} \right.$$

$$x := 1 \quad RF(f, x) = 3i$$

Нижче наведемо деякі корисні комбінації клавіш для обчислення виразів, які дуже часто зустрічаються при розв'язанні різних інженерних задач. Так обчислення границь функції *f(x)* при *x*, що наближається до *a* (тільки в символьному вигляді) здійснюється:

$\lim_{x \rightarrow a} f(x)$ - комбінація клавіш *[ctrl]L*,

границя ліворуч $\lim_{x \rightarrow a^-} f(x)$ - комбінація клавіш *[ctrl]B*,

границя праворуч $\lim_{x \rightarrow a^+} f(x)$ - комбінація клавіш *[ctrl]A*.

Обчислення невизначеного інтеграла від підінтегральної функції *f(t)* (у символьному вигляді):

$\int f(t)dt$ - комбінація клавіш *[ctrl]I*.

Обчислення визначеного інтеграла від підінтегральної функції $f(t)$ з межами інтегрування – нижньою a і верхньою b :

$$\int_a^b f(t) dt \text{ \& .}$$

Обчислення першої похідної функції $f(t)$ за змінною t :

$$\frac{d}{dt} f(t) \text{ ?}$$

Обчислення n -й похідної функції $f(t)$ за змінною t :

$$\frac{d^{n^{\ast}}}{dt^n} f(t) \text{ - комбінація клавіш [ctrl]+?}$$

5.1. Структура пакета Maple V 4 і основи роботи з ним

Якщо розглядати Maple V 4 з точки зору програмного продукту, то він складається з ядра – процедур, написаних на СІ і найвищою мірою оптимізованих, бібліотеки, написаної на Maple мові та інтерфейсу. Ядро виконує більшість базисних операцій. Бібліотека містить безліч команд – процедур, що виконуються у режимі інтерпретації. Програмуючи власні процедури, користувач може поповнювати ними стандартний набір, таким чином, розширювати можливості Maple.

Робота в Maple відбувається в режимі сесії (session) – користувач вводить речення (команди, вирази, процедури), що сприймаються й інтерпретуються Maple. Те, що при цьому з'являється на екрані дисплея, умовно розділяється на три частини: область введення (Input Region), що складається з речень, що набираються, область виведення (Output Region) і тексти коментарів (Text Region). Область виведення може включати результати виконання математичних і алгоритмічних операцій, графічні образи (двовимірна і тривимірна графіка). Область введення і виведення разом з відповідними коментарями називається групою (Group). Групи розділяються сепараторами.

Натискання клавіші Enter запускає виконання речення. Якщо введене закінчене речення, то йде виконання, інакше Maple очікує його завершення. Знайшовши помилку, Maple друкує в наступному рядку повідомлення про неї.

Результати роботи можуть бути збережені у файлах різних форматів. Існують два види Maple файлів. За замовчуванням усі результати роботи (області введення і виведення, коментарі) записуються у файл із розширенням 'ms'. Якщо задати режим збереження стану сесії (активний режим «Save Kernel»), то крім того, у файлі з розширенням 'm' будуть записані поточні призначення змінних і коди введених процедур. При записі у файли з іншими розширеннями зберігаються тільки області введення і тексти коментарів.

5.2. Вхідна мова програми Maple V 4.

Визначення мови можна розділити на символи, висловлення (лексеми), синтаксис і семантику (тлумачення).

Символи: 26 великих і малих літер латинського алфавіту, 10 цифр і 32 спеціальних символи.

Лексеми: до лексем відносяться ключові слова, оператори програмування, рядки, натуральні числа і розділові знаки.

Ключові слова – слова, що мають спеціальне значення, які не можна застосовувати як ідентифікатори. Наприклад, найменування функцій, команди Maple.

Оператори програмування. У Maple існує три види операторів, що відрізняються за кількістю операндів: двомісні (бінарні), одномісні (унарні), нульарні (не мають операндів). Таких операторів всього три (ditto - оператори), а саме оператори звертання до попереднього обчислення.

Синтаксис – кожна команда Maple повинна завершуватися роздільником ; або : . Якщо введення речення завершується роздільником ; то в рядку під реченням відразу буде відгук: результат виконання команди або повідомлення про помилку. Роздільник : використовується коли команди виконуються системою, а її результат не виводиться на екран.

У Maple використовуються круглі, квадратні і фігурні дужки. Призначення круглих дужок – обрамляти аргументи функцій і параметри в записі команд і задавати порядок при побудові математичних виразів. Квадратні дужки потрібні для роботи й індексних величин. Фігурні дужки використовуються для формування множин, наприклад система функцій.

Лапки: призначення подвійних лапок – це економія зусиль при звертанні до отриманих результатів обчислень. ") – позначається попереднє введення ""- пара відноситься до передостаннього результату і т.д.

Зворотні лапки (') вказують на рядок символів, причому двоє зворотних лапок підряд дозволяють включити самі лапки в рядок.

Застосування знака (') доцільно в двох випадках:

- для скасування присвоєння деякого значення змінної;
- для використання всередині команд з індексними параметрами (sum, product):

```
> sum ('i^ 2', 'i'=1..6);
```

Найпростіші об'єкти в Maple і в той же час найпростіші вирази це числа, рядки і ____ . Оскільки Maple оперує із символами, то числа не завжди виражаються в десятковому представленні. Раціональні числа при записі використовують оператор розподілу. При необхідності можливе представлення точних значень як раціональних чисел, та до звичайних десяткових в одному з видів:

```
> 2.3; > .143* 10^ (-44); > float (3141, -3); 3.141
```

5.3. Типи об'єктів у Maple. Аналіз структури об'єктів

Кожному виразу (і іншому об'єкту) в Maple відповідає зв'язаний з ним тип об'єкта. Базисними типами об'єктів для виразів є string, integer, float, fraction, арифметичні оператори '+', '*', '^' і function. Для визначення типу об'єкта використовується команда Whattype:

```
> whatype (15/37); fraction
> whatype ((x+3) * (y-4)); *
```

Хоча завжди можна зробити запит Maple про тип об'єкта, задалегідь задати тип об'єкта неможливо. Оскільки іноді можливий спектр обчислень визначається типом змінної, то за допомогою команди Type можна зробити запит, що дозволяє визначити її тип.

```
> colour := 'red';
> colour;
red
> type (colour, string);
true
```

Існують ще дві корисні команди для аналізу структури об'єктів – команда hastype, що повідомляє, чи містить об'єкт підоб'єкт даного типу, і команда has, що повідомляє чи міститься визначений підоб'єкт в об'єкті.

```
> hastype (x^2+3*x+5, '*'); true
> has(x^2+3*x+5, 3); true
> has(x^2+3*x+5, 2*x); false
```

Хоча ці приклади досить очевидні, але при роботі з великими об'єктами ці процедури корисні.

Кожен об'єкт Maple складається з підоб'єктів відомого типу, що також складаються з менших підоб'єктів, аж до елементарних об'єктів.

Засоби Maple дозволяють досліджувати і діставати індивідуальні елементи, що складають об'єкт. Ця можливість корисна при оперуванні з великими об'єктами і реалізується командами op і pops.

Команда pops повідомляє, скільки підоб'єктів першого рівня знаходяться в об'єкті, а команда op використовується, щоб відобразити ці підоб'єкти у вигляді послідовності виразів:

```
> object := 3 * x^2 + 2 * x - 3; object := 3x^2 + 2x - 3
> pops(object); 3 > op(object); 3x^2, 2x, -3
```

Команда op може використовуватися, щоб витягти індивідуальні елементи з об'єкта, а використовувана рекурентно може дістатися ще глибше:

```
> op(1, object); 3x^2, > op(1, op(1, object)); 3; op(nops(object), object);
-3
```

Якщо досліджуваний об'єкт – індексна змінна (iname), то команда pops (iname) повертає число індексів, op (i, iname) повертає i-ий індекс, а op(0, iname) – повертає ім'я індексної змінної.

```
> pops(A [i, j]); 2 > op(1, A [i, j]); i, > op(0, A [i] [j]) A,
```

Крім констант, які можна вводити у вигляді цілих, раціональних чисел з плаваючою точкою, Maple містить такі математичні константи:

```
Pi, exp(1), 1, infinity, -infinity, _____, true, false
```

рядки →

Елементарні вирази в Maple можуть утворювати більш складні об'єкти послідовності виразів, відділені комами.

Оскільки більшість команд вимагають введення параметрів у вигляді послідовності виразів, а інші навпаки представляють результат у вигляді послідовності виразів, то такий тип даних є досить важливим.

> 1, 2, 3, 4, 5; > a + b, c + d, e + f

Для спрощення створення послідовності виразів використовується оператор \$ один або разом з оператором діапазону:

> a \$ 6; a, a, a, a, a, a

> 2^i \$ 1..3; 2, 4, 8

> seq(i! / i^2, i=1..4); 1, $\frac{1}{2}$, $\frac{2}{3}$, $\frac{3}{2}$

5.4. Функції в Maple

Найбільш загальний метод задання функції в Maple, що вживається найчастіше, – це визначення процедури, зокрема задання функціональних операторів. Функціональний оператор задає функцію або послідовність функцій від однієї або кількох змінних і записується у вигляді:

(послідовність змінних) → (послідовність виразів)

> F1:=(x, t) → x^2+t^2; F1:=(x, t) → x^2+t^2

> F1(y, tau); y^2+tau^2

Інший спосіб задання функціонального оператора – використання команди unapply

> F1:=unapply(x^2+t^2, x, t); F1:=(x, t) → x^2+t^2

> F1(1, 2); 3

5.5. Оператори в Maple

Крім згаданих арифметичних, логічних операторів, операторів ditto та функціонального в Maple є ще багато інших.

1. Оператор композиції @. Застосовується для створення складної функції і записується у вигляді:

• f@g - для створення композиції функції f і g чи

• f@@n - для n-кратного застосування функції f

> (ln @ sin)(x); ln(sin(x))

> f:=x→1/(1+x); (f@@5)(x); f:=x

2. Нейтральний оператор. Нейтральні оператори визначаються користувачем. Ім'я нейтрального оператора повинне починатися із символу & і може супроводжуватися допустимим в Maple ім'ям за винятком синтаксичних роздільників.

- define (a a (oper)) – oper – ім'я оператора, aa – ім'я абстрактного алгебраїчного об'єкта
- define (oper; property 1, property 2, ..)

Ці команди визначають правила обчислення і спрощення оператора

> define (linear ('& L'))

> & L (5*x+3*y) 5&L(x)+ 3&L(y)

Якщо наступний об'єкт функція-процедура (Fname (x1, x2)), то команда pops повертає число аргументів цієї функції, команда op (i, Fname (x1, x2, ..)) повертає i-ий аргумент, а команда op (0, Fname (x1, x2, ..)) – ім'я функції.

> pops (F (x, y, z)); 3 > op (1..3, F(x, y, z)); x, y, z.

5.6. Команди перетворення виразів, типів, оцінювання

Команди Maple використовують такі вирази Maple як: списки, набори й інші об'єкти Maple.

Maple має більш ніж 2500 команд, що зберігаються в ядрі Maple, основній бібліотеці і спеціалізованих пакетах. Більшість команд Maple записані цілком у символах нижнього регістра. Оскільки Maple «чутливий до регістра», то, по-перше, вирази “diff” і “Diff” різні, а, по-друге, незалежно від того чи використовуються вони як імена змінних чи як команди, Maple завжди розглядає їх як різні об'єкти. Імена команд Maple захищені, тому використання їх як ідентифікаторів призведе до помилки.

Кожна команда Maple використовує таку послідовність параметрів як аргумент > ім'я команди (параметр 1, параметр 2, ...);

Незалежно від того скільки параметрів задано, параметри команди завжди закладені в круглі дужки (). Інші типи дужок не будуть приводити до інтерпретації записаного виразу як команди. Деякі команди мають обмеження на тип об'єктів, що вони застосовують для введення, а також послідовність, і мінімальна кількість параметрів з якої вони можуть викликатися. У той же час більшість команд можуть використовувати більше параметрів, ніж їхнє мінімальне число. Ці екстра-параметри можуть включати велику кількість додаткових можливостей, у тому числі, опції, що керують функціонуванням команди. Наприклад:

> Diff (3*x^2+2*x-6, x); $\frac{d}{dx} (3x^2 + 2x - 6)$

> diff (3*x^2+2*x-6, x); 6

Після завантаження Maple жодна з команд не є цілком завантаженою в пам'ять. Однак велика кількість стандартних команд мають покажчик їхнього перебування при завантаженні, і при звертанні до них відбувається автоматичне завантаження. Інші команди, що постійно знаходяться в основній бібліотеці, автоматично не завантажуються, а повинні на початку

явно завантажуватися командою `readlib`. Один раз завантажену команду немає необхідності перезавантажувати протягом поточного сеансу.

5.7. Команди в спеціалізованих пакетах

Maple містить кілька десятків спеціалізованих наборів команд, що називаються пакетами (`linalg`, `geometry`, `finance`).

Підпрограми в цих пакетах не завантажуються автоматично, і не можуть ініціалізуватися командою `readlib`. Один зі способів виклику цих команд полягає у використанні команди `with` (ім'я пакета) для завантаження покажчиків до всіх команд пакета. Інший спосіб полягає в тому, щоб перед назвою команди додавати назву пакета, а саму команду взяти в квадратні дужки:

> ім'я пакета [ім'я команди] ();

Перетворення виразів

Оскільки Maple це пакет символічного перетворення, то команди перетворення виразів (спрощення, підстановки, приведення до потрібного виду) грають досить важливу роль. Розглянемо найбільш важливі з них:

`factor (w)` – операція факторизації (вирази представляються у вигляді множників);

`expand (w)` – розкриття дужок;

`simplify (w)` – спрощення виразів;

`subs (old=new, expr)` – підстановка виразів `new` замість `old`, у виразі `expr`;

`denom (rat)` і `numer (rat)` – виділення знаменника і чисельника раціонального дробу;

`lhs (eq)` і `rhs (eq)` – виділення лівої і правої частини виразу.

(`<opt1, opt2 ... opt3 >`) відповідно до заданих опцій `exp`, `ln`, `polar`, `radical`, `sqrt`, `trig` і `in`. при `trig` спрощення буде використовувати багато тригонометричних співвідношень. В якості опції можуть задаватися співвідношення у вигляді рівностей.

Перетворення типів

Крім відомих команд роботи з типами виразів `type`, `whattype`, існує команда перетворення типів виразів: `convert (expr, option)` `option` – велика кількість для перерахування, `convert (list, array)`, `convert (list, vector)`, `convert (list1, list2 ... listn, matrix)`, `convert (list, poly)`, `convert (poly, list)`...

Операції оцінювання.

Оскільки тільки для змінної скалярного типу виводяться її значення, то для змінних складних типів рядок введення викликає область виведення з ім'ям змінної. Для перегляду значення змінної існують команди:

`eval (array)` – видача вмісту масиву `array`;

`evalf (expr)` – видача раціонального виразу у вигляді числа з плаваючою точкою;

`evalc (expr)` – обчислення комплексного виразу;

`evalm (expr)` – обчислення матричного виразу.

При роботі з дробовими числами корисні такі команди:

`frac (expr)` – обчислення дробової частини дійсного виразу;

`round (expr)` – округлення дійсного виразу `expr`;

`trunc (expr)` – обчислення цілої частини виразу `expr`.

При роботі з комплексними виразами:

`conjugate (cmplx)` – обчислення комплексної сполученої величини для `cmplx`;

`Im (cmplx)`, `Re (cmplx)` – визначення уявної і дійсної частини `cmplx`;

`polar (cmplx)` – перетворення `cmplx` у полярні координати.

Операції з поліномами

Для всіх пакетів аналітичних обчислень операції з поліномами є базовими і часто використовуються для інших перетворень у формулах. Під поліномом у Maple розуміється сума виразів з невід'ємними степенями. Так, що поліномами є константа, змінна, вираз. Поліноми бувають однієї чи декількох змінних:

`coeff (poly, var, int)` – обчислення коефіцієнта при n -му степені змінної `var` для полінома `poly`;

`coeff (poly, var)` – обчислення коефіцієнтів полінома `poly` при змінній (змінних) `var`;

`degree (poly, var)` – обчислення степеня полінома `poly` за змінною `var`;

`quo (poly1, poly2, var)` – обчислення частки від ділення двох поліномів;

`_ (poly1, poly2, var)` – обчислення залишку від ділення двох поліномів.

Дві корисні команди для роботи з поліномами (завантажуються за допомогою команди `readlib`)

`root (poly, n)` – знаходження кореня n -го степеня з полінома;

`psqrt (poly)` – знаходження квадратного кореня з полінома.

5.8. Розв'язання рівнянь і нерівностей

Для аналітичного розв'язання алгебраїчних рівнянь використовується команда: `solve (eqns, vars)`, `eqns` – рівняння чи система рівнянь, `vars` – змінна чи група змінних у фігурних дужках. Система рівнянь задається у вигляді множини, де множиною є сукупність розділених комами об'єктів, взята у фігурні дужки.

Рівняння можуть бути задані безпосередньо в тілі команди, а можуть бути введені прирівнюванням виразів деякій змінній `eqns`. Якщо в якості першого параметра команди `solve` ввести вираз, то прирівнювання цього

виразу до нуля проводяться автоматично. Для збереження розв'язків зручно ввести змінну і звертатися до конкретного розв'язку по індексу:

```
> sols:= solve (x^3+x=0, x);  
sols:= 0, 1, -1
```

```
> sols[2];
```

Для присвоєння знайдених значень змінних, щодо яких шукався розв'язок, застосовується команда assign. Для скасування призначення змінних застосовується команда unassign чи “.”.

```
> s:= solve ({a*x+b*y=2, 2*x+3*y=a}, {x, y});  
s:= {y = _____, x = _____}
```

```
> assign (s); x; y;
```

При достатній кількості часу і пам'яті Maple завжди відшукає розв'язок системи лінійних рівнянь. Для нелінійних рівнянь може бути знайдено кілька розв'язків і може виявитися, що розв'язок не знайдений. Тоді Maple просто видасть запит на введення, очікуючи нової команди.

Якщо не вказані всі змінні, щодо яких повинен бути знайдений розв'язок, то відповідь буде видана для всіх змінних.

```
> solve (a* x+b=0):  
{b=-ax, x=x, a=a}
```

Якщо у виразі відповіді з'явилася функція Root of, це означає, що Maple не може виразити корені в радикалах, і розв'язок виражається через корені аргументу цієї функції.

Конструкція Root of (a(x)=0, x) використовується для позначення будь-якого кореня рівняння a(x)=0. Розширений формат команди дозволяє вказати той корінь з безлічі коренів, що знаходиться поблизу значення z:

```
Root of (a(x)=x, x, z).
```

```
> y:=root of (x^4-16=0, x, -1.7*I); evalf (y);  
y:= root of (-z^4-16, -1.7I)  
-2.000000 I
```

Для чисельного розв'язку системи рівнянь eqns щодо змінних vars використовується команда fsolve:

```
fsolve (eqns, vars, option);
```

За допомогою параметрів options можуть бути задані додаткові умови, які встановлюють, що: complex – розшукуються комплексні корені; a..b – для пошуку коренів заданий інтервал [a..b]; maxsols=n – визначене число розшукуваних розв'язків; fulldigits- використовується арифметика з максимальною мантисою.

5.9. Команди реалізації операцій математичного аналізу

Дослідження будь-якої функції починається із знаходження області визначення функції. Цю операцію важко реалізувати автоматично, однак

розв'язання цього питання часто пов'язано з розв'язанням нерівностей, що накладають обмеження на припустимі значення аргументу функції. Ця задача може бути успішно розв'язана засобами Maple наступним накладенням умов на аргумент досліджуваної функції командою

> assume (a>0); логічний вираз,

> solve (x²+x>5, x);

$\sqrt{x^2+x-5}$ > assume (x=')

Наступною операцією матаналізу, що застосовується для дослідження функції, є обчислення границь:

limit (expr, x=val, dir); limit (expr, x=val, dir);

expr x=val dir

left (), right () real

$\lim_{n \rightarrow \infty} \frac{n \sin(n!)}{n^2+1}$

> limit (n * sin (n!)/(n^2+1), n=infinity; value (''))

$\lim_{n \rightarrow \infty} \frac{n \sin(n!)}{n^2+1} = 0$

Для дослідження екстремумів функції як однієї, так і багатьох змінних використовується команда extrema (expr, constr, vars, nv), expr – вираз екстремуму, який потрібно знайти, constr - обмеження, vars - змінна, за якою розшукується екстремум, а nv – ім'я змінної, якій будуть присвоєні координати точок екстремуму. Перед звертанням до функції extrema її необхідно викликати зі стандартної бібліотеки командою readlib.

Знайти екстремум функції $y = \arctan(x) - \frac{1}{2} \ln(1+x^2)$,

>readlib (extrema);

>extrema (y = arctan(x) - ln(1 + x^2)/2, {}, x, 's');s;

$\left\{ \frac{1}{4} \pi - \frac{1}{2} \ln(2) \right\} \quad \{ \{x=1\} \}$

Для пошуку мінімуму і максимуму функції використовуються відповідні команди:

minimize (expr, vars, rangs), maximize (expr, vars, rangs):expr – вираз, vars - змінна, за якою шукається мінімум чи максимум, а rangs – інтервал зміни змінних, може використовуватися рядок 'infinite', тобто max чи min буде розшукуватися на всій числовій осі.

Перевірити безперервність алгебраїчного виразу, що залежить від змінної x, на відрізку [x1,x2] можна за допомогою команди iscont (expr, x=x1,x2); результатом буде логічна константа (true) чи (false). Для визначення точок, у яких порушується безперервність виразу expr за змінною x, використовується команда discont (expr, x).

Для знаходження точок розриву функції використовується команда

Singular (expr, vars), expr – вираз, що залежить від змінної vars, причому expr може залежати від декількох змінних:

>readlib (singular) :singular(*fan(x/(x-y)),x*);

$$\left\{ x = -\frac{-2_N\pi y - \pi y}{-2 + 2_N\pi y + \pi} \right\}, \{x-y\}, \text{де } _N - \text{будь-яке ціле число.}$$

5.10. Диференціювання та інтегрування в Maple

Обчислення похідної виразу *expr* за змінною *x* здійснюється за допомогою команд *diff (expr,x)*, *Diff (expr,x)*. Ці команди можуть бути використані для обчислення власних похідних функцій багатьох змінних, тоді використовується формат:

diff (expr,x1\$n1,x2\$n2,...).

Тут *expr* – вираз, що залежить від змінних *x1,x2...; n1,n2* – порядки диференціювання за попередніми змінними.

Для інтегрування використовується процедура *int (expr,ras)*, *Int (expr,ras)*. У такому форматі ці команди використовуються для знаходження невизначених інтегралів. Для визначених інтегралів використовується конструкція виду: *int (expr,var -a..b)*, *expr* – інтегральний вираз за змінною *var* на відрізку інтегрування *[a,b]*. Для знаходження подвійних, потрійних інтегралів необхідно застосувати цю команду кілька разів.

5.11. Команди для розв'язання задач лінійної алгебри

Основними об'єктами команд лінійної алгебри є матриці і вектори. Матрицею в MAPLE вважається двовимірний масив, у якого індекси змінюються від одиниці. Так що, наприклад, змінна описана як *array (0..3, 1..4)*, матрицею не вважається. Для задання матриць використовується два способи: за допомогою команд *matrix* і *array*, а саме:

>a:=array (1..3, [])

>a:=matrix (n,m,[val1,val2,..]), *n* – число рядків, *m* – стовпців, *val1,val2* – значення елементів матриці.

matrix (n,m,f), *f*-функція двох цілих змінних індексів матриці, за допомогою яких присвоюється значення елементам матриці.

Вектором у MAPLE вважається одновимірний масив, який можна визначити за допомогою опису:

array (1..n, [val1,val2,..]), *n* – розмірність вектора. Інший спосіб описання вектора реалізує команда *vector* з пакета *linalg*:

vector (n,[val1,val2,..]); або *vector (n,f)*; *f* – функція від індексу, що використовується для генерації вектора.

Робота зі структурою матриці і вектора

Для з'ясування розмірності матриці M існує дві команди:

`rowdim (M)` – видає число рядків матриці M .

`coldim (M)` – видає число стовпців.

Число елементів вектора можна визначити за допомогою `vectdim (M)`.

Для зміни розмірності матриці (тобто для видалення або додавання рядків чи стовпців) використовуються команди:

`delcols (M,i..j)` – видаляє рядки з номерами $i..j$.

`delrows (M,i..j)` – видаляє стовпці.

`extend (M,m,n,x)` – m – число рядків, що додаються, n – стовпців.

`copyinto (A;B,i,j)` – матриця A копіюється в матрицю B , починаючи з елемента i,j .

Для горизонтального злиття матриць використовується команда `concat(M1,M2)`, для вертикального – команда `stack (A, Y)`. Щоб виділити (без видалення) з матриці M рядок з номером i , використовується команда `row(M,i)`, для виділення стовпця – `col (M, i)`.

Для виділення підматриці, що включає елементи стовпців з номерами від $i1$ до $i2$ і рядків з номерами від $j1$ до $j2$, використовується команда

`submatrix (M, i1..i2,j1..j2)`, для вектора - `subvector (V, i..j)`.

Виділення мінора матриці M для елемента з індексами i,j здійснюється командою `minor (M, i, j)`.

5.12. Основні матричні і векторні операції

До основних матричних та векторних операцій слід віднести:

`add (A,B)` – додавання матриць або `evalm (A+Y)`;

`multiply (A, B)` – добуток матриць або `evalm (A*Y)`;

`evalm (M^n)` – піднесення матриці M до степеня n ;

`inverse (M)` чи `evalm (1/M)` – обчислення зворотної матриці

`transpose (M)` – транспонування матриці;

`det (M)` – обчислення визначника матриці з пакета `(linalg)`

`rank (M)` – обчислення рангу матриці

`eigenvals (M, vect)` – пошук власних чисел і власних векторів.

Основними векторними операціями є:

`angle (V,U)` – визначення кута між векторами V, U .

`basis ([V1,V2,...Vk])` – визначення базису для векторів $V1,V2,...Vk$.

Матричні рівняння можна розв'язувати використовуючи процедури:

`linsolve (M, B)` і `leastsqrs (M, B)` – розв'язання за методом найменших квадратів.

Векторний аналіз

dotprod (U, V) - скалярний добуток векторів U і V.

crossprod (U, V) – векторний добуток векторів U і V.

norm (M) – обчислення норми матриці.

grad (f,x) – обчислення градієнта скалярної функції за аргументом x.

diverge (F, x) – обчислення дивергенції.

curl (V, x) – обчислення ротора тривимірного вектора V за трьома змінними x.

laplacian (f, X) – обчислення Лапласіана функції f, за змінними X.

jacobian (V, X) - обчислення матриці Якобі для вектора V за X.

Змінні X задаються у вигляді [x,y,z]; як опціональний параметр використовується конструкція coords=cylindrical.

5.13. Команди для розв'язання диференціальних рівнянь

Для знаходження аналітичних розв'язків диференціальних рівнянь використовується команда:

dsolve (eqns, vars, option), де eqns – одне диференціальне рівняння або система рівнянь щодо невідомих функцій vars, option – додаткові умови, що дозволяють вказати метод розв'язання задачі (наприклад, type=numeric – для чисельного розв'язку).

За замовчуванням прийнято type=exact; тому MAPLE намагається знайти розв'язок в аналітичному вигляді. Отримуваним розв'язком буде містити невизначені константи _31, _32.., або параметр _T у випадку, коли розв'язок видається в неявному вигляді. Щоб одержати розв'язок в явному вигляді, для функцій vars необхідно вказати опцію type=explicit.

Для задачі Коші в рівняння eqns потрібно включити також початкові умови, а для крайової задачі – відповідно крайові умови. Якщо MAPLE може знайти розв'язок, але число крайових чи початкових умов менше порядку системи, то у відповіді будуть присутні невизначені константи. Команди

```
>deqn:=diff (y(x),x$2)+3*dyf(y(x)x)+z*y(x).
```

```
>bvp:=y(0)=0, y(1)=1;
```

```
>dsolve ({deqn, bvp}, y(x) );
```

$$\frac{e^{(-2,x)}}{e^{(-2)} - e^{(-1)}} - \frac{e^{(-x)}}{e^{(-2)} - e^{(-1)}}$$

Для розв'язання нелінійних задач, що не піддаються аналітичним методам, використовується конструкція команди dsolve з опцією series для пошуку розв'язку у вигляді розкладання в ряд і опцію tplsace для застосування методу перетворення Лапласа. При використанні цих опцій задається значення перемінної >Order:=4, що задає кількість членів розкладання.

Для числового розв'язку диференціального рівняння використовується опція numeric, що обчислює розв'язок диференціального рівняння в точці. А саме:

```
>F:=dsolve({deqn,bvp},y(x),type=numeric, option);  
>x=1.5;  
>F(DC);
```

В якості опцій в попередній конструкції можуть використовуватися параметри, що задають метод розв'язання:

method=2kf45 – метод Р.К. Фельберга, порядку 4-5.

method=dverk78 – метод Р.К. 7-8 порядку.

5.14. Графіка в MAPLE

Майже всі опції двовимірної графіки, можуть бути застосовані для всіх розглянутих команд і дозволяють керувати виглядом зображення: детальністю графіка, типом ліній, що виводяться і заповнювачів, розміщенням написів і т.д. Опції йдуть відразу за обов'язковими параметрами в командах, причому у випадку їхньої відсутності використовуються установки за замовчуванням. Перелічимо найбільш важливі з них:

acords = polar - тип координат (полярні чи декартові).

axes = val – тип виведених осей координат, val→normal звичайні осі,

boxed – графік, що береться в рамку, Frame – осі з центром у лівому нижньому кутку.

scaling= val – тип масштабу рисунка. Якщо val→CONSTRA, то графік виводиться з однаковим масштабом по осях координат,

UNCONSTRAINED– графік масштабується по розмірах графічного вікна.

style = LINE/POINT – виведення лініями або крапками.

numpoints=n – число точок графіка, що обчислюються (n =49)

color= black за замовчуванням

xtickmarks=nx – число насічок по осі абсцис (x)

ytickmarks=ny – число насічок по осі ординат (y)

thickness=n – товщина лінії, $n = 1, 2, 3, \dots$

linestyle=n – тип виведеної лінії (наприклад, пунктир)

symbol=n – тип символу, яким виводиться – точка

font=[vfont,style,size] – задання шрифту для виведення тексту

vfont – ім'я шрифту (times,courier,helvetica чи symbol), style – стиль шрифту, size – розмір символу.

Команди двовимірної графіки.

Основною командою двовимірної графіки є команда *plot*, що знаходиться в стандартній бібліотеці і не вимагає попереднього виклику. В залежності від представлення вхідних параметрів вона дозволяє рисувати графіки функцій однієї змінної, параметрично заданої функції, набору точок і т.д.

plot ({func1, func2,...}, x=a...b, y=c..d<options>); - команда побудови графіків функцій однієї змінної, заданої явною формулою, a..b – інтервал зміни x, c..d – виведений інтервал на осі ординат.

Для виведення параметрично заданої кривої використовуються формули:

plot ([funcx(t), funcy(t), t=a..b], <options>),
funcx(t), funcy(t) – функції координат, що залежать від параметрів.

Для зображення набору точок використовується команда:

plot ([x1,y1,x2,y2,...], x=a..b, y=c..d,<options>), де [x1,y1,x2,y2,...] – набір точок.

logplot (exp2,var1=a..b,<options>) – побудова графіка виразу в логарифмічному масштабі по осі Y.

loglogplot (exp2,var1=a..b,<options>) – побудова графіка exp2 у логарифмічному масштабі по осі X та по осі Y.

textplot ([exp2x,exp2y,string],<options>) – виведення текстового рядка string, починаючи з точки з координатами exp2x,exp2y.

polarplot ([rad, ang,var=a..b],<options>) – виведення графіка в полярних координатах, де функція радіуса rad і кута ang залежать від змінної var, що змінюється на відрізку [a,b].

fieldplot ([expr1,expr2], x=a..b,y=c..d,<options>) – побудова векторного поля, обумовленого виразами expr1,expr2; змінні x та y змінюються на відрізках a..b, c..d.

gradplot (expr,var1=a..b,var2=c..d,<options>) – зображення векторного поля, яке задається градієнтом виразу expr, що обчислюється за змінними var1 і var2.

odeplot (S, vars, r1, r2, <options>) – зображення розв'язку диференціального рівняння.

Часто буває необхідним сполучити на одному рисунку графічні образи, отримані за допомогою різних графічних команд. Для цього результат дії кожної команди повинен бути привласнений деякій змінній. При цьому виведення на екран не відбувається:

Для зображення отриманих графічних образів служить команда: `display ([pic1,pic2,..], <options>)`, де образи `pic1,pic2,..` виводяться на одному рисунку в загальних осях координат.

При вказанні опції `insequence=true` образи `pic1, pic2,..` будуть спливати один за одним, складаючи кадри мультфільму:

`animate (F, x=a..b, t=c..d, frame=n)`. Тут `F` – вираз, що залежить від змінних `x` і `t`, що змінюються відповідно в інтервалах `[a..b]`, `[c..d]`. Змінна `x` відповідає осі абсцис, `t` – змінна часу, опція `frame` задає число кадрів.

Двовимірні графічні структури

`POINTS ([X1,Y1], [X2,Y2],...)` – набір точок з координатами.

`CURVE ([[x11,y11],..[x1n,y1n]], [[x21,y21],.. [x2n,y2n],..[[xmn,ymn]])` `m` наборів по `n` точок, що визначають незамкнуті криві. Точки кожного з наборів будуть послідовно з'єднані відрізками.

`POLIGONS (.....)` `m` наборів по `n` точок, що визначають замкнуті криві.

`TEXT ([x,y], 'string', horizontal, vertical)` – графічний виведення тексту, починаючи з позиції, обумовленої типом горизонтального і вертикального вирівнювання (`alignright`, `alignleft`, `alignabove`, `alignbelow`).

Для виведення графічних структур `str1,..strn` на екран, використовується команда `PLOT ({str1,..,strn})`;

Контрольні запитання

1. Які є ППП для автоматизації математичних обчислень?
2. Які є спільні та відмінні риси у інтерфейсі користувача у ППП "MathCad", "Maple"?
3. Як об'єкти, види змінних та вирази використовуються у зазначених вище ППП?
4. Які існують стандартні функції у зазначених вище ППП?
5. Які можливі аналітичні перетворення у ППП "MathCad", "Maple"?
6. Як обчислити похідну від заданого виразу?
7. Як обчислити первісну?
8. Як визначити границю функції?
9. Як розкласти вираз на множники?
10. Як розкласти функцію у ряд Тейлора.
11. Як обчислити градієнт функції?
12. Як ввести вектор, матрицю?
13. Як обчислити скалярний, векторний добуток векторів?
14. Як обчислити слід матриці, обернену матрицю?
15. Як обчислити власні числа та власні вектори матриці?
16. Як обчислити визначник матриці?
17. Як розв'язати лінійне алгебраїчне рівняння?
18. Як розв'язати систему лінійних алгебраїчних рівнянь?
19. Як розв'язати диференціальне нелінійне рівняння?
20. Як розв'язати систему диференціальних рівнянь?
21. Які умовні оператори та оператори циклу ППП "MathCad", "Maple", вам відомі?
22. Які команди двовимірної графіки із зазначених вище ППП вам відомі?
23. Які опції команд тривимірної графіки ППП "MathCad", "Maple", вам відомі?

ЛИТЕРАТУРА

1. Хоменко А.Д. MS Word 97. - СПб.: BHV- Санкт-Петербург, 1998. - 1056 с.
2. Персон Р. Excel 97 в подлиннике. - СПб.: BHV-Санкт-Петербург, 1997. - 1312 с.
3. Microsoft Office в целом. - СПб.: BHV- Санкт-Петербург, 1997. - 336 с.
4. Говорухин В.Н., Цибулин В.Г. Введение в Maple. Математический пакет для всех.-М.1997.-208 с.
5. Дьяконов В.П., Абраменков И.В., MathCad 8.0 в математике, физике и в Internet.- М.: "Нолидж", 1998.-352 с.
6. Дьяконов В.П. Справочник по MathCAD PLUS 7.0 PRO - М.: СК Пресс, 1998.-352 с.
7. Дьяконов В.П. Справочник по применению системы PC MatLAB.- М.: Наука, 1993.-112 с.
8. В.М.Дубовий, Р.Н.Кветний Основи застосування ЕОМ в інженерній діяльності. - К.: ІСД МО України, 1994. - 285 с.
9. Богумирский Б. Энциклопедия Windows_98. - СПб.: Питер, 1998.- 816 с.
10. Маликов В. Т.; Кветный Р.Н. Вычислительные методы и применение ЭВМ: Учеб. пособие.-К.: Выща шк. Головное изд-во, 1989. - 213 с.
11. К.Корн, Т. Корн. Справочник по математике для научных работников и инженеров.-М.: Наука, 1973. - 832 с.

Навчальне видання

Володимир Юрійович Кошобинський

Прикладні програмні системи

Конспект лекцій

Оригінал-макет підготовлено автором

Редактор С. А. Малішевська

Навчально-методичний відділ ВДТУ
Свідоцтво Деркомінформу України
серія ДК № 746 від 25.12.2001
21021, м.Вінниця, Хмельницьке шосе, 95, ВДТУ

Підписано до друку *11.07.03*
Формат 29,7x42 1/4
Друк різнографічний
Тираж *100* прим.
Зам. № *2003 - 119*

Гарнітура Times New Roman
Папір офсетний
Ум. друк. арк. *313*

Віддруковано в комп'ютерному інформаційно-видавничому центрі
Вінницького державного технічного університету
Свідоцтво Деркомінформу України
серія ДК № 746 від 25.12.2001
21021, м. Вінниця, Хмельницьке шосе, 95