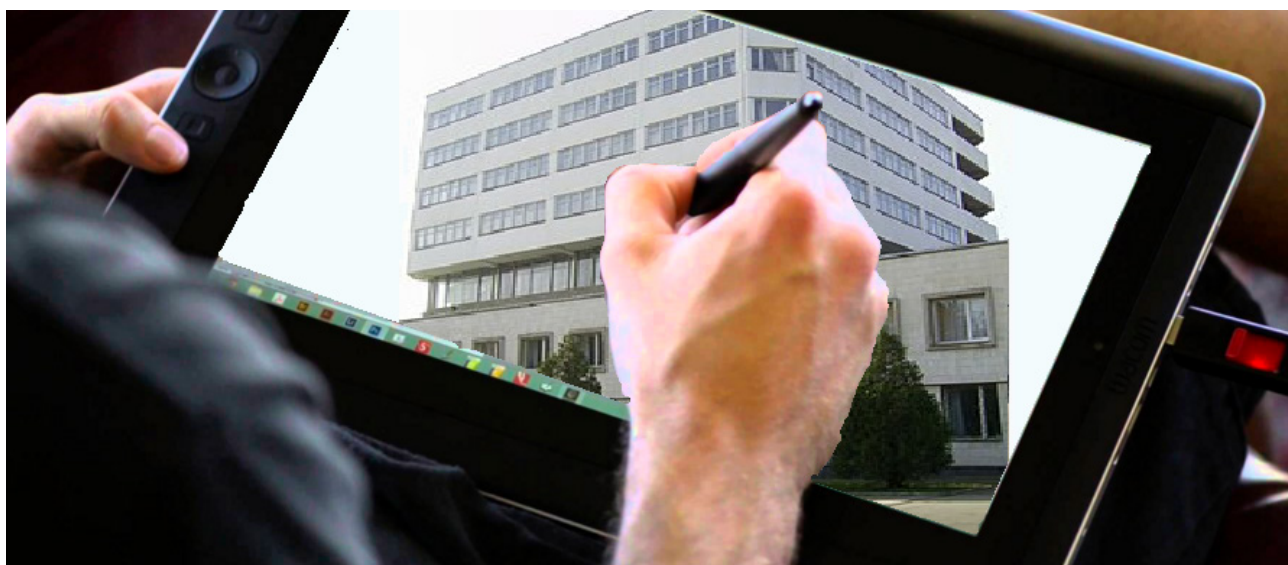


**О. Н. Романюк, В. П. Майданюк**

## **ПРАКТИКУМ**

**для самостійної роботи студентів**

**з навчальної дисципліни «Комп'ютерна графіка»**



Міністерство освіти і науки України  
Вінницький національний технічний університет

## **ПРАКТИКУМ**

**для самостійної роботи студентів  
з навчальної дисципліни  
«Комп'ютерна графіка»**

**Електронний практикум  
комбінованого (локального та мережного) використання**

Вінниця  
ВНТУ  
2022

УДК 004.925  
Р 69

Рекомендовано до видання Вченою радою Вінницького національного технічного університету Міністерства освіти і науки України (протокол № 7 від 31.03.2022 р.)

Рецензенти:

**С. В. Павлов**, доктор технічних наук, професор ВНТУ

**О. М. Васілевський**, доктор технічних наук, професор ВНТУ

**Т. І. Коробейнікова**, кандидат технічних наук, доцент НУ «Львівська політехніка»

**Романюк, О. Н.**

Р 69 Практикум для самостійної роботи студентів з навчальної дисципліни «Комп'ютерна графіка» : електронний практикум комбінованого (локального та мережного) використання [Електронний ресурс] / О. Н. Романюк, В. П. Майданюк. – Вінниця : ВНТУ, 2022. – 77 с.

Практикум присвячений розв'язанню типових задач з комп'ютерної графіки, а також тестуванню знань студентів.

Практикум рекомендується для самостійної роботи студентів з навчальної дисципліни «Комп'ютерна графіка», які навчаються за бакалаврською програмою зі спеціальності 121 «Інформаційні технології».

УДК 004.925

## ЗМІСТ

ВСТУП.....	4
1 МЕТОДИ ТА ЗАСОБИ ФОРМУВАННЯ ЗОБРАЖЕНЬ.....	5
2 ПРОЦЕДУРИ КОМП'ЮТЕРНОЇ ГРАФІКИ.....	34
3 КОДУВАННЯ ГРАФІЧНОЇ ІНФОРМАЦІЇ.....	55
ТЕСТ.....	63
ЛІТЕРАТУРА.....	76

## ВСТУП

Бурхливий розвиток засобів обробки інформації, підвищення рівня автоматизації процесів виробництва і керування приводить до постійного зростання ролі комп'ютерної графіки, оскільки зображення є найбільш інформативним каналом зв'язку між людиною та ЕОМ.

Для ефективного розв'язання задач користувачу потрібно дати великий обсяг інформації в зручному для нього вигляді з заданою точністю в реальному масштабі часу.

Комп'ютерна графіка дозволяє суттєво збільшити пропускну спроможність інформаційного каналу, через який здійснюється двосторонній зв'язок користувача й комп'ютера, в зв'язку з чим роль і значення графічного подання результатів обчислень в промисловості та науково-дослідній практиці безупинно зростає.

У практикуму з комп'ютерної графіки наведено приклади розв'язання різних типів задач, які найбільш поширені в даній галузі, також тести для перевірки отриманих знань.

Використання практикуму сприятиме актуалізації процесу підготовки, індивідуалізації та диференціації освітнього процесу, запровадженню особистісного підходу в навчальній діяльності.

Розв'язання практичних занять з комп'ютерної графіки дасть можливість поглибити знання, які було отримано на лекційних заняттях, та підтвердити і реалізувати навички, що були сформовані на лабораторних та практичних заняттях з дисципліни «Комп'ютерна графіка». Важливим завданням є формування компетентностей, що дозволяють студенту реалізовувати на практиці отримані знання.

# 1 МЕТОДИ ТА ЗАСОБИ ФОРМУВАННЯ ЗОБРАЖЕНЬ

**1. Визначити мінімальну кількість інтерполяційних тактів, необхідну для формування вектора ортогональними кроковими приростами**

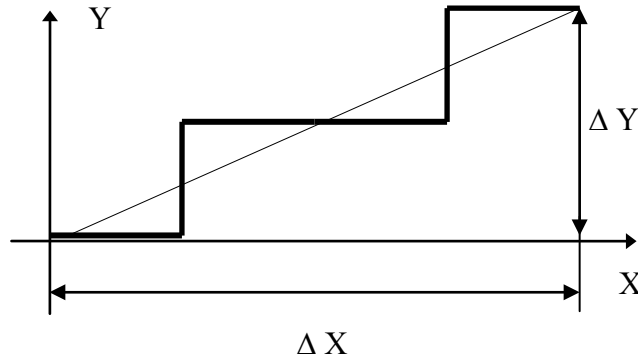


Рисунок 1.1 – Формування вектора вертикальними та горизонтальними кроковими приростами

Формування вектора здійснюється вертикальними та горизонтальними кроковими приростами (рис. 1.1).

Кожний з ортогональних кроків унеможливує переміщення по іншій координаті. Оскільки переміщення в напрямку осі абсцис здійснюється тільки горизонтальними кроками, то їх кількість дорівнює  $\Delta X$ .

Аналогічно, кількість інтерполяційних тактів в напрямку осі ординат дорівнює  $\Delta Y$ . Загальна кількість інтерполяційних тактів  $\Delta X + \Delta Y$ .

**2. Виконайте інтерполювання відрізка прямої з приростами  $\Delta X = 5$ ,  $\Delta Y = 2$  за алгоритмом Петуха А. М., Обідника Д. Т.**

Інтерполювання за алгоритмом Петуха А. М., Обідника Д. Т. здійснюється згідно з формулами

$$\begin{aligned} O\Phi_0 &= \lfloor \text{БП}/2 \rfloor, \Delta = \text{БП} - \text{МП}, \\ \begin{cases} O\Phi_{i+1} = O\Phi_i - \text{МП} & \text{при } O\Phi_i \geq 0, \\ O\Phi_{i+1} = O\Phi_i + \Delta & \text{при } O\Phi_i < 0, \end{cases} \end{aligned}$$

де БП і МП, відповідно, більше та менше значення приростів  $\Delta X, \Delta Y$ .

$$\begin{aligned} \text{БП} &= 5, \quad \text{МП} = 2, \quad \Delta = 3. \\ O\Phi_0 &= \lfloor 5/2 \rfloor = 2; \quad O\Phi_1 = 2 - 2 = 0; \\ O\Phi_2 &= 0 - 2 = -2; \quad O\Phi_3 = -2 + 3 = 1; \\ O\Phi_4 &= 1 - 2 = -1; \quad O\Phi_5 = -1 + 3 = 2 \end{aligned}$$

При  $O\Phi_i \geq 0$  виконується горизонтальний крок, а при  $O\Phi_i < 0$  – діагональний. Сформовану траєкторію відображено на рис. 1.2.

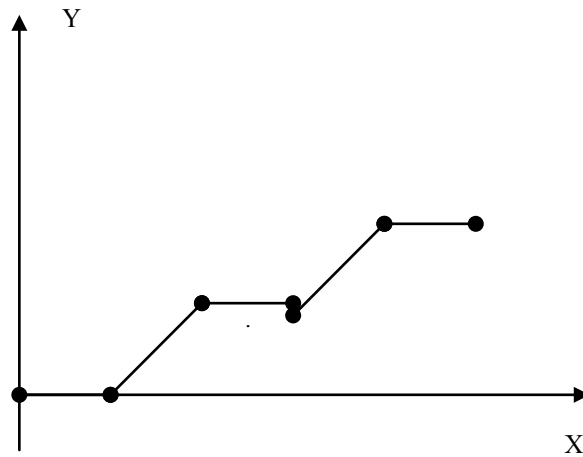


Рисунок 1.2 – Формування вектора кроковими приростами

**3. *Визначити тип вектора, який формується вертикальними та діагональними кроками, для якого знак оцінювальної функції в кожному інтерполяційному такті змінюється на протилежний, а значення адрес пікселів зменшується***

Оскільки крокова траєкторія формується вертикальними та діагональними приростами, то зрозуміло, що у вектора приріст  $\Delta Y$  більший за приріст  $\Delta X$ .

При додатному знакові оцінювальної функції виконується вертикальний крок, а при від'ємному – діагональний.

За умовою задачі знак оцінювальної функції змінюється в кожному інтерполяційному такті, тобто, за кожним вертикальним кроком формується діагональний і навпаки.

Оскільки діагональний крок містить приріст по осі  $Y$ , то можна констатувати, що на кожне переміщення по осі  $X$  припадає два переміщення по осі  $Y$ , тобто  $\Delta Y = 2\Delta X$ .

Оскільки адреси пікселів при формуванні вектора зменшуються, то це означає, що він формується в напрямку до початкової точки екрана.

На рис. 1.3 наведено зображення крокової траєкторії.

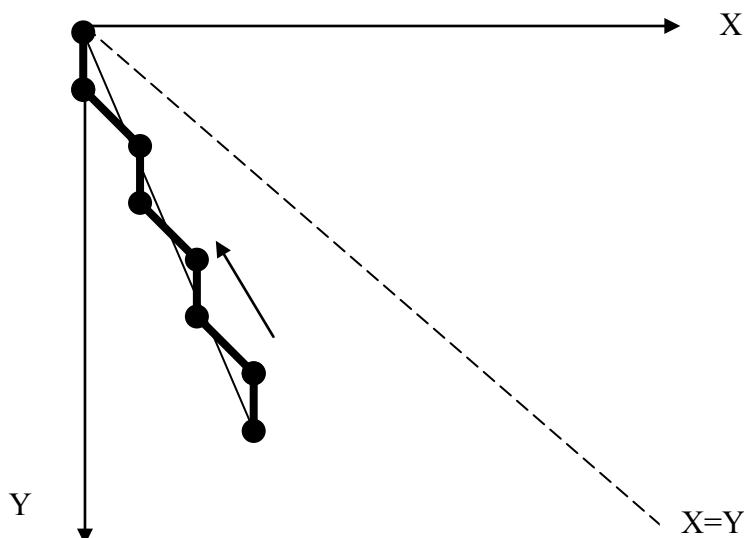


Рисунок 1.3 – Зображення крокової траєкторії

**4. Довести, що максимальна похибка алгоритмів лінійної інтерполяції з восьмивекторним направленням крокових переміщень не може бути меншою половини кроку дискретизації**

Крокова траєкторія, яка формується алгоритмами з восьмивекторним направленням крокових приростів, охоплює горизонтальні, вертикальні та діагональні прирости, що дозволяє вибрати найближчі точки решітки відносно ідеального відрізка прямої.

Відрізок прямої, відносно найближчих точок решітки дискретного простору, має дві похибки:  $\delta^+$  та  $\delta^-$  (рис. 1.4). З двох можливих точок решітки, одна з яких знаходиться вище, а друга – нижче відрізка прямої, доцільно вибрати ту, для якої похибка  $\delta^+$  чи  $\delta^-$  менша за модулем.

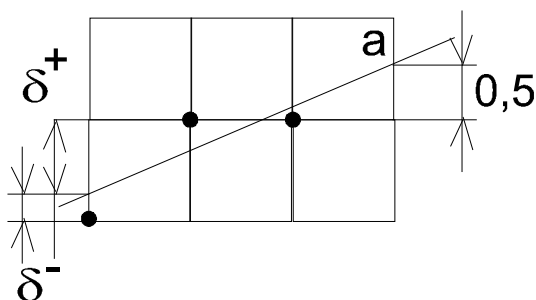


Рисунок 1.4 – Розподіл похибок

У випадку (а), коли ідеальний відрізок прямої проходить через середину дискрети, то  $\delta^+ = |\delta^-| = 0,5$ ; тому найближча точка решітки віддалена від векторів на 0,5 кроку дискретизації. В цьому випадку маємо два ідентичних варіанти вибору найближчих точок решітки, які найближче знаходяться до відрізка прямої. Кожна з цих точок віддалена від відрізка на половину кроку дискретизації.

Наведений приклад показує, що досягти меншої похибки в цьому випадку неможливо, тому максимальна похибка алгоритмів лінійної інтерполяції не може бути меншою половини кроку дискретизації.

**5. Визначити значення оцінювальної функції в точці (6,2) при формуванні відрізка прямої за алгоритмом Петуха, Обідника за умови, що  $\Delta x = 8, \Delta y = 3, x_n = 0, y = 0$**

За умовою виконано 4 горизонтальних і 2 діагональних координатних прирости. При формуванні горизонтального кроку значення оцінювальної функції визначається за формулою

$$ОФ_{i+1} = ОФ_i - МП,$$

де МП – менший з приростів. В даному випадку МП = 3.



При виконанні діагонального кроку

$$ОФ_{i+1} = ОФ_i + \Delta,$$

де  $\Delta = БП - МП$ .

$$\Delta = 5.$$

Підсумовуючи вказане, знаходимо

$$ОФ_6 = ОФ_0 - 4 МП + 2\Delta.$$

Для алгоритму Петуха, Обідника  $ОФ_0 = \lfloor БП/2 \rfloor = 4$ . Враховуючи початкове значення оцінювальної функції знаходимо

$$ОФ_6 = 4 - 12 + 10 = 2.$$

**6. Визначте тип вектора, який, згідно з алгоритмом лінійної інтерполяції Петуха А. М., Обідника Д. Т., має всі значення останньої, рівні початковому значенню**

Оцінювальна функція обчислюється за формулами:

$$\begin{cases} ОФ_0 = \lfloor БП/2 \rfloor \\ ОФ_{i+1} = ОФ_i - МП \text{ при } ОФ_i \geq 0, \\ ОФ_{i+1} = ОФ_i + \Delta \text{ при } ОФ_i < 0, \end{cases}$$

де МП та БП – відповідно, менший та більший прирости відрізка прямої,  $\Delta = БП - МП$ .

Оскільки початкове значення оцінювальної функції завжди більше нуля, то

$$ОФ_1 = ОФ_0 - МП.$$

При  $МП = 0$ ,  $ОФ_1 = ОФ_0 = ОФ_i$ ,  $i = 1, БП$

**7. Довести, що початкове та кінцеве значення оцінювальної функції при формуванні вектора за алгоритмом лінійної інтерполяції Петуха-Обідника рівні**

Оцінювальна функція розраховується за формулами:

$$ОФ_{i+1} = \begin{cases} ОФ_i - МП \text{ при } ОФ_i \geq 0, \\ ОФ_i + \Delta \text{ при } ОФ_i < 0, \end{cases}$$

де МП, БП – відповідно, менший та більший прирости,  $\Delta = БП - МП$ ,  $ОФ_0 = \text{int}(БП/2)$ . При  $ОФ_i \geq 0$  формується горизонтальний (вертикальний) крок, а при  $ОФ_i < 0$  – діагональний.

Кількість діагональних кроків дорівнює МП, а кількість горизонтальних (вертикальних) – БП - МП. Оскільки виконується БП інтерполяційних тактів, кінцеве значення оцінювальної функції дорівнює

$$O\Phi_k = O\Phi_{БП} = O\Phi_0 - МП*(БП - МП) + \Delta*МП = O\Phi_0 - МП*\Delta + МП*\Delta = O\Phi_0.$$

**8. Визначити мінімальну кількість інтерполяційних тактів алгоритмів лінійної інтерполяції з восьми векторним направленням крокових приростів**

Формування крокової траєкторії за вказаних алгоритмів здійснюється горизонтальними (вертикальними) та діагональними кроками. Діагональний крок містить в собі одночасне переміщення по обох координатах. При  $\Delta x > \Delta y$  крокова траєкторія складається з горизонтальних та діагональних приростів. Враховуючи, що діагональний крок містить переміщення в горизонтальному напрямку, робимо висновок, що кількість кроків для досягнення кінцевої точки дорівнює  $\Delta x$ .

Аналогічно можна показати, що при  $\Delta y > \Delta x$  потрібно виконати  $\Delta y$  кроків. Таким чином, кількість інтерполяційних тактів для алгоритмів з восьми векторною орієнтацією крокових приростів дорівнює більшому приросту.

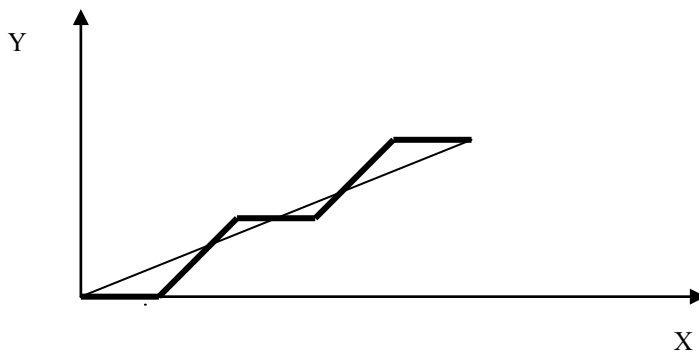


Рисунок 1.5 – Крокова траєкторія

**9. Розробити структурну схему лінійного інтерполятора, який формує крокову траєкторію за методом оцінювальної функції**

За базовий алгоритм використаємо алгоритм оцінювальної функції, згідно з яким виконуються такі дії:

$$\Delta = БП - МП; O\Phi_0 = [БП/2],$$

де БП, МП – відповідно, більший та менший прирости.

$$O\Phi_{i+1} = \begin{cases} O\Phi_i - МП \text{ при } O\Phi_i \geq 0 \\ O\Phi_i + \Delta \text{ при } O\Phi_i < 0 \end{cases}$$



**10. Визначити значення інтенсивностей кольору точок для забезпечення антиаліаїзину векторної границі з приростами  $\Delta X = 5, \Delta Y = 2$  за умови, що інтенсивність кольору  $I = 10$**

Коефіцієнт нахилу вектора дорівнює  $\frac{\Delta Y}{\Delta X}$ . Задамо кут нахилу відношенням

$$\frac{I}{I_x}, \text{ тобто } \frac{\Delta Y}{\Delta X} = \frac{I_x}{I}. I_x = \frac{\Delta Y * I}{\Delta X} = \frac{2 * 10}{5} = 4.$$

Таким чином,  $I_x = 4, I = 10$ . Виконаємо інтерполювання з параметрами  $I_x, I$  при кількості інтерполяційних тактів  $\Delta X = 5$ .

$$\text{БП} = 10, \text{МП} = 4$$

$$\text{ОФ}_0 = [\text{БП}/2] = 5; \Delta = \text{БП} - \text{МП} = 6$$

$$\text{ОФ}_1 = \text{ОФ}_0 - \text{МП} = 5 - 4 = 1,$$

$$\text{ОФ}_2 = \text{ОФ}_1 - \text{МП} = 1 - 4 = -3,$$

$$\text{ОФ}_3 = \text{ОФ}_2 + \Delta = -3 + 6 = 3,$$

$$\text{ОФ}_4 = \text{ОФ}_3 - \text{МП} = 3 - 4 = -1,$$

$$\text{ОФ}_5 = \text{ОФ}_4 + \Delta = -1 + 6 = 5.$$

Встановленням  $I_0 = 5, I_1 = 1, I_2 = 3, I_3 = 3, I_4 = 1, I_5 = 5$  забезпечує згладження ступінчастої форми вектора.

**11. Визначити кількість інтерполяційних тактів для формування траєкторії кола ортогональними кроковими приростами**

Розглянемо перший квадрат (рис. 1.7). Крокова траєкторія формується горизонтальними та вертикальними елементарними переміщеннями. Кількість крокових приростів у напрямку координат  $X$  дорівнює радіусу  $R$ , оскільки вертикальні кроки не забезпечують переміщення по осі абсцис. Аналогічна ситуація має місце є для ординатного напрямку. Таким чином, кількість крокових приростів для формування траєкторії в першому квадраті дорівнює  $2R$ , а для всього кола

$$2 R * 4 = 8 R.$$

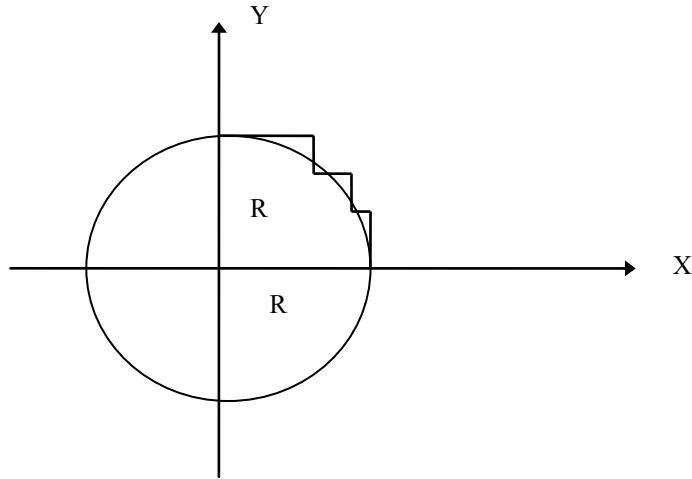


Рисунок 1.7 – Формування кола

**12. Визначити п'ять точок траєкторії дуги кола, яка починається з точки (6,0) (рис. 1.8) за умови, що інтерполювання здійснюється за методом оцінювальної функції проти годинникової стрілки, а радіус кола дорівнює 6**

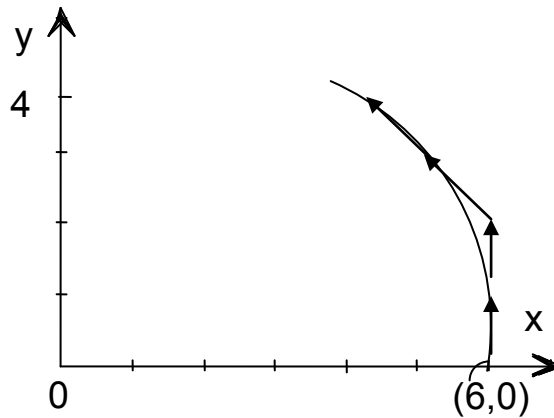


Рисунок 1.8 – Формування кола кроковими приростами

Оскільки інтерполювання здійснюється проти годинникової стрілки, то координати  $x$  точок траєкторії будуть зменшуватись, а координати  $y$  – збільшуватись (рис. 1.8). Формули для розрахунку оцінювальної функції мають вигляд

$$O\Phi_{i+1} = \begin{cases} O\Phi_i + 2y_i + 1 & \text{при } O\Phi < 0 \\ O\Phi + 2(y_i - x_i) + 2 & \text{при } O\Phi \geq 0 \end{cases}$$

$$O\Phi_0 = -R.$$

$$O\Phi_0 = -6.$$

$$O\Phi_1 = -6 + 2 * 0 + 1 = -5$$

$$O\Phi_2 = -5 + 2 * 1 + 1 = -2$$

$$O\Phi_3 = -2 + 2 * 2 + 1 = +3$$

$$O\Phi_4 = 3 + 2*(3-5) + 2 = +5$$

Згідно зі знаками оцінювальної функції робимо висновок, що в перших двох тактах формуються вертикальні кроки, а в двох наступних – діагональні. Початкова точка дуги має координати (6,0).

**13. Визначити та обґрунтувати напрямки траєкторії в режимі доведення в кінцеву точку дуги**

В першому секторі (рис. 1.9, а) крокова траєкторія формується горизонтальними та діагональними (комбінованими) кроковими приростами

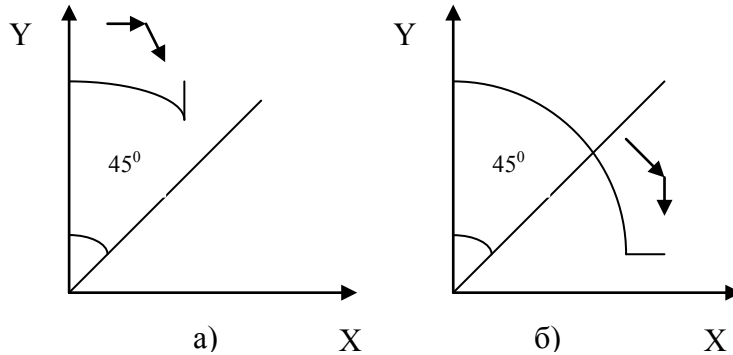


Рисунок 1.9 – Ділянка формування кола

Враховуючи, що діагональний приріст містить переміщення по осі X, то можна констатувати, що всі елементарні кроки до точки  $X = Y$  містять переміщення по осі абсцис.

Звідки випливає, що при формуванні траєкторії гарантовано буде досягнуто координату  $X_k$ , а доведення буде необхідне для координати  $Y_k$ .

Аналогічно можна показати, що при формуванні дуги в другому секторі доведенню підлягає координата  $X_k$ .

**14. Розрахувати 4 крокові прирости для параболи  $Y = 2X^2$  за методом оцінювальної функції**

Оцінювальна функція визначається за формулою  $OF_i = y - 2x_i^2$ .

При  $OF \geq 0$  (рис. 1.9) виконується крок по осі X, а при  $OF < 0$  по осі Y. Після кроку по осі X нове значення  $x_{i+1} = x_i + 1$ .

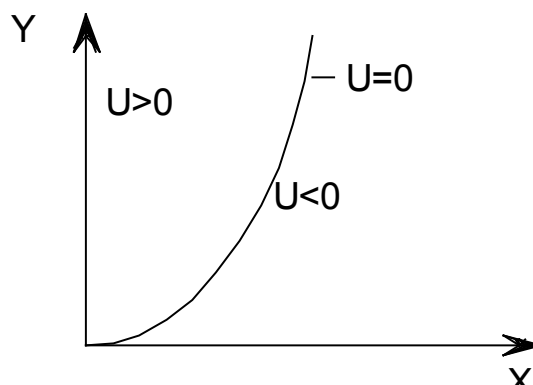


Рисунок 1.10 – Ділянка формування параболи

$$\text{Тоді } O\Phi_{i+1} = y_i - 2*(x+1)^2 = y_i - 2*x_i^2 - 2*x + 2 = y_i - 4x_i - 2.$$

При виконанні кроку по осі Y нове значення аргумента буде розраховуватись як  $y_i = y_i + 1$ .

$$\text{Тоді } O\Phi_{i+1} = y_i + 1 - 2*x_i^2 = O\Phi_i + 1.$$

Розрахуємо 4 крокові прирости за умови, що  $O\Phi_0 = 0$ .

1.  $O\Phi_1 = 0 - 4 - 2 = -6$  ,  $y = 0$  ,  $x = 1$ .
2.  $O\Phi_2 = -6 + 1 = -5$  ,  $y = 1$  ,  $x = 1$ .
3.  $O\Phi_3 = -5 + 1 = -4$  ,  $y = 2$  ,  $x = 1$ .
4.  $O\Phi_4 = -4 + 1 = -3$  ,  $y = 3$  ,  $x = 1$ .

**15. Привести вираз для кривої Ерміта, початкова та кінцева точки якої мають координати  $(0, \gamma)$ ,  $(1, \beta)$ , а похідні в початковій та кінцевій точках відповідно рівні  $\alpha, \gamma$**

Кубічна крива в формі Ерміта задається рівнянням

$$x(t) = at^3 + bt^2 + ct + d.$$

Знайдемо значення невідомих a, b, c, d.

$$\begin{aligned} x(0) &= d, & x(1) &= a+b+c+d, \\ x'(0) &= c, & x'(1) &= 3a+2b+c. \end{aligned}$$

Таким чином

$$\begin{aligned} \gamma &= c, \\ \beta &= a+b+c+d, \\ \alpha &= c, \\ v &= 3a+2b+c. \end{aligned}$$

Розв'язок системи дає невідомі a, b, c, d.

**16. Визначити геометричний вектор Безьє для кривої  $X(t) = 3t + 15t^2 + 4t + 1$  за умови, що  $P_2 = 7$ ,  $P_3 = 9$**

Геометричний вектор Безьє має вигляд

$$G = [ \quad ],$$

де  $P_1 = x(0)$ ,  $P_2 = x(1)$ ,  $P_3 = x'(0)$ ,  $P_4 = x'(1)$ .

$$G = \begin{vmatrix} P_1 \\ P_4 \\ R_1 \\ R_4 \end{vmatrix}$$

Підставляючи в  $x(t)$  значення 0 та 1, одержимо

$$P1 = 1, \quad P4 = 23.$$

Згідно з алгоритмом Безьє значення  $R1$  та  $R4$  визначаємо за формулами:

$$R1 = 3 * (P2 - P1),$$

$$R4 = 3 * (P4 - P3),$$

$$R1 = 18, \quad R2 = 42.$$

Таким чином, геометричний вектор Безьє для даного прикладу має вигляд

$$G = \begin{vmatrix} 1 \\ 23 \\ 18 \\ 42 \end{vmatrix}.$$

**17. Знайти вираз для полінома Лагранжа за умови, що базові точки мають такі координати: (7, 10), (9, 30), (11, 50)**

Інтерполяційний поліном Лагранжа  $n$ -го степеня для даної множини точок має вигляд

$$y = \sum_{s=0}^n \left( \frac{(x-x_0)\dots(x-x_{s-1})(x-x_{s+1})\dots(x-x_n)}{(x_s-x_0)\dots(x_s-x_{s-1})(x_s-x_{s+1})\dots(x_s-x_n)} \right) y_s.$$

У даному випадку маємо

$$\begin{aligned} y &= \frac{(x-x_2)(x-x_3)}{(x_1-x_2)(x_1-x_3)} y_1 + \frac{(x-x_1)(x-x_3)}{(x_2-x_1)(x_2-x_3)} y_2 + \frac{(x-x_1)(x-x_2)}{(x_3-x_1)(x_3-x_2)} y_3 = \\ &= \frac{(x-9)(x-11)}{(7-9)(7-11)} 10 + \frac{(x-7)(x-11)}{(9-7)(9-11)} 30 + \frac{(x-7)(x-9)}{(11-7)(11-9)} 50 = \\ &= \frac{(x-9)(x-11)}{(-2)(-4)} 10 + \frac{(x-7)(x-11)}{2(-2)} 30 + \frac{(x-7)(x-9)}{4(2)} 50 = \\ &= (x^2 - 11x - 9x + 99) \frac{10}{8} + (x^2 - 11x - 7x + 77) \left(-\frac{30}{4}\right) + (x^2 - 7x - 9x + 63) \frac{50}{8} = \\ &= 1,25x^2 - 25x + 123,75 - 7,5x^2 + 135x - 577,5 + 6,25x^2 - 100x + 393,75 = \\ &= 10x - 60 \end{aligned}$$

**Висновок:** в даному випадку крива вироджена в відрізок прямої, який проведений через три точки.



**18. Побудуйте сегмент кривої за допомогою форми Ерміте для таких граничних умов**

$$P1=[00], P4=[1 0]$$

$$R1=[01], R4=[1 0]$$

Замінімо значення граничних умов у вираз для рівняння кривої Ерміте та отримаємо тип кривої (рис. 1.11)

$$[x(t) \ y(t)] = [t^3 \ t^2 \ t \ 1] \cdot \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} = [t^3 \ t^2 \ t \ 1] \cdot \begin{bmatrix} -1 & 1 \\ 2 & -2 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$

Параметр  $t$  змінюється в межах  $[0, 1]$ . Встановимо крок зміни, що дорівнює  $0,1$ . У результаті отримуємо таблицю значень координат залежно від параметра  $t$  і, відповідно до цих значень, побудуємо сегмент кривої

$t$	$x(t)$	$y(t)$
0.0	0.000	0.000
0.1	0.019	0.081
0.2	0.072	0.128
0.3	0.153	0.147
0.4	0.256	0.144
0.5	0.375	0.125
0.6	0.504	0.096
0.7	0.637	0.063
0.8	0.768	0.032
0.9	0.891	0.009
1.0	1.000	0.000

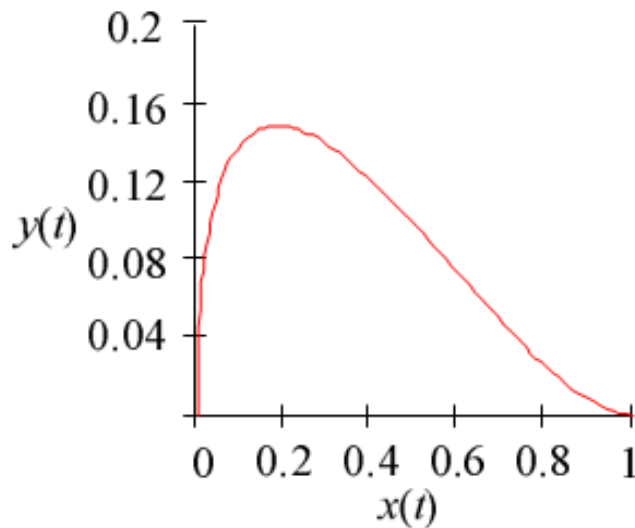


Рисунок 1.11 – Крива Ерміта

**19. Побудувати інтерполяційний поліном Лагранжа згідно з даними, які задано таблицею**

X	0	1	2	3
y	-2	-5	0	-4

Запишемо формулу для інтерполяційного многочлена в формі Лагранжа.

$$L(x) = y_0 \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1 \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} +$$

$$+ y_2 \frac{(x-x_0)(x-x_1)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)}.$$

Підставимо дані з таблиці

$$L(x) = -2 \frac{(x-1)(x-2)(x-3)}{(0-1)(0-2)(0-3)} - 5 \frac{(x-0)(x-2)(x-3)}{(1-0)(1-2)(1-3)} + 0 \frac{(x-0)(x-1)(x-3)}{(2-0)(2-1)(2-3)} +$$

$$+ 0 \frac{(x-0)(x-1)(x-2)}{(3-0)(3-1)(3-2)} = -\frac{17}{6}x^3 + 12,5x^2 - \frac{38}{3}x - 2.$$

20. Задано відрізок, координати кінців якого  $P_3(-\frac{5}{2}, -1)$ ,  $P_4(\frac{3}{2}, 2)$ .

«Відсікаюче» вікно –  $x_{\min}, x_{\max}, y_{\min}, y_{\max}$ , які дорівнюють  $(-1, 1, -1, 1)$  (рис. 1.12)

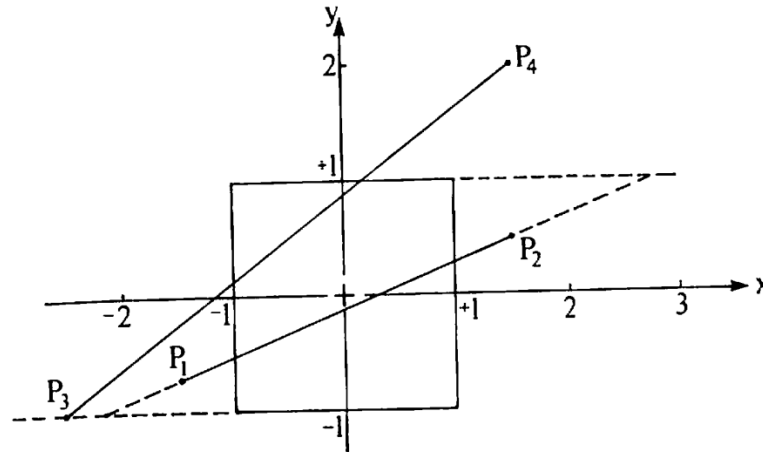


Рисунок 1.12 – Відсікання відрізків

Знайти видиму частину відрізка.

Точки перетину задаються параметрами

$$t_L = \frac{3}{8}, \quad t_{\Pi} = \frac{7}{8}, \quad t_H = 0, \quad t_B = \frac{2}{3}.$$

Видно, що всі чотири значення задовольняють умову  $0 \leq t \leq 1$ . Але точок перетину тільки дві (!) для опуклого многокутника. Далі треба відібрати два значення параметра з чотирьох. У даному випадку потрібно підставляти значення параметра і звіряти з рисунком. Отримаємо, що шуканими значеннями параметра є  $t_L = \frac{3}{8}$  і  $t_B = \frac{2}{3}$ , які визначають координати точок перетину  $(-1, 1/8)$  і  $(1/6, 1)$ .

21. Нехай задано повністю видимий відрізок із кінцевими точками  $P_1(-\frac{1}{2}, 1/2)$ ,  $P_2(\frac{1}{2}, -1/2)$ , в тому ж вікні  $(-1, 1, -1, 1)$ . (рис. 1.13). З'ясувати, видимий даний відрізок чи невидимий

Розрахунки, аналогічні попередньому прикладу, дають значення для параметрів, які відповідають перетинам відрізка зі сторонами вікна.

$$t_L = -\frac{1}{2}, \quad t_{\Pi} = \frac{3}{2}, \quad t_H = \frac{3}{2}, \quad t_B = -\frac{1}{2}$$

Усі значення параметра знаходяться поза інтервалом  $0 \leq t \leq 1$ .

Це означає, що відрізок не перетинає межі вікна, але чи він повністю видимий, чи повністю невидимий, Наприклад, для повністю невидимого відрізка  $P_3\left(\frac{3}{2}, -1/2\right)$ ,  $P_4\left(2, \frac{1}{2}\right)$  на рис. 1.13 розраховані значення параметра, які відповідають перетинам відрізка зі сторонами вікна, будуть

$$t_L = -5, \quad t_{\Pi} = -1, \quad t_H = -\frac{31}{2}, \quad t_B = \frac{3}{2}.$$

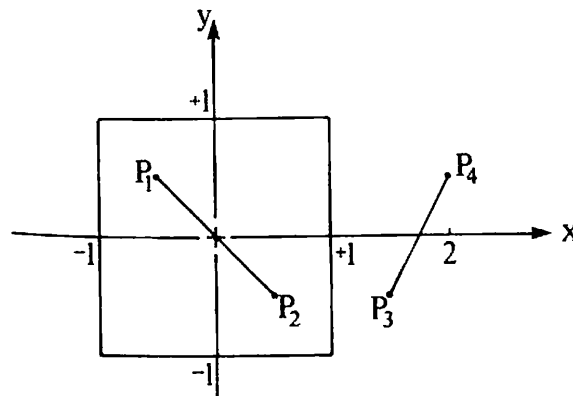


Рисунок 1.13 – Повністю видимий і повністю невидимий відрізки

Знову, як і в попередньому випадку, всі значення параметра знаходяться поза інтервалом  $0 \leq t \leq 1$ . Проте відрізок повністю невидимий. Тобто встановити, чи відрізок повністю видимий чи повністю невидимий, цим методом неможливо.

**22. Приймаючи крок  $h = 0,05$ , побудувати інтерполяційний поліном Ньютона для функції  $y = 2 - 3x + 4x^2 - 6x^3$ , заданої нижченаведеною таблицею**

$x$	1	1,05	1,1	1,15	1,2	1,25	1,3
$y$	-3	-3,685	-4,445	-5,285	-6,207	-7,218	-8,321

Для практичного використання інтерполяційну формулу Ньютона записують у вигляді

$$P_n(x) = y_0 + q\Delta q + \frac{q(q-1)}{2!}\Delta^2 y_0 + \dots + \frac{q(q-1)\dots(q-n+1)}{n!}\Delta^n y_0,$$

де  $q = \frac{x - x_0}{h}$  – число кроків, необхідних для досягнення точки  $x$ , виходячи з точки  $x_0$ .

Складаємо таблицю різниць.

$x$	$y$	$\Delta y$	$\Delta^2 y$	$\Delta^3 y$
1	-3	0,685	-0,075	0,005
1,05	-3,685	0,76	-0,08	0,002
1,1	-4,445	0,84	-0,082	0,007
1,15	-5,285	0,922	-0,089	0,003
1,2	-6,207	1,011	-0,092	
1,25	-7,218	1,103		
1,3	-8,321			

Оскільки різниці третього порядку практично постійні, то у формулі Ньютона  $n = 3$ . Приймавши  $x_0 = 1$ ,  $y_0 = -3$ , матимемо

$$P_n(x) = -3 + 0.685q - 0.075 \frac{q(q-1)}{2} + 0.005 \frac{q(q-1)(q-2)}{6},$$

або

$$P_n(x) = -3 + 0.685q - 0.075 \frac{q(q-1)}{2} + 0.005 \frac{q(q-1)(q-2)}{6}$$

$$P_n(x) = -3 + 0.685q - 0.0375q(q-1) + 0.00083q(q-1)(q-2),$$

Це і шуканий інтерполяційний поліном Ньютона.

**23. Визначити рівняння площини, що проходить через три точки  $(5, 4, 2)$ ,  $(-1, 7, 3)$ , і  $(2, -2, 9)$**

Якщо площина проходить через три точки  $(5, 4, 2)$ ,  $(-1, 7, 3)$ , і  $(2, -2, 9)$ , то визначник матиме вигляд

$$\begin{aligned} \begin{vmatrix} e_1 & e_2 & e_3 & e_4 \\ 5 & 4 & 2 & 1 \\ -1 & 7 & 3 & 1 \\ 2 & -2 & 9 & 1 \end{vmatrix} &= e_1 \begin{vmatrix} 4 & 2 & 1 \\ 7 & 3 & 1 \\ -2 & 9 & 1 \end{vmatrix} - e_2 \begin{vmatrix} 5 & 2 & 1 \\ -1 & 3 & 1 \\ 2 & 9 & 1 \end{vmatrix} + \\ &+ e_3 \begin{vmatrix} 5 & 4 & 1 \\ -1 & 7 & 1 \\ 2 & -2 & 1 \end{vmatrix} - e_4 \begin{vmatrix} 5 & 4 & 2 \\ -1 & 7 & 3 \\ 2 & -2 & 9 \end{vmatrix} = \\ &= 27e_1 + 39e_2 + 45e_3 - 381e_4 \end{aligned}$$

Отже, маємо рівняння площини, що проходить через три точки  $27x + 39y + 45z - 381 = 0$ .

**24. Визначити точки перетину трьох площин  $3x + 5y + z = 2$ ,  $7x - 4z = -1$ ,  $2y + 5z + 8 = 0$**

Визначник із коефіцієнтів, що задають площини, буде

$$\begin{vmatrix} e_1 & e_2 & e_3 & e_4 \\ 3 & 5 & 1 & -2 \\ 7 & 0 & -4 & 1 \\ 0 & 2 & 5 & 8 \end{vmatrix} = -199e_1 + 237e_2 - 314e_3 + 137e_4.$$

Відповідно, однорідні координати точки перетину  $(-199, 237, -314, 137)$ , а декартові  $(-199/137, 237/137, -314/137)$ .

**25. Знайти координати точки перетину  $M$  двох прямих  $AB$  і  $CD$**

Два рівняння прямих, що проходять через точки  $AB$  і  $CD$  відповідно, будуть

$$\frac{x - x_A}{x_B - x_A} = \frac{y - y_A}{y_B - y_A}$$

$$\frac{x - x_C}{x_D - x_C} = \frac{y - y_C}{y_D - y_C}$$

Рівняння прямої, що проходить через точки  $A$ ,  $B$  і точку  $M$ , перепишемо у вигляді

$$x_M = x_A + (y_M - y_A) \frac{x_B - x_A}{y_B - y_A}.$$

Знайдене  $x_M$  підставимо в рівняння та розв'яжемо його відносно  $y_M$

$$y_M = \frac{y_A \cdot \rho(y_D - y_C) - x_A(y_D - y_C) - y_C(x_D - x_C)}{\rho(y_D - y_C) - (x_D - x_C)}$$

$$\rho = \frac{x_B - x_A}{y_B - y_A}.$$

Знайдене  $y_M$  підставимо у вираз для  $x_M$  і цим знайдемо обидві координати точки перетину прямих  $AB$  і  $CD$  (рис. 1.14).

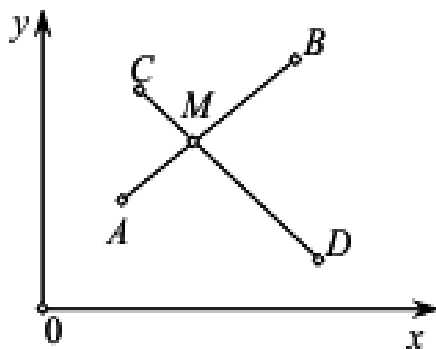


Рисунок 1.14 –  $M$  – точка перетину двох прямих  $AB$  і  $CD$

### 26. Знайти кривизну кола

Кривизною  $k$  кривої в точці  $M_0$  називається границя відношення  $\frac{\Delta\theta}{\Delta L}$  при  $M \rightarrow M_0$ , де  $\Delta\theta$  – менший з кутів між дотичними до кривої в точках  $M$  і  $M_0$ , а  $\Delta L$  – довжина дуги  $M_0M$  (рис. 1.15).

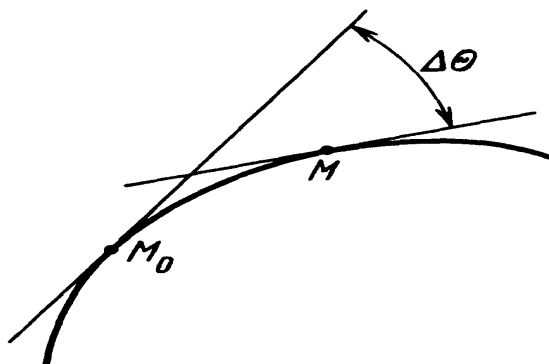


Рисунок 1.15 – До визначення кривизни кривої

Вираз для кривизни такий

$$k(t) = \frac{|x'(t)y''(t) - x''(t)y'(t)|}{(x'^2(t) + y'^2(t))^{3/2}}$$

При явному заданні кривої  $y = y(x)$

$$k(x) = \frac{|y''(x)|}{(1 + y'(x)^2)^{3/2}}$$

Із параметричних рівнянь для кола  $x = R\cos t$ ,  $y = R\sin t$  знаходимо

$$\begin{aligned} x'(t) &= -R\sin t, & y'(t) &= R\cos t, \\ x''(t) &= -R\cos t, & y''(t) &= -R\sin t. \end{aligned}$$

Підставляючи знайдені похідні, знаходимо

$$k(t) = \frac{|x'(t)y''(t) - x''(t)y'(t)|}{(x'^2(t) + y'^2(t))^{3/2}} = \frac{1}{R}$$

Видно, що кривизна кола не залежить від параметра  $t$  і є постійною величиною.

### 27. Знайти кусково-лінійну інтерполяцію функції, яка задана таблично

$x_i$	-1.5	-1.0	-0.5	0.0	0.5	1.0	1.5
$f(x_i)$	-0.7	0.0	0.7	1.0	0.7	0.0	-0.7

Знайдемо наближене значення функції в точці  $x = -1.3$ , використовуючи лінійну інтерполяцію. Точка  $x = -1.3$  знаходиться між фіксованими значеннями  $x_0 = -1.5$ ,  $x_1 = -1$ . Використовуючи табличні дані для цього відрізка з рівняння прямої  $x_M = x_A + (y_M - y_A) \frac{x_B - x_A}{y_B - y_A}$ , знаходимо  $F(-1.3) = -0.42$ .

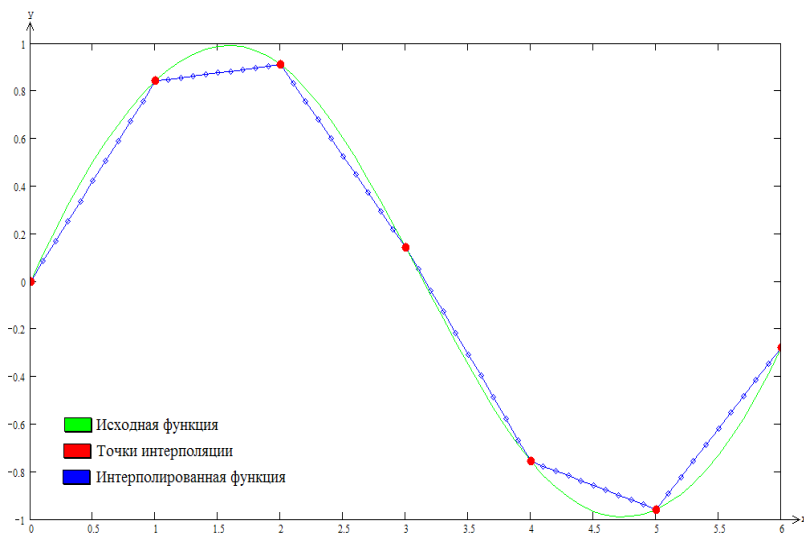


Рисунок 1.16 – Кусково-лінійна функція

### 28. Знайти рівняння кривої

Нехай крива  $C$  описується у вигляді деякої, невідомої нам квадратичної функції

$$C : y = ax^2 + bx + c$$

$$y(1) = 1, y(2) = 0, y(4) = 4;$$

Тоді, оскільки  $C$  проходить через точки  $A_1, A_2, A_3$ , то після підстановки координат цих точок у рівняння кривої, отримуємо такі тотожності



$$\begin{cases} y(1) = 1, \\ y(2) = 0, \\ y(4) = 4. \end{cases}$$

Тому

$$\begin{cases} a + b + c = 1, \\ 4a + 2b + c = 0, \\ 16a + 4b + c = 4. \end{cases}$$

Необхідно вирішити лінійну систему, що складається з трьох рівнянь щодо трьох змінних  $a, b, c$ . Скористаємося для її вирішення методом Гаусса

$$\left( \begin{array}{ccc|c} 1 & 1 & 1 & 1 \\ 4 & 2 & 1 & 0 \\ 16 & 4 & 1 & 4 \end{array} \right) \sim \left( \begin{array}{ccc|c} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & -4 \\ 0 & 0 & 1 & 4 \end{array} \right)$$

У результаті отримуємо,  $a = 1, b = -4, c = 4$ .

Таким чином, крива має таке рівняння:  $C : y = x^2 - 4x + 4$ .

**29. Методом ЦДА розкласти в растр відрізок з точки  $(0, 0)$  до точки  $(9, 4)$ . Подати результати покрокового виконання алгоритму.**

**Початкові умови  $x_1 = 0, y_1 = 0, x_2 = 9, y_2 = 4$ , довжина  $= 9, \Delta x = 9, \Delta y = 4$**

Результати ітераційного процесу подаємо у вигляді нижченаведеної таблиці.

$i$ , Крок	$x$	$y$	Plot $(x, y)$ , Відображені точки
0	0,5	0,5	(0, 0)
1	1,5	17/18	(1, 0)
2	2,5	25/18	(2, 1)
3	3,5	33/18	(3, 1)
4	4,5	41/18	(4, 2)
5	5,5	49/18	(5, 2)
6	6,5	57/18	(6, 3)
7	7,5	65/18	(7, 3)
8	8,5	73/18	(8, 4)
9	9,5	81/18	(9,4)

На рис. 1.17 наведено результат роботи алгоритму.

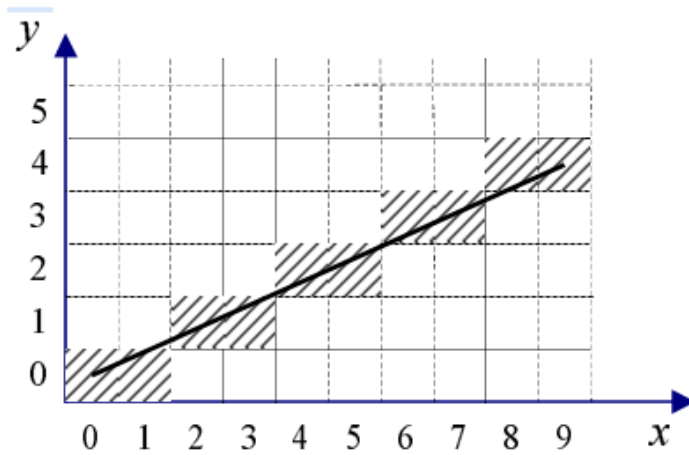


Рисунок 1.17 – Результат розкладання відрізка в растр із використанням методу цифрового диференціального аналізатора

**30. Розкласти в растр відрізок з точки  $(0, 0)$  у точку  $(9, 4)$ , використовуючи алгоритм Брезенхема. Подати результати покрокового виконання алгоритму для випадку  $\Delta x = 9, \Delta y = 4$**

Граф-схему алгоритму наведено на рисунку 1.18.

Початкові дані:  $x_1 = 0, y_1 = 0, x_2 = 9, y_2 = 4, e = 2\Delta y - \Delta x = 2 \cdot 4 - 9 = -1$

Результати ітераційного процесу подаємо у вигляді такої таблиці

$i$ , крок	$e$	$x$	$y$	$e'$	Plot $(x, y)$ , точки на екрані
0		0	0	-1	(0, 0)
1	-1	1	0	-7	(1, 0)
2	7	2	1	-3	(2, 1)
3	-3	3	1	5	(3, 1)
4	5	4	2	-5	(4, 2)
5	-5	5	2	3	(5, 2)
6	3	6	3	-7	(6,3)
7	-7	7	3	1	(7,3)
8	1	8	4	-9	(8,4)
9	-9	9	4	-1	(9,4)

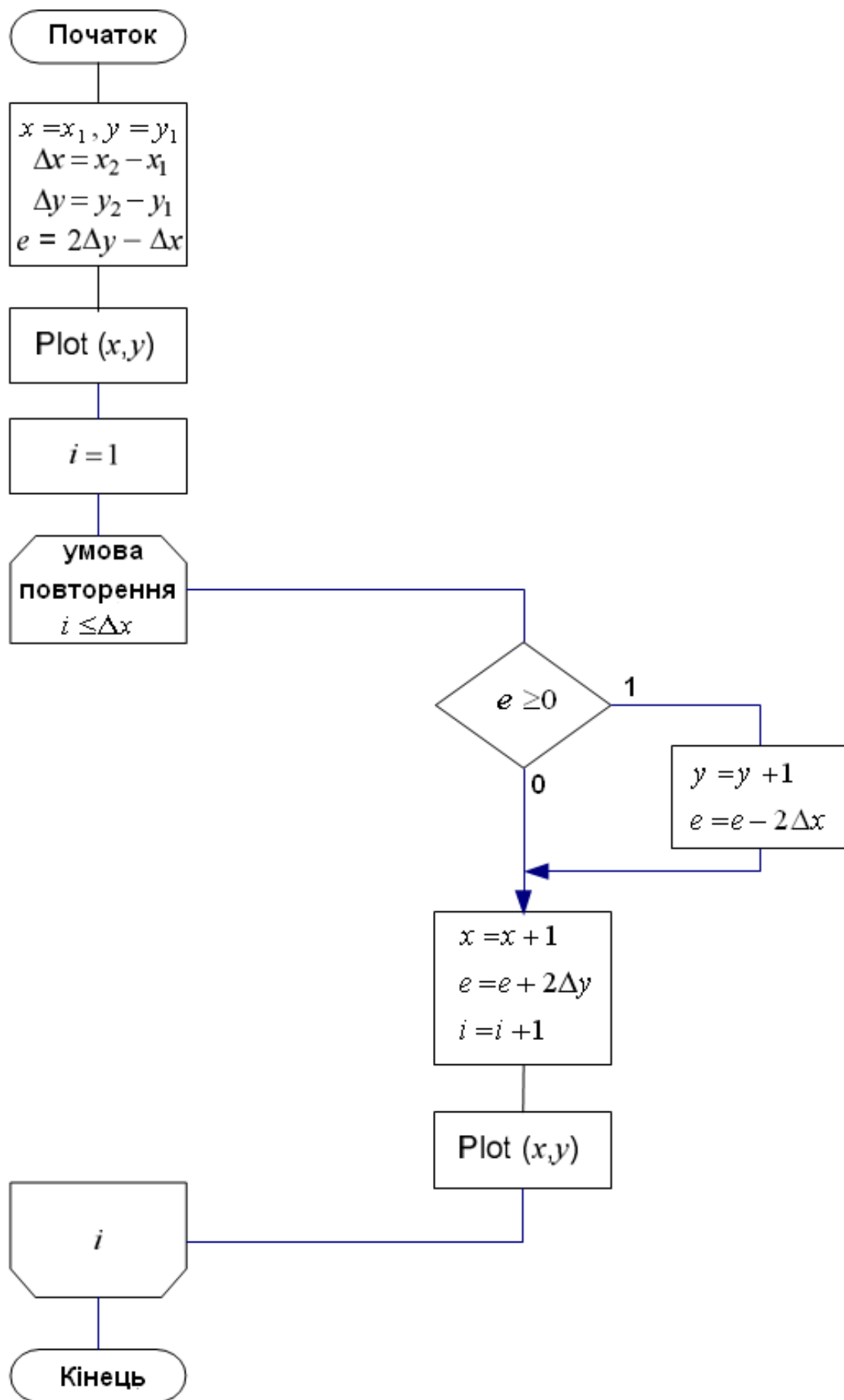


Рисунок 1.18 – Граф-схема алгоритму Брезенхема

Результат розкладання відрізка в растр із використанням алгоритму Брезенхема наведено на рис. 1.19.

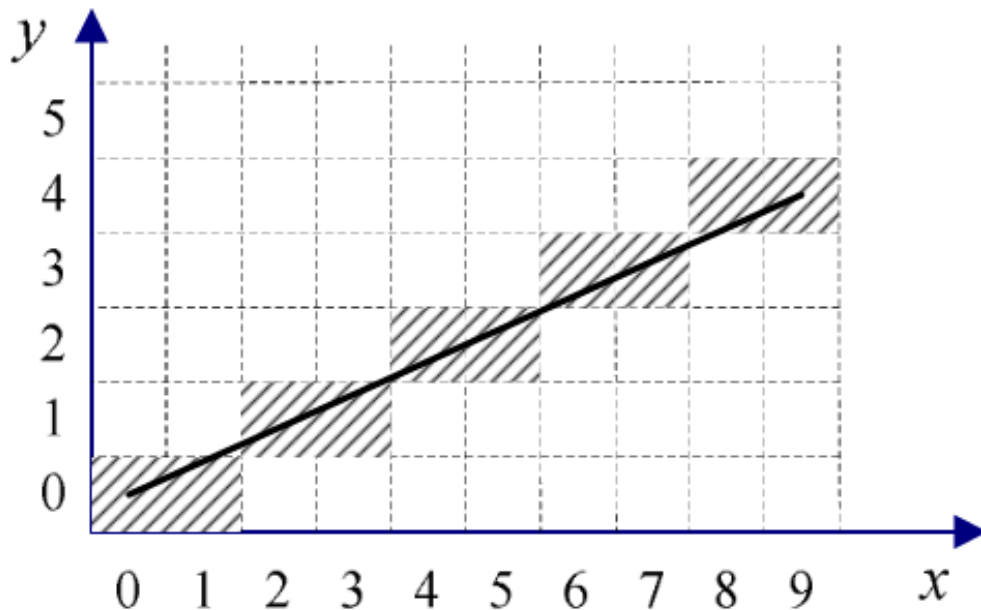


Рисунок 1.19 – Результат розкладання відрізка в растр із використанням алгоритму Брезенхема

**31. Приймаючи крок  $h = 0,05$ , побудувати інтерполяційний поліном Ньютона для функції  $y = 2 - 3x + 4x^2 - 6x^3$ , заданої таблицею**

$x$	1	1,05	1,1	1,15	1,2	1,25	1,3
$y$	-3	-3,685	-4,445	-5,285	-6,207	-7,218	-8,321

Для практичного використання інтерполяційну формулу Ньютона записують у вигляді

$$P_n(x) = y_0 + q\Delta y_0 + \frac{q(q-1)}{2!}\Delta^2 y_0 + \dots + \frac{q(q-1)\dots(q-n+1)}{n!}\Delta^n y_0, \quad (1)$$

де  $q = \frac{x - x_0}{h}$  – число кроків, необхідних для досягнення точки  $x$ , виходячи з точки  $x_0$ .

Складаємо таблицю різниць.

$x$	$y$	$\Delta y$	$\Delta^2 y$	$\Delta^3 y$
1	-3	0,685	-0,075	0,005
1,05	-3,685	0,76	-0,08	0,002
1,1	-4,445	0,84	-0,082	0,007
1,15	-5,285	0,922	-0,089	0,003
1,2	-6,207	1,011	-0,092	
1,25	-7,218	1,103		
1,3	-8,321			

Оскільки різниці третього порядку практично постійні, то у формулі Ньютона вважаємо  $n = 3$ . Приймаючи, матимемо:

$$P_n(x) = -3 + 0.685q - 0.075 \frac{q(q-1)}{2} + 0.005 \frac{q(q-1)(q-2)}{6},$$

$$P_n(x) = -3 + 0.685q - 0.075 \frac{q(q-1)}{2} + 0.005 \frac{q(q-1)(q-2)}{6}$$

або

$$P_n(x) = -3 + 0.685q - 0.0375q(q-1) + 0.00083q(q-1)(q-2),$$

Це і шуканий інтерполяційний поліном Ньютона.

### **32. Навести структурну схему підключення інтерполятора до відеопам'яті**

Інтерполяційні пристрої формують адреси точок траєкторії в дискретному координатному просторі. Вони підключаються до координатних лічильників, які формують адреси точок відеопам'яті. На рис. 1.20 наведено структурну схему відключення інтерполятора до відеопам'яті.

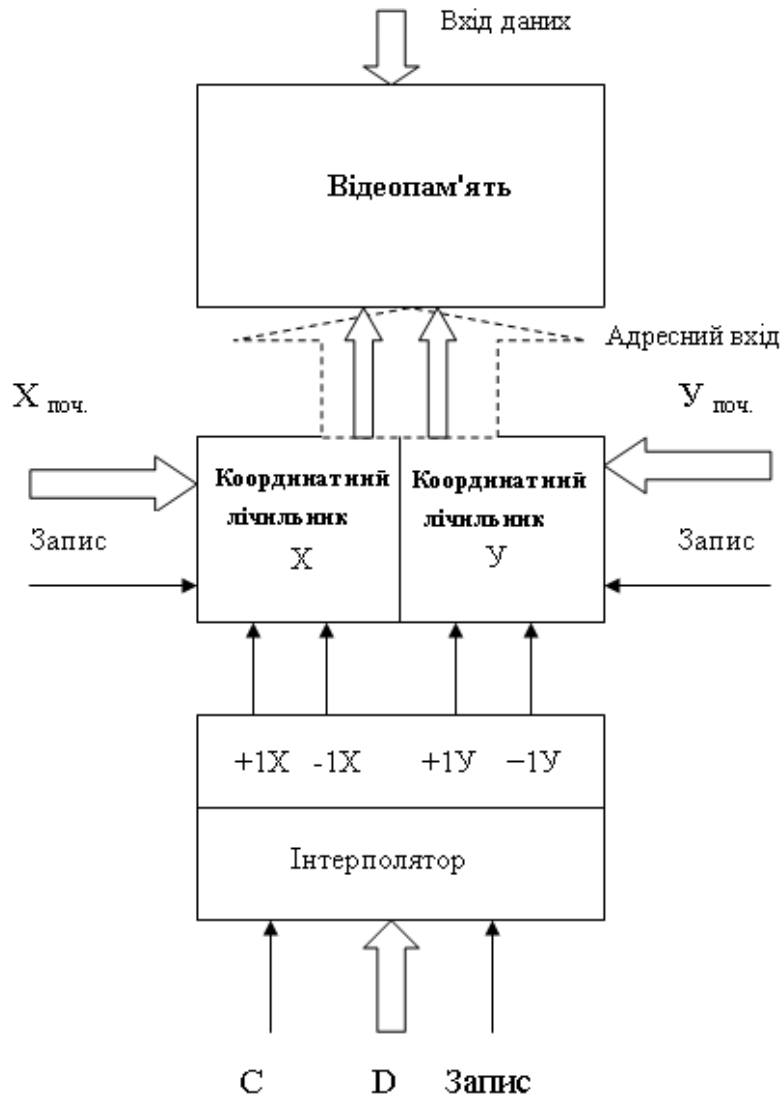


Рисунок 1.20 – Структурна схема підключення інтерполятора до відеопам'яті

### 33. Навести структурну схему параметричного лінійного інтерполятора

Структурну схему параметричного лінійного інтерполятора наведено на рис. 1.21.

Інтерполятор містить два регістри для зберігання приростів заданого вектора, які є керувальними кодами для двійкового помножувача, а також два двійкових помножувача. Останні за  $2^n$  тактів формують на своїх виходах кількість імпульсів, які дорівнюють заданим приростам.

На рис.1.22 наведено приклад формування параметричним лінійним інтерполятором відрізка прямої з  $\Delta X = 5$ ,  $\Delta Y = 3$ .

Наведений параметричний лінійний інтерполятор формує відрізки прямої, незалежно від їх довжини, за  $2^n$  тактів, що не є доцільним. Час інтерполювання можна зменшити за рахунок нормалізації – одночасного збільшення вихідних приростів в задане число разів. При цьому цикл інтерполювання зменшують у стільки ж разів.

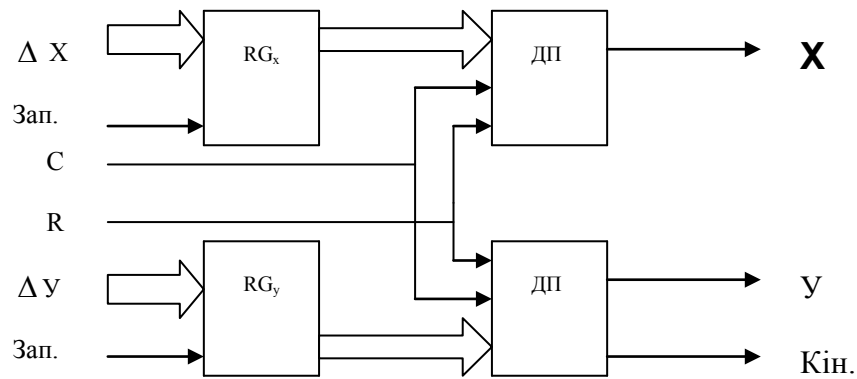


Рисунок 1.21 – Структурна схема параметричного лінійного інтерполятора

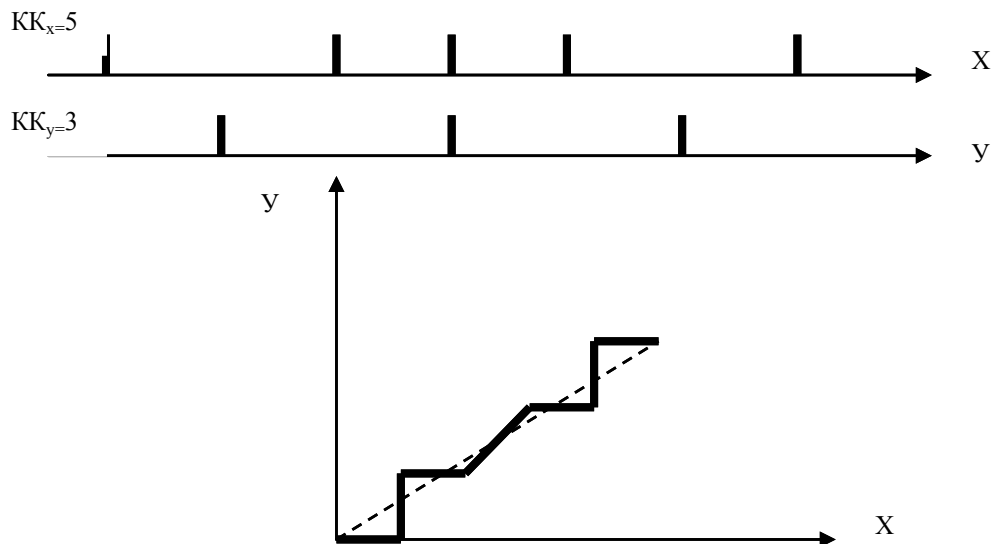


Рисунок 1.22 – Приклад формування відрізка прямої параметричним лінійним інтерполятором

### 34. Побудувати фрактал Коха та знайти його довжину

Крива Коха є типовим геометричним фракталом. Процес її побудови виглядає так (рис. 1.23): беремо одиничний відрізок, розділяємо на три рівні частини і замінюємо середній інтервал рівностороннім трикутником без цього сегмента. В результаті утворюється ламана, що складається з чотирьох ланок довжини  $1/3$ . На наступному кроці повторюємо операцію для кожної з чотирьох ланок, що вийшли, і т. д. Гранична крива і є крив Коха.

Довжина кривої Коха описується виразом

$$P_n = s \cdot \left(\frac{4}{3}\right)^n$$

де  $n$  – кількість ітерацій,  $s$  – довжина початкового відрізка

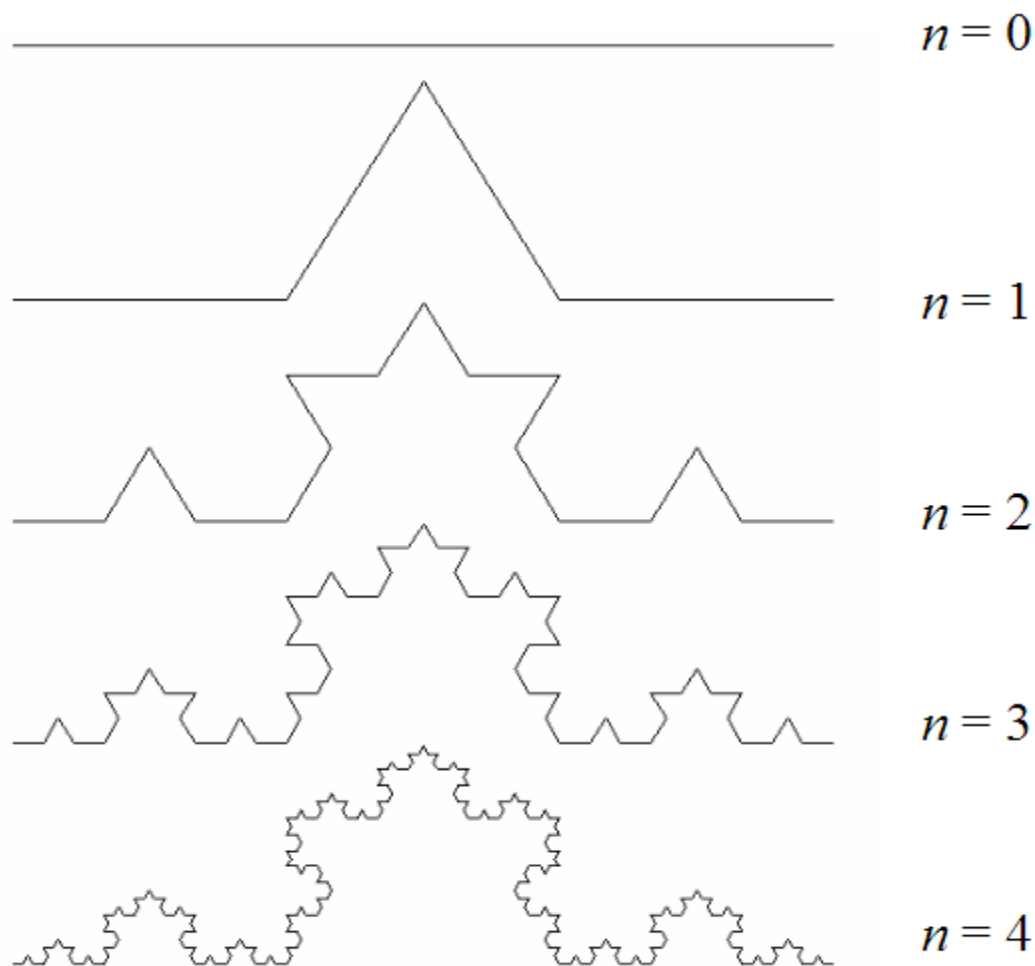


Рисунок 1.23 – Побудова фрактала Коха

**35. Побудувати інтерполяційний багаточлен Ньютона для функції  $f(x) = \sqrt[3]{x}$  с з вузлами інтерполяції  $x_0 = 1, x_1 = 2, x_2 = 3$**

Вузлів інтерполяції три, отже, багаточлен буде другого порядку. Обчислимо відповідної різниці.

$i$	$x_i$	$y_i$	$\Delta y_i$	$\Delta^2 y_i$
0	1	1,000	0,260	-0,78
1	2	1,260	0,182	
2	3	1,442		



Підставимо їх у формулу (1) і, враховуючи, що  $h = 1$ , одержимо

$$P_2(x) = 1 + 0,260(x-1) - \frac{0,078}{2}(x-1)(x-2)$$

розкривши дужки і привівши подібні члени, одержимо інтерполяційний багаточлен Ньютона

$$P_2(x) = -0,039x^2 - 0,377x + 0,662$$

Ввівши змінну

$$t = \frac{(x-x_0)}{h}$$

формула набуде іншого вигляду. Оскільки

$$x - x_0 = th, \quad x - x_1 = x - x_0 - h = th - h = (t-1)h, \quad x - x_2 = (t-2)h, \\ \dots, \quad x - x_{n-1} = (t - (n-1))h,$$

то, підставивши отримані результати, одержимо

$$P_n(x) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!} \Delta^2 y_0 \dots + \frac{t(t-1)(t-n+1)}{n!} \Delta^n y_0$$

**36. Записати рівняння дотичної до кривої  $u = v + 1$  на поверхні  $x = u \cos v, y = u \sin v, z = 2u$  в точці  $M(u = 1, v = 0)$**

Знайдемо  $\vec{r}_u$  і  $\vec{r}_v$

$$\vec{r}_u = (x'_u, y'_u, z'_u) = (\cos v, \sin v, 2).$$

У точці  $M$   $\vec{r}_u = (1, 0, 2)$ .

$$\vec{r}_v = (x'_v, y'_v, z'_v) = (-u \sin v, u \cos v, 0)$$

У точці  $M$   $\vec{r}_v = (0, 1, 0)$ .

$$d\vec{r} = du(\vec{r}_u + \vec{r}_v \frac{dv}{du}) = du(\vec{r}_u + \vec{r}_v),$$

бо з рівняння кривої маємо  $du = dv$  і  $\frac{dv}{du} = 1$ .

Підставимо координати векторів  $\vec{r}_u$  і  $\vec{r}_v$ . Одержимо  $d\vec{r} = du(1, 1, 2)$ . Рівняння дотичної до кривої в точці  $M(1, 0, 2)$  запишеться у вигляді

$$\frac{x-1}{1} = \frac{y-0}{1} = \frac{z-2}{2} = t, \quad \text{або} \quad x = t+1, \quad y = t, \quad z = 2t+2.$$

**37. Записати рівняння обвідної сім'ї поверхонь  $x+c^2y+z-2c=0$**

Диференціюємо за  $c$   $F'_c = 2cy - 2 = 0$ . Позбуваємось  $c$ ,  $c = \frac{1}{y}$  і підставимо в рівняння сім'ї  $x + \frac{1}{y} + z - 2\frac{1}{y} = 0$ , або  $xy + yz - 1 = 0$ .

**38. Записати рівняння характеристик сім'ї поверхонь  $x+c^2y+z-2c=0$**

Диференціюємо за параметром  $c$   $F'_c = 2cy - 2 = 0$ . При фіксованому  $c = c_0$  два рівняння  $x+c_0^2y+z-2c_0=0$  і  $2c_0y-2=0$ , які є рівняннями площин, визначають характеристику – пряму їх перетину. Її напрямний вектор має координати  $(-c_0, 0, c_0)$ , або  $(-1, 0, 1)$ .

**39. Навести приклад побудови кубічної кривої Безьє за алгоритмом розбиттям de Casteljau**

Нехай маємо криву, побудовану за точками 1-2-3-4 (рис. 1.24).

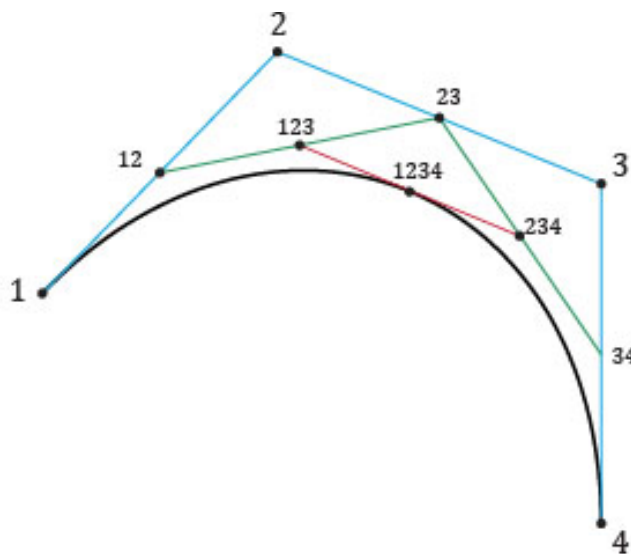


Рисунок 1.24 – Побудова кривої Безьє

Виберемо певний параметр  $t^*$  (для рисунка  $t^* = 0.5$ ) і знайдемо точки на відрізках 1-2, 2-3 та 3-4, які відповідають лінійній інтерполяції з параметром  $t$ . Назвемо точки 12, 23, 34 відповідно.

Потім повторимо операцію на відрізках 12-23, 23-34, знайшовши точки 123 і 234. І, нарешті, побудувавши лінійну інтерполяцію за  $t$  для відрізка 123-234, отримаємо точку 1234, що лежить на кривій Безьє, а криві, побудовані - 12-123-1234 та 1234-234-34-4, повторюють оригінальну криву.

## 2. ПРОЦЕДУРИ КОМП'ЮТЕРНОЇ ГРАФІКИ

**1. Визначити тип трикутника, який найбільш доцільний для рендерингу Гуро з точки зору обчислювальних затрат та точності розрахунку інтенсивностей кольору пікселів**

Попереднім етапом рендерингу Гуро є триангуляція ділянки, обмеженою полігоном.

Згідно з рендерингом зафарбування виконується між ребрами АВ і АС (рис. 2.1).

Якщо відрізок АС паралельний осі абсцис, то відпадає необхідність в рендерингу нижнього ребра.

Приріст інтенсивності кольору вздовж ребра АВ визначається за функцією

$$\Delta I_{AB} = \frac{I_A - I_B}{BP_{AB}},$$

де  $BP_{AB}$  – більший з приростів ребра АВ.

$BP_{AB} = AB$  за умови, що відрізок АВ паралельний осі ординат.

Таким чином, найбільш доцільним для рендерингу Гуро є прямокутний трикутник, катети якого паралельні осям координат.

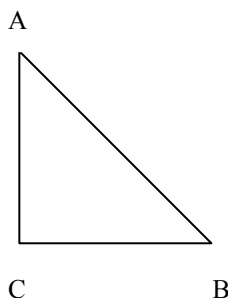


Рисунок 2.1 – Найдоцільніший трикутник для рендерингу

**2. Визначити кількість оптоелектронних пар передекранної сенсорної панелі для забезпечення режиму точного позиціювання за умови, що використовується екран стандарту Artist 1+.**

Передекранна сенсорна панель містить пари оптоелектронних елементів, які розміщені по периметру екрана. Кожна з таких пар має джерело світла та його приймачі.

При дотику оператором передекранної панелі виконується введення в ЕОМ координат X і Y положення об'єкта на екрані.

Режиму Artist 1+ відповідає адресний простір 1024\*768 точок.

Позначимо через  $n$  кількість оптоелектронних пар, яку необхідно забезпечити для однієї зі сторін екрана, яка охоплює  $N$  точок.

Тоді  $N/n$  визначає розмір макрозони для режиму позиціювання.

Для забезпечення ідентифікації кожної точки макрозони відношення  $N/n$  має дорівнювати  $n$ , тобто  $N/n = n$ .  $N = n^2$ .

$$\text{Звідси } n = \lceil \sqrt{N} \rceil.$$

$$\text{При } N=1024, n=32.$$

$$\text{Якщо } N=68, n=29.$$

**3. Визначити типи чотирикутників, для яких при заповненні за критерієм зв'язності необхідно дві точки-затравки за умови, що виконується горизонтальна растеризація, а одну з точок-затравок розміщено в верхньому кутку чотирикутника**

Більш як 2 точки-затравки потрібно за умови, що багатокутник не є випуклим.

Можливі варіанти неопуклих чотирикутників наведено на рис. 2.2.

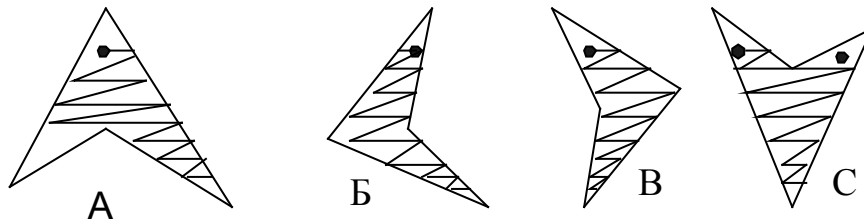


Рисунок 2.2 – Типи багатокутників

Як видно з рисунка, точки-затравки при горизонтальній растеризації потрібні для випадків А і С (рис. 2.2).

**4. Навести приклад появи точок-просікання при заповненні ділянок, обмежених полігоном**

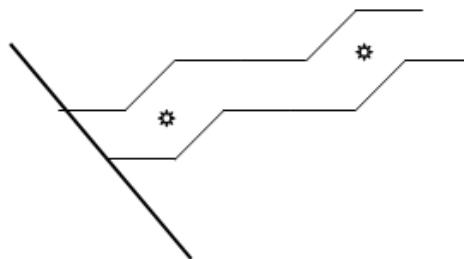


Рисунок 2.3 – Артефактні точки

Артефактні точки виникають у випадках зміщення діагональних кроків при неортогональній растеризації.

**5. Навести приклад рядка растеризації зі значеннями змінної для заповнення контуру при заповненні за критерієм парності**

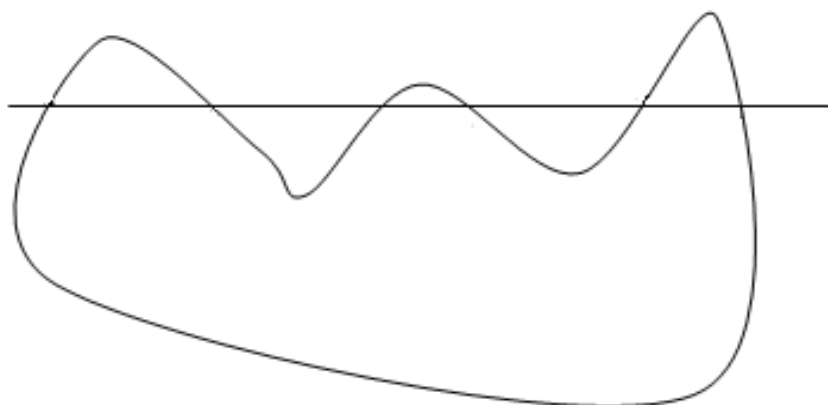


Рисунок 2.4 – Вихідний контур

Як правило, вводять допоміжну змінну з нульовим початковим станом. При кожному перетині контуру сканувальним відрізком прямої стан змінної змінюють на протилежний (рис. 2.5).

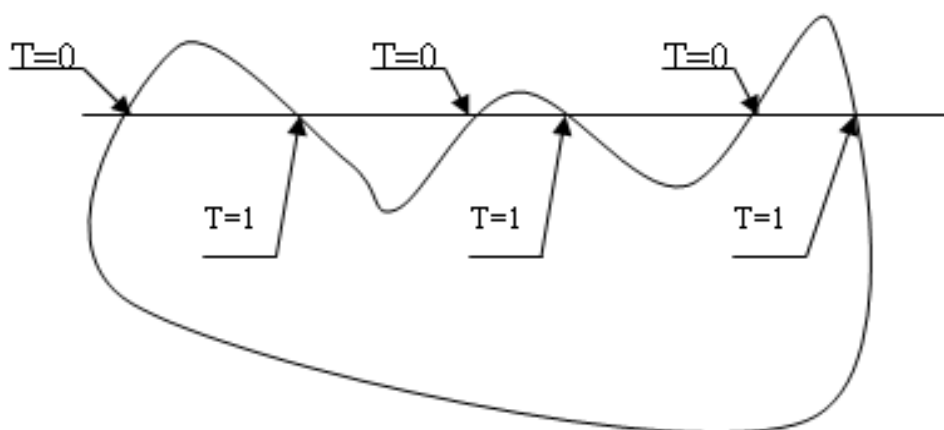


Рисунок 2.5 – Рядок растеризації

**6. Виконати опис полігональної сітки, наведеної на рис. 2.6 згідно з методом явного задання ребер**

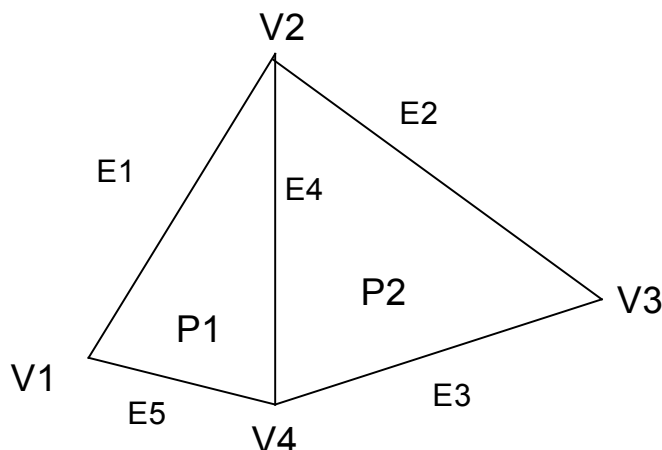


Рисунок 2.6 – Полігональна сітка

$$V = (V_1, V_2, V_3, V_4) = ((X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3), (X_4, Y_4, Z_4))$$

$$E_1 = (V_1, V_2, P_1, L)$$

$$E_3 = (V_3, V_4, P_2, L)$$

$$E_2 = (V_2, V_3, P_2, L)$$

$$E_4 = (V_1, V_2, P_1, P_2)$$

$$P_1 = (E_1, E_4, E_5)$$

$$P_2 = (E_2, E_3, E_4)$$

Метод забезпечує швидкий пошук спільних ребер, унеможливорює дублювання вершин при заданні багатокутників.

**7. Виконати опис полігональної сітки, зображеної на рисунку 2.6, шляхом явного задання багатокутників і з використанням покажчиків**

Опис полігональної сітки при явному заданні багатокутників має вигляд:

$$P_1 = (V_1, V_2, V_4) = ((X_1, Y_1, Z_1), (X_2, Y_2, Z_2), \dots (X_4, Y_4, Z_4))$$

$$P_2 = (V_2, V_3, V_4) = ((X_2, Y_2, Z_2), (X_3, Y_3, Z_3), \dots (X_4, Y_4, Z_4))$$

Явне задання багатокутників характеризується надлишковістю, оскільки для вершин  $V_2$  та  $V_4$  опис проведений двічі.

Опис логічної сітки при використанні покажчиків має вигляд

$$P_1 = (V_1, V_2, V_3, V_4) = ((X_1, Y_1, Z_1), (X_2, Y_2, Z_2), (X_3, Y_3, Z_3), (X_4, Y_4, Z_4))$$

$$P = (1, 2, 4) \quad P = (4, 2, 3)$$

При заданні  $P$  цифр в дужках – є покажчики вершин в описі  $V$ .

**8. Виконати відсікання відрізка прямої, зображеного на рисунку 2.7, згідно з алгоритмом ділення відрізка навпіл**

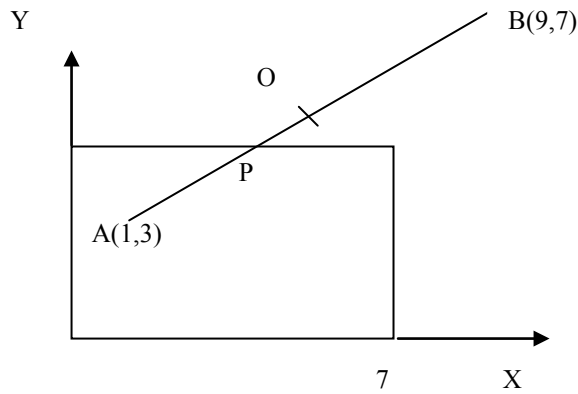


Рисунок 2.7 – Відсікання відрізка

Знаходимо координати точки O

$$X_0 = X_A + \frac{X_B - X_A}{2} = 1 + \frac{9-1}{2} = 5,$$

$$Y_0 = Y_A + (Y_B - Y_A)/2 = 3 + (7-3)/2 = 5.$$

Відрізок OB, згідно з алгоритмом Коена-Сазерленда, відкидається.  
Знаходимо координати точки P

$$X_p = X_A + (X_0 - X_A)/2 = 1 + (5-1)/2 = 3,$$

$$Y_p = Y_A + (Y_0 - Y_A)/2 = 3 + (5-3)/2 = 4.$$

Відрізок OP згідно алгоритму Коена-Сазерленда відкидається.  
Відрізок AP належить вікну.

**9. Виконати відсікання відрізків прямих, зображених на рисунку 2.8, згідно з алгоритмом Коена-Сазерленда**

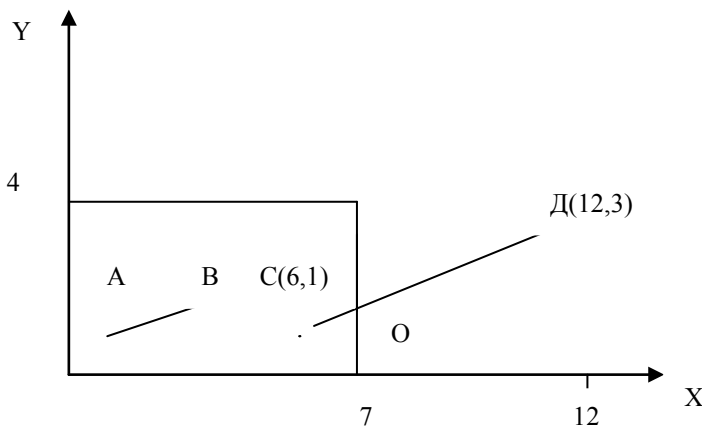


Рисунок 2.8 – Відрізки прямих

Визначимо чотирирозрядні коди для точок А та В за такими ознаками:

	Вище	Нижче	Лівише	Правіше
А	0	0	0	0
В	0	0	0	0

Оскільки чотирирозрядні коди точок А і В нульові, то відрізок знаходиться всередині вікна.

Для точок С і Д чотирирозрядні коди дорівнюють: точка С – 0000, точка Д – 0001.

Оскільки операція порозрядного логічного добутку дає нульовий результат, то необхідно виконати відсікання.

Відрізок прямої СД задаємо рівнянням виду:

$$Y = 1 + 2/6 * (x - 6) = 1 + 1/6 (x - 6).$$

Максимальна абсциса границі вікна дорівнює 7. Знаходимо Y для x = 7.

$$Y = 1 + 1/6 = 7/6.$$

Точка О має координати (7, 7/6). Відрізок ОД відкидається.

**10. Трикутник з вершинами А(2, 1), В(2, 3), С(4, 1) повертається на кут  $\pi/2$  щодо початку координат. Обчислити координати підсумкового трикутника А'В'С'**

Перейдемо до проєктивних координат А = (2, 1, 1), В = (2, 3, 1), С = (4, 1, 1) і скористаємося матричним поданням проєктивного обертання

$$A' = R_{\pi/2}(A) = \begin{pmatrix} 1 & -1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 2 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} -1 \\ 2 \\ 1 \end{pmatrix}.$$

Аналогічно - В ' = (-3, 2, 1), С' = (-1, 4, 1). Отже, повертаючись до декартових координат, маємо: А '(-1, 2), В'(-3, 2), С'(-1, 4).

**11. Сформувати полігональної сітку з чотирьох багатокутників з дев'ятьма вершинами та дванадцятьма ребрами та навести опис**

На рисмунку 2.9 зображено простий приклад полігональної сітки з чотирьох багатокутників з дев'ятьма вершинами та дванадцятьма ребрами. Розглянуто кілька різних подань.



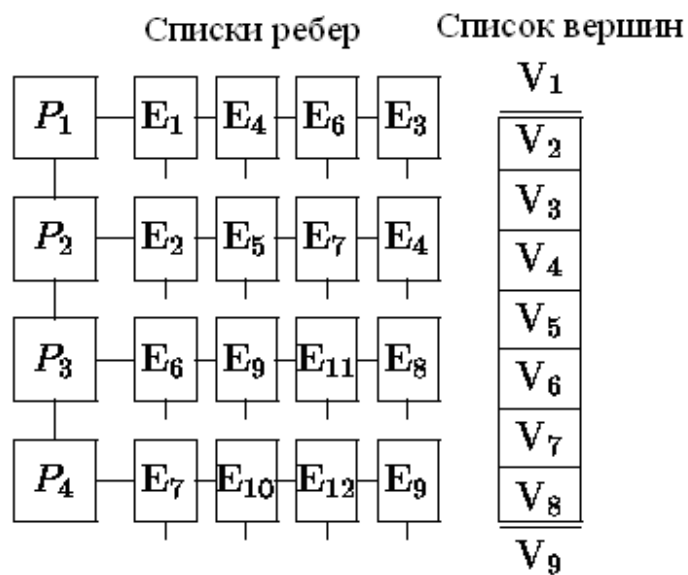
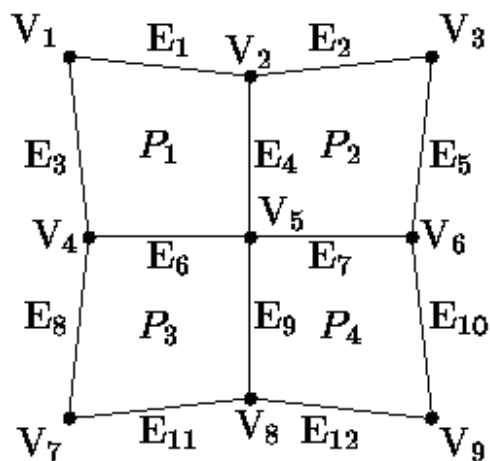


Рисунок 2.9 – Подання полігональної сітки

Елементи списку ребер містять покажчики на вершини у списку вершин, що утворюють дане ребро. Для забезпечення пошуку всіх вершин, що утворюють даний багатокутник, потрібно мати зворотні покажчики від вершини до одного з інцидентних до неї ребер.

**12. Визначити відстань між точкам екрана, при якій афekt аліайзингу невидимий для спостерігача**

При формуванні зображення на стадії растеризації виникають спотворення, зумовлені дискретним характером формування зображень і недостатньою роздільною здатністю пристроїв відображення. На краях об'єктів з'являються яскраво виражені сходинки або зубці. Даний артефакт отримав назву ступінчастого ефекту чи ефекту аліайзингу. Ефект аліайзингу погіршує якість сформованого зображення, що передбачає розробку спеціальних методів і

засобів його усунення. Дослідження показали, що при використанні 17'' монітора та розміщенні спостерігача на відстані 65 см від екрана для повного усунення ефекту аліайзингу потрібен монітор з роздільною здатністю як мінімум 4000 × 4000 пікселів.

Діагональ монітора 17 дюймов. 1 дюйм = 2.54 сантиметра. Отже, діагональ монітора 43,18 сантиметра.

Знайдемо розмір сторін монітора за формулою  $a^2 + a^2 = b^2$ , де  $a$  – сторона монітора, а  $b$  – його діагональ.

$$a = \sqrt{\frac{b^2}{2}} = \sqrt{\frac{(43,18)^2}{2}} = 30,52 \text{ см}$$

Відстань між точками екрана дорівнює  $\frac{30,52 \text{ см}}{4000} = 0,00736 \text{ см}$ .

**13. Нанести фактуру на октант сфери. Візерунок – проста сітка з прямих, які перетинаються (рис. 2.10)**

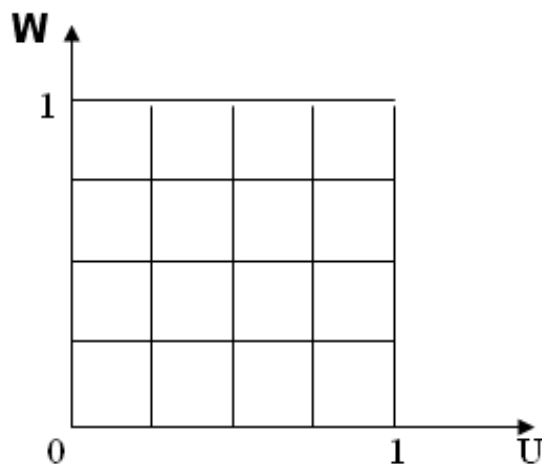


Рисунок 2.10 – Вигляд візерунка

Параметричне подання октанта сфери

$$X = \sin Q \sin \varphi,$$

$$Y = \cos \varphi,$$

$$Z = \cos Q \sin \varphi.$$

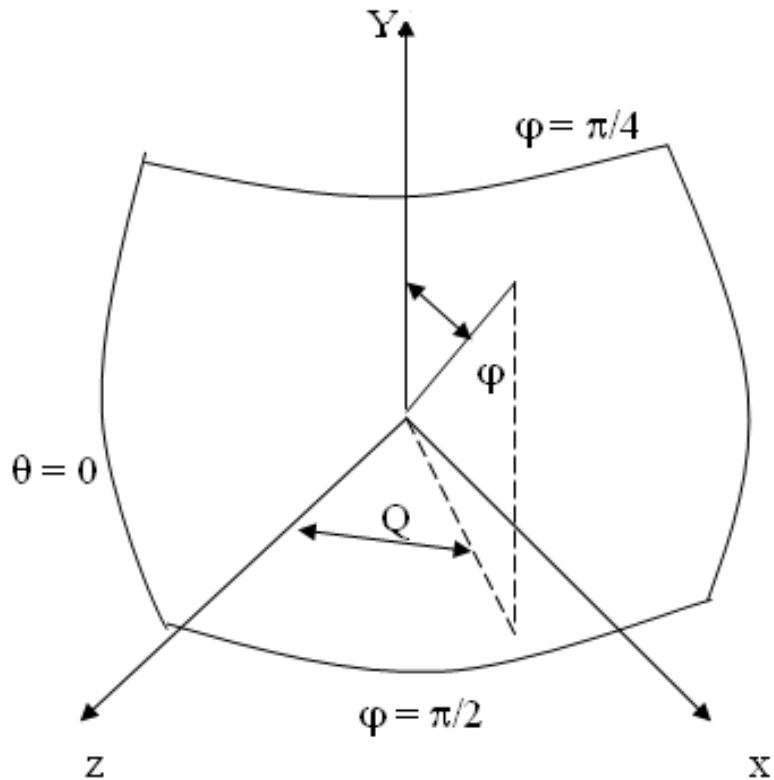


Рисунок 2.11 – Півсфера

Нехай функція відображення лінійна, тобто,

$$Q = Au + B, \quad \varphi = Cw + D$$

і кути візерунка переходять в кути октанта

$$u = 0, w = 0 \text{ при } Q = 0, \varphi = \pi/2,$$

$$u = 1, w = 0 \text{ при } Q = \pi/2, \varphi = \pi/2,$$

$$u = 0, w = 1 \text{ при } Q = 0, \varphi = \pi/4,$$

$$u = 1, w = 1 \text{ при } Q = \pi/2, \varphi = \pi/4.$$

Звідки  $A = \pi/2, B = 0, C = -\pi/4, D = \pi/2$ , тобто,  $Q = u\pi/2, \varphi = \pi/2 - \pi w/4$ .

Обернене перетворення має такий вигляд:  $u = Q/(\pi/2), w = (\pi/2 - \varphi)/\pi/4$ .

У цьому методі рисунок наноситься на рівну поверхню, і вона після цього залишається рівною. Для того, щоб поверхня здавалася нерівною, можна відцифрувати фотографію нерегулярною фактурою та відобразити її на поверхню.

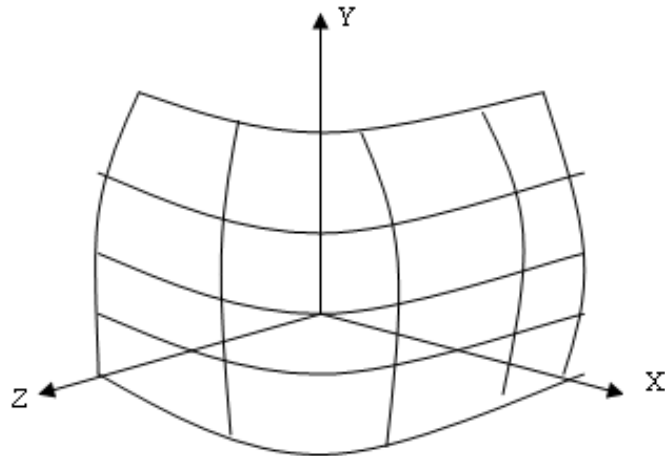


Рисунок 2.12 – Поверхня з візерунком

**14. Побудувати матрицю обертання на кут  $\phi$  навколо прямої  $L$ , що проходить через точку  $A(a, b, c)$  і має напрямний вектор  $(l, m, n)$ . Можна вважати, що напрямний вектор прямої є одиничним:  $l^2 + m^2 + n^2 = 1$**

Розв'язання задачі розбивається на кілька етапів. Опишемо послідовно кожен із них.

**1-й крок.** Перенесення на вектор  $-A(-a, -b, -c)$  за допомогою матриці

$$[T] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -a & -b & -c & 1 \end{bmatrix}.$$

Внаслідок цього перенесення важливо, щоб пряма  $L$  проходила через початок координат.

**2-й крок.** Поєднання осі аплікат з прямою  $L$  двома поворотами навколо осі абсцис та осі ординат.

1-й поворот – навколо осі абсцис на кут  $\psi$  (що підлягає визначенню). Щоб знайти цей кут, розглянемо ортогональну проєкцію  $L$  вихідної прямої  $L$  на площину  $X = 0$ .

Напрямний вектор прямої  $L'$  визначається просто – він дорівнює  $(0, m, n)$ .

Звідси ж випливає, що  $\cos \psi = \frac{n}{d}$ ,  $\sin \psi = \frac{m}{d}$ , де  $d = \sqrt{m^2 + n^2}$ .

Відповідна матриця обертання має такий вигляд:

$$[R_x] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{n}{d} & \frac{m}{d} & 0 \\ 0 & -\frac{m}{d} & \frac{n}{d} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Під дією перетворення, що описується цією матрицею, координати вектора  $(l, m, n)$  зміняться. Підрахувавши їх, у результаті отримаємо  $(l, m, n, 1)[R_x] = (l, 0, d, 1)$ .

2-й поворот – навколо осі ординат на кут  $\theta$ , що визначається співвідношеннями  $\cos \theta = l, \sin \theta = -d$ .

Відповідна матриця обертання записується у такому вигляді:

$$[R_y] = \begin{bmatrix} 1 & 0 & d & 0 \\ 0 & 1 & 0 & 0 \\ -d & 0 & l & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**3-й крок.** Обертання навколо прямої L на заданий кут  $\varphi$ .

Оскільки тепер пряма L збігається з віссю абсцис, то відповідна матриця має такий вигляд:

$$[R_z] = \begin{bmatrix} \cos \varphi & \sin \varphi & 0 & 0 \\ -\sin \varphi & \cos \varphi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

**4-й крок.** Поворот навколо осі ординат на кут  $-\theta$ .

**5-й крок.** Поворот навколо осі абсцис на кут  $-\psi$ .

Обертання у просторі некомутативно. Тому порядок, у якому проводяться обертання, є дуже суттєвим.

**6-й крок.** Перенесення на вектор  $A(a, b, c)$ .

Перемноживши знайдені матриці порядку їх побудови, отримаємо таку матрицю:  $[T] * [R_x] * [R_y] * [R_z] * [R_y]^{-1} * [R_x]^{-1} * [T]^{-1}$ .

Випишемо остаточний результат, вважаючи для простоти, що вісь обертання L проходить через початкову точку.

$$\begin{bmatrix} l^2 + \cos \varphi(1 - l^2) & l(1 - \cos \varphi)m + n \sin \varphi & l(1 - \cos \varphi)n - m \sin \varphi & 0 \\ l(1 - \cos \varphi)m - n \sin \varphi & m^2 + \cos \varphi(1 - m^2) & m(1 - \cos \varphi)n + l \sin \varphi & 0 \\ l(1 - \cos \varphi)n + m \sin \varphi & m(1 - \cos \varphi)n - l \sin \varphi & n^2 + \cos \varphi(1 - n^2) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

15. Сформувати чотирирозрядні коди для відрізків прямих, зображених на рис. 2.13

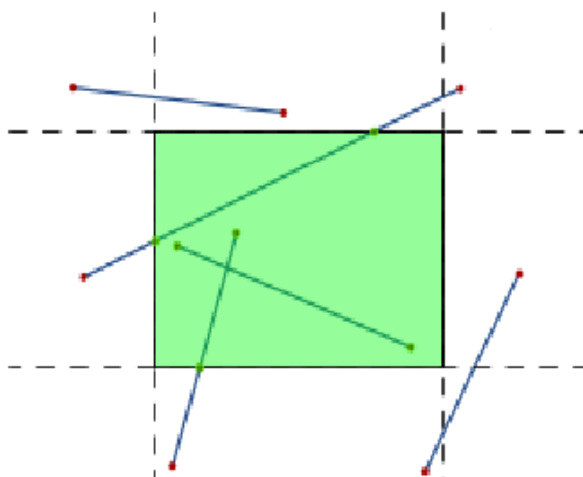


Рисунок 2.13 – Вихідні відрізки

Для перевірки на відсікання границі ділянок індикації проводять так, щоб вони ділили поле, на якому знаходиться зображення, на дев'ять підділянок. Кожній з них присвоюють чотирирозрядний код. Цей код присвоюють кінцевим точкам відрізка, який знаходиться у відповідних підділянках:

1001	1000	1010
0001	0000	0010
0101	0100	0110

Одиниці у відповідних розрядах коду означають: у першому – точка над верхнім краєм області індикації; у другому – точка під нижнім краєм ділянки індикації; у третьому – точка праворуч від області індикації; у четвертому – точка зліва від лівого краю області індикації.

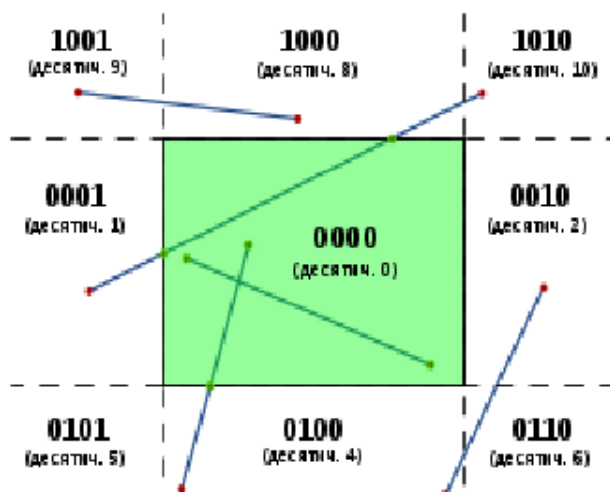


Рисунок 2.14 – Значення кодів

**16. Навести приклад контуру, де порушується принцип парності перетину зі сканувальним рядком**

Алгоритми заповнення ділянок, в яких застосовують перевірку на парність, базуються на тому, що довільна пряма перетинає будь-яку замкнену криву парну кількість разів. Якщо відомо, що перша точка відповідної лінії лежить за контуром, то, виконавши обхід цієї лінії, шляхом відрахувань кількості перетинів можна встановити, які саме її відрізки розміщені в ділянці. Якщо число перетинів непарне, то відповідний відрізок розміщений у внутрішній ділянці, в іншому випадку – поза ділянкою.

На рис. 2.15 для другого сканувального рядка кількість перетинів непарна.

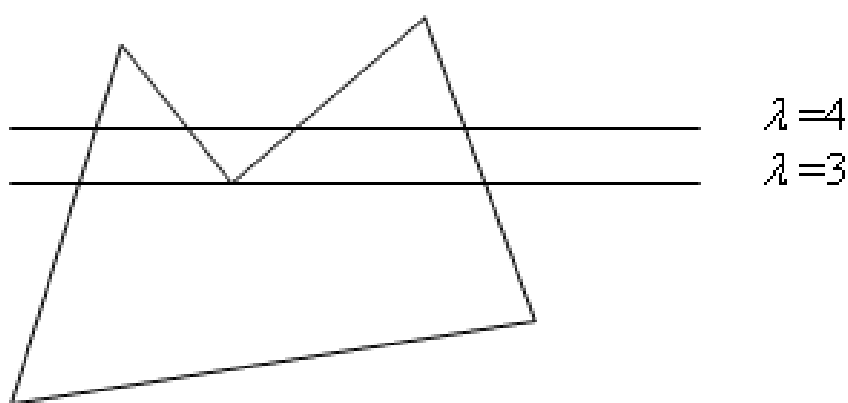


Рисунок 2.15 – Перетини контуру

**17. Навести алгоритм заповнення для частини контуру за критерієм парності, зображеного на рис. 2.16**

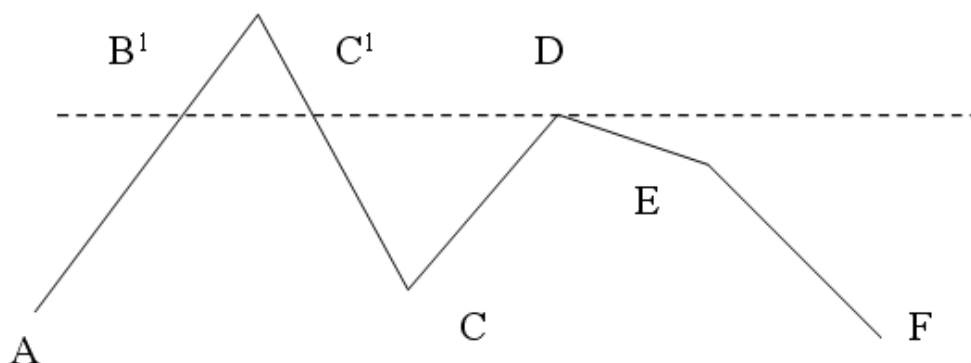


Рисунок 2.16 – Частина контуру

За умовою задано метод заповнення ділянки, обмеженої сторонами багатокутника, що використовує принцип парності. Будемо вважати, що сторони багатокутника задані координатами вершин багатокутника  $x_i, y_i$ , яким відповідає

максимальне значення  $y$  або мінімальне значення  $x$ , а також прирости  $\Delta x$ ,  $\Delta y$  для визначення координат іншої вершини.

*Першим* кроком алгоритму є сортування сторін відповідно до максимальних значень  $y_i$ , а при їхній рівності – за мінімумом  $x_i$ . У випадку рівності  $x_i$  використовуються значення  $\Delta x$ , а потім  $\Delta y$ . При цьому вибирається сторона з найменшим алгебраїчним значенням. У результаті одержуємо упорядкований список сторін контуру – список черговості.

*Другий* крок алгоритму виконується для всіх суміжних сторін списку з однаковими значеннями  $y$ , що відповідає локальному максимуму. Через вершину багатокутника, що відповідає цим сторонам, проводиться пряма, паралельна осі  $X$ , і визначаються точки її перетину з іншими сторонами, що розташовані в списку вище проаналізованих. Дані прямі поділяють контур на прошарки, що містять парне число меж. Кожному прошарку відповідає спеціальний список, названий поточним, що містить межі. Для формування поточного списку список черговості має мітки трьох типів.

Мітка *M1* визначає число меж, переданих зі списку черговості в поточний список: дві – якщо точка відповідає максимуму; одна – якщо вона не є максимумом; більше двох, якщо одному значенню  $y$  відповідає більше одного максимуму.

Мітка *M2* служить для введення в поточний список наступної пари меж при переході в наступний прошарок. Її значення відповідає значенню  $y$ , при котрому цей перехід повинен відбутися. Зазначеними мітками мітять перші сторони зі списку черговості, з якими перетинаються прямі, проведені через вершини багатокутника, що є локальними максимумами

Мітка *M3* визначає число нових меж, до яких відбувається звертання після завершення роботи з поточною межею. Таблиця ілюструє формування списку черговості і міток для одержання поточного списку на прикладі фрагмента області, наведеної на рис. 2.16.

Опис меж

Сторона:	$X$	$Y$	Мітка 1	Мітка 2	Мітка 3
$AB$	$X_B - X_A$	$Y_B - Y_A$	2	$Y_D$	1
$AC$	$X_C - X_A$	$Y_C - Y_A$	1	—	0
$DC$	$X_C - X_D$	$Y_C - Y_D$	2	—	0
$DE$	$X_E - X_D$	$Y_E - Y_D$	1	—	1
$EF$	$X_F - X_E$	$Y_F - Y_E$	1	—	1

*Третім* кроком алгоритму є безпосереднє заповнення ділянки послідовно для кожного прошарку. Цей процес починається з першої пари меж списку черговості шляхом формування та оновлення поточного списку, використовуючи наявні мітки, що дозволяють видаляти та вносити нові межі. Організація поточного списку дає можливість реалізувати для кожного прошарку принцип парності  $y$  у зв'язку з тим, що всі локальні максимуми визначені. Потрібно відзначити, що наведений алгоритм не враховує наявність горизонтальних сторін багатокутника. Проте модифікація алгоритму для урахування цих ситуацій не має істотних труднощів.



### 18. Навести у графічній формі різні варіанти подання площин

У тривимірній графіці широко використовують багатокутники («грань об'єкта»), багатокутник розташовується у певній площині.

Рисунок 2.17 демонструє кілька основних форм подання площини. У випадку (рис. 2.17, а) площина задається точкою А і вектором нормалі  $n$ , тоді рівняння площини можна записати у вигляді

$$x_n \cdot (x - x_a) + y_n \cdot (y - y_a) + z_n \cdot (z - z_a) = 0$$

У разі (рис. 2.17, б) площина задається трьома точками. Параметричне подання площини наведено у випадку (рис. 2.17, в), рівняння площини має вигляд

$$P(s, t) = A + b \cdot s + c \cdot t,$$

де  $s$  та  $t$  – параметри, задаючи які можна ідентифікувати відповідну точку на площині.

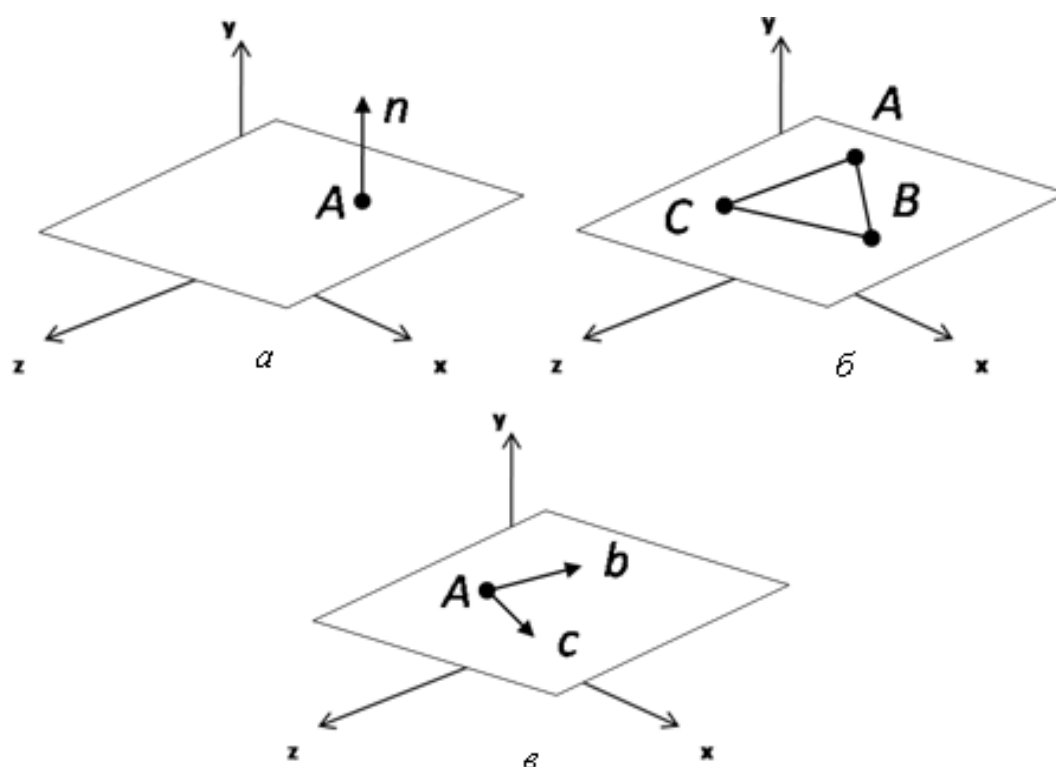


Рисунок 2.17 – Приклади задання площини

**19. Заповнити багатокутник (рис. 2.18), який задається замкненою ламаною лінією без самоперетинів**

Найпростіший спосіб заповнення багатокутника полягає у перевірці належності кожної точки цього багатокутника. Більш ефективні алгоритми використовують той факт, що сусідні піксели, ймовірно, мають однакові характеристики (крім пікселів граничних ребер). Ця властивість називається просторовою когерентністю.

У разі багатокутника когерентність пікселів визначається вздовж сканувального рядка. Сканувальні рядки, зазвичай, змінюються від «верху» багатокутника до його «низу». Характеристики пікселів змінюються лише там, де ребро багатокутника перетинає рядок. Ці перетини ділять сканувальний рядок на ділянки заповнених і незаповнених пікселів.

Розглянемо, які випадки можуть виникнути при розподілі на ділянки сканувального рядка.

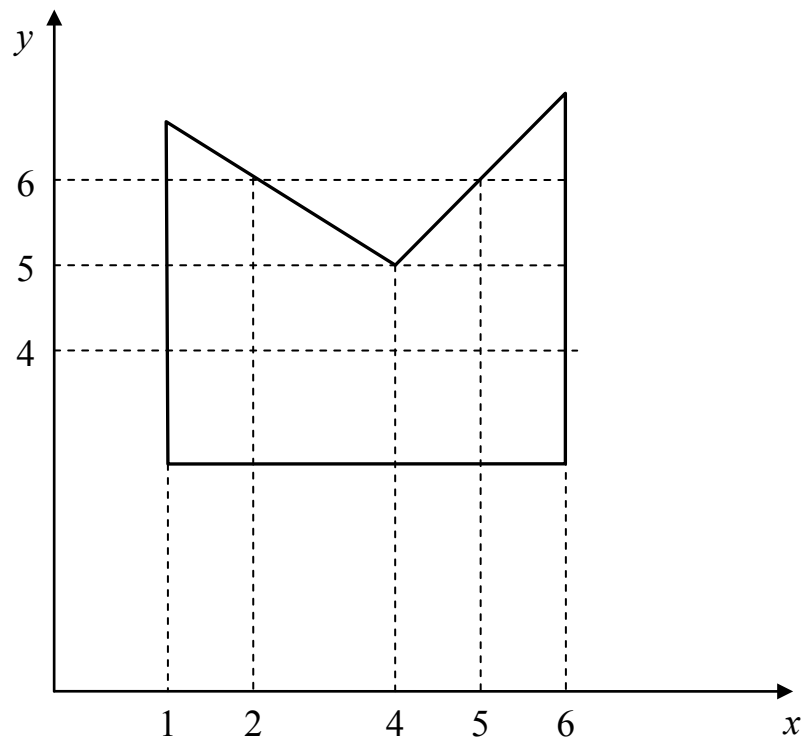


Рисунок 2.18 – Проходження сканувальних рядків

1. *Простий* випадок. Наприклад, на рис. 2.18 сканувальний рядок  $y = 4$  перетинає багатокутник при  $x = 1$  і  $x = 6$ . Виходить три ділянки  $x < 1$ ,  $1 \leq x \leq 6$ ,  $x > 6$ . Сканувальний рядок  $y = 6$  перетинає багатокутник при  $x = 1$ ,  $x = 2$ ,  $x = 5$ ,  $x = 6$ . Виходить п'ять ділянок  $x < 1$ ,  $1 \leq x \leq 2$ ,  $2 < x < 5$ ,  $5 \leq x \leq 6$ ,  $x > 6$ . У цьому випадку  $x$  сортується у порядку зростання. Далі перелік  $x$  розглядається попарно. Між парами точок перетину заповнюються всі пікселі. Для  $y = 4$  заповнюються пікселі в інтервалі  $(1, 6)$ , для  $y = 6$  заповнюються пікселі в інтервалах  $(1, 2)$  і  $(5, 6)$ .

2. Сканувальний рядок проходить через вершину (рис. 2.19).

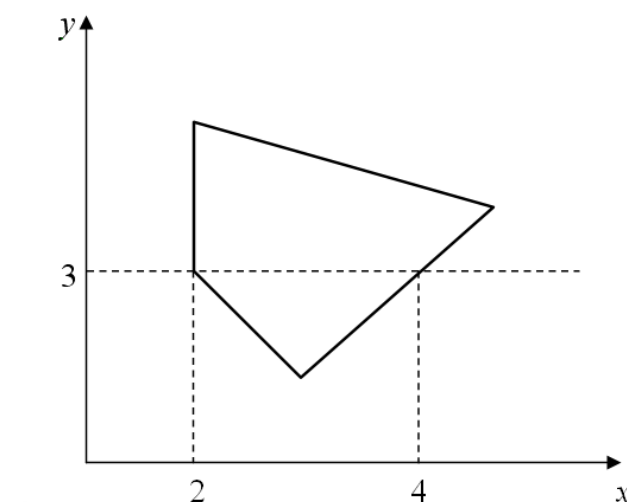


Рисунок 2.19 – Проходження сканувального рядка

Наприклад, по сканувальному рядку  $y = 3$  впорядкований список  $x$  вийде як (2, 2, 4). Вершина багатокутника була врахована двічі, і тому інтервал, що зафарбовується, виходить хибним: (2, 2). Отже, при перетині вершини сканувальним рядком вона має враховуватися один раз. І список з  $x$  у наведеному прикладі буде (2, 4).

3. Сканувальний рядок проходить через локальний мінімум або максимум (див. рис. 2.18 при  $y = 5$ ).

У цьому випадку враховуються всі перетини вершин сканувальним рядком. На рис. 2.18 за  $y = 5$  формується список  $x$  (1, 4, 4, 6). Інтервали, що заповнюються: (1, 4) і (4, 6). Умова знаходження локального мінімуму чи максимуму визначається під час розгляду кінцевих вершин для ребер, з'єднаних у вершині. Якщо в обох кінців координати  $y$  більші, ніж у вершини перетину, то вершина є локальним мінімумом. Якщо менші, то вершина перетинає локальний максимум.

Для прискорення роботи алгоритму використовується перелік активних ребер (ПАР). Цей список містить ті ребра багатокутника, які перетинають сканувальний рядок. При перетині чергового сканувального рядка вершини багатокутника з ПАР видаляються ребра, які знаходяться вище, і додаються кінці, які перетинає сканувальний рядок. При роботі алгоритму знаходять перетини сканувального рядка тільки з ребрами з ПАР.

**20. Задано відрізок, координати кінців якого  $P_3(-\frac{5}{2}, -1)$ ,  $P_4(\frac{3}{2}, 2)$ .**

**«Відсікаюче» вікно –**

$x_{min}$ ,  $x_{max}$ ,  $y_{min}$ ,  $y_{max}$ ,

**координати якого дорівнюють (-1, 1, -1, 1) (рис. 2.20).**

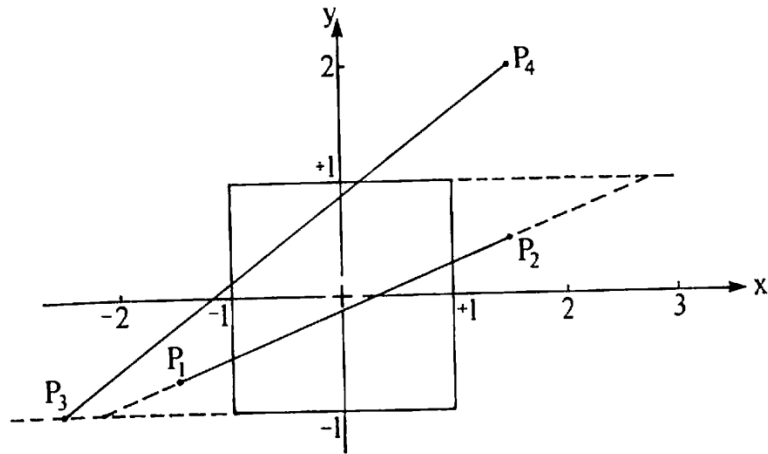


Рисунок 2.20 – Відсікання частково видимих відрізків

Знайдемо видиму частину відрізка.

Точки перетину задаються параметрами

$$t_L = \frac{3}{8}, \quad t_{\Pi} = \frac{7}{8}, \quad t_H = 0, \quad t_B = \frac{2}{3}.$$

Видно, що всі чотири значення задовольняють умову  $0 \leq t \leq 1$ . Але точок перетину тільки дві для опуклого багатокутника. Далі треба відібрати два значення параметра із чотирьох. У даному випадку треба підставляти значення параметра і звіряти з рисунком. Отримаємо, що шуканими значеннями параметра є

$$t_L = \frac{3}{8} \text{ і } t_B = \frac{2}{3},$$

які визначають координати точок перетину  $(-1, 1/8)$  і  $(1/6, 1)$ .

**21. Нехай задано повністю видимий відрізок із кінцевими точками  $P_1(-\frac{2}{2}, \frac{1}{2}), P_2(\frac{1}{2}, -\frac{1}{2})$ , в тому ж вікні  $(-1, 1, -1, 1)$  (рис. 2.21). З'ясувати, видимий даний відрізок чи невидимий**

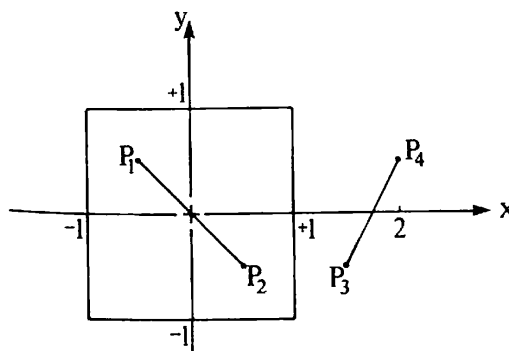


Рисунок 2.21 – Повністю видимі і повністю невидимі відрізки

Розрахунки, аналогічні попередньому прикладу, дають значення для параметрів, які відповідають перетинам відрізка зі сторонами вікна.

$$t_L = -\frac{1}{2}, \quad t_{\Pi} = \frac{3}{2}, \quad t_H = \frac{3}{2}, \quad t_B = -\frac{1}{2}$$

Усі значення параметра знаходяться поза інтервалом  $0 \leq t \leq 1$ .

Це означає, що відрізок не перетинає межі вікна, але він повністю видимий, чи повністю невидимий. Наприклад, для повністю невидимого відрізка  $P_3(\frac{3}{2}, -\frac{1}{2}), P_4(2, \frac{1}{2})$  на рис. 2.21 розраховані значення параметрів, які відповідають перетинам відрізка зі сторонами вікна, будуть

$$t_L = -5, \quad t_{\Pi} = -1, \quad t_H = -\frac{31}{2}, \quad t_B = \frac{3}{2}$$

Знову, як і в попередньому випадку, всі значення параметра знаходяться поза інтервалом  $0 \leq t \leq 1$ . Проте відрізок повністю невидимий. Тобто встановити, чи відрізок повністю видимий чи повністю невидимий, цим методом неможливо.

**22. Навести приклад, що ілюструє роботу алгоритму, що використовує z-буфер**

Розглянемо прямокутник, координати кутових точок якого рівні  $P_1(10, 5, 10), P_2(10, 25, 10), P_3(25, 25, 10), P_4(25, 5, 10)$  і трикутник з вершинами  $P_5(15, 15, 15), P_6(25, 25, 5), P_7(30, 10, 5)$ , що його «протикає» (рис. 2.22, а)

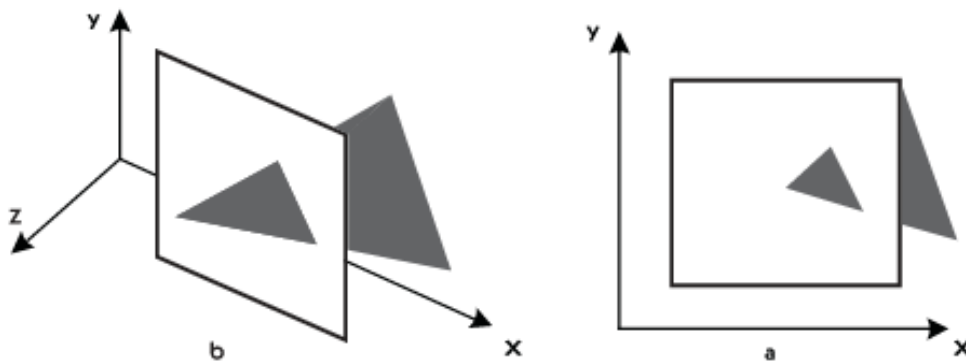


Рисунок 2.22 – «Протикаючий» трикутник



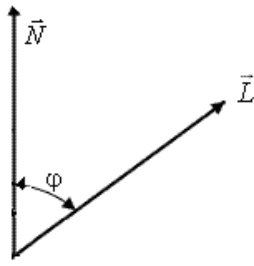


Рисунок 2.24 – Растрове розгорнення прямокутника

$$I_d = I_0 \cdot k_d \cdot \cos \varphi = 100 \cdot 0,5 \cdot 0,5 = 25$$

24. Знайти значення спекулярної складової світла в заданій точці за умови, що інтенсивність джерела світла дорівнює 100, а кут між нормаллю до поверхні і напрямком на джерело світла доповнює  $30^\circ$ , кут між нормаллю до поверхні і напрямком на спостерігача доповнює  $60^\circ$ , коефіцієнт відбиття  $k_s$  дорівнює 0,5; коефіцієнт спекулярності поверхні дорівнює 2

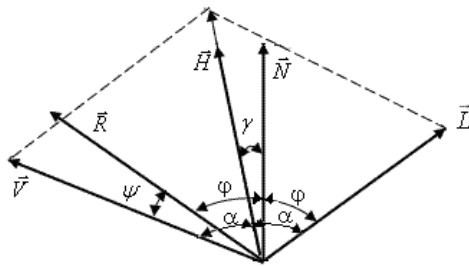


Рисунок 2.25 – Вектори до точки поверхні

За наявності в сцені точкового джерела світла інтенсивність спекулярного відбиття пропорційна (косинусу кута)<sup>n</sup> між векторами  $\vec{V}$  і  $\vec{R}$ . У цьому випадку для обчислення інтенсивності спекулярної складової використовують формулу

$$I = I_0 \cdot k_s \cdot (\vec{R} \cdot \vec{V})^n$$

де  $I_0$  – інтенсивність джерела світла,

$k_s \in [0, 1]$  – коефіцієнт спекулярного відбиття,

$n$  – коефіцієнт спекулярності поверхні.

Вектор  $\vec{R}$  визначає напрямлення до спостерігача. Вектор  $\vec{R}$  називають вектором дзеркального відбиття.

На рис. 2.25 відображено необхідні побудови для знаходження косинусу кута між векторами  $\vec{V}$  і  $\vec{R}$ .

$$I = I_0 \cdot k_s \cdot (\vec{R} \cdot \vec{V})^n = 100 \cdot 0,5 \cdot \cos^2 30^\circ = 100 \cdot 0,5 \cdot 0,75 = 37,5.$$

Потрібно зазначити, що всі вектори мають бути нормалізовані.

### 3 КОДУВАННЯ ГРАФІЧНОЇ ІНФОРМАЦІЇ

*1. Визначити необхідний обсяг відеопам'яті для різних графічних режимів екрана монітора, якщо відома глибина кольору на одну точку*

Режим екрана	Глибина кольору (бітів на точку)				
	4	8	16	24	32
640 на 480					
800 на 600					
1024 на 768					
1280 на 1024					

Для зразка розглянемо екран 640 \* 480. Усього на екрані 640 \* 480 = 307200 точок.

Необхідний обсяг відеопам'яті

$$V = 4 \text{ біти} * 307200 = 1228800 \text{ бітів} = 153600 \text{ байтів} = 150 \text{ Кбайтів.}$$

Аналогічно розраховується необхідний обсяг відеопам'яті інших графічних режимів.

Режим екрана	Глибина кольору (бітів на крапку)				
	4	8	16	24	32
640 на 480	150 Кб	300 Кб	600 Кб	900 Кб	1,2 Мб
800 на 600	234 Кб	469 Кб	938 Кб	1,4 Мб	1,8 Мб
1024 на 768	384 Кб	768 Кб	1,5 Мб	2,25 Мб	3 Мб
1280 на 1024	640 Кб	1,25 Мб	2,5 Мб	3,75 Мб	5 Мб

*2. Для зберігання растрового зображення розміром 128 × 128 пікселів відвели 4 Кб пам'яті. Якою є максимально можлива кількість кольорів на панелі зображення?*

Визначимо кількість точок зображення. 128 \* 128 = 16384 пікселів  
Обсяг пам'яті зображення 4 Кб виразимо в бітах.

$$4 \text{ Кб} = 4 * 1024 = 4096 \text{ байтів} = 4096 * 8 \text{ бітів} = 32768 \text{ бітів}$$

Знайдемо глибину кольору 32768 : 16384 = 2

$N = 2^I$ , де N – число кольорів на панелі. N = 4.



**3. Який обсяг відеопам'яті необхідний для зберігання чотирьох сторінок зображення, якщо бітова глибина дорівнює 24, а роздільна здатність дисплея –  $800 \times 600$  пікселів?**

Знайдемо обсяг відеопам'яті для однієї сторінки

$$800 * 600 * 24 = 11520000 \text{ бітів} = 1440000 \text{ байтів} = 1406,25 \text{ Кб} \approx 1,37 \text{ Мб}$$

$$1,37 * 4 = 5,48 \text{ Мб} \approx 5.5 \text{ Мб} \text{ для зберігання 4 сторінок.}$$

**4. Визначити обсяг відеопам'яті монітора, який необхідний для реалізації графічного режиму монітора High Color з роздільною здатністю  $1024 \times 768$  пікселів і палітрою кольорів з 65536 кольорів**

Режим High Color – це 16 біт на точку. Об'єм пам'яті можна знайти, визначивши кількість точок на екрані і помноживши на глибину кольору, тобто, 16.

За формулою  $K = 2^I$ , де  $K$  – кількість кольорів,  $I$  – глибина кольору, визначимо глибину кольору.  $2^I = 65536$ .

$$\text{Глибина кольору становить: } I = \log_2 65536 = 16 \text{ біт}$$

$$\text{Кількість точок зображення дорівнює: } 1024 \times 768 = 786432.$$

Обсяг відеопам'яті дорівнює

$$16 \text{ бітів} \times 786432 = 12582912 \text{ бітів} = 1572864 \text{ байтів} = 1536 \text{ Кб} = 1,5 \text{ Мб}$$

**5. У процесі перетворення растрового графічного зображення кількість кольорів зменшилася з 65536 до 16. У скільки разів зменшиться обсяг пам'яті, яку він займає?**

Щоб закодувати 65536 різних кольорів кожної точки потрібно 16 бітів. Щоб закодувати 16 кольорів потрібно 4 біти. Отже, обсяг пам'яті зменшився у  $16 : 4 = 4$  рази.

**6. Чи достатньо відеопам'яті обсягом 256 Кбайтів для роботи монітора в режимі  $640 \times 480$  з палітрою з 16 кольорів?**

Визначемо обсяг відеопам'яті, яка необхідна для роботи монітора в режимі  $640 \times 480$  і палітрою в 16 кольорів.

$$V = I * X * Y = 640 * 480 * 4 = 1228800 \text{ бітів} = 153600 \text{ байтів} = 150 \text{ Кб.}$$

$150 < 256$ , отже пам'яті достатньо.

**7. Вкажіть мінімальний обсяг пам'яті (у кілобайтах), достатній для зберігання будь-якого растрового зображення розміром  $256 \times 256$  пікселів, якщо відомо, що для зображення використовується палітра з 216 кольорів. Саму палітру не потрібно зберігати**

Знайдемо мінімальний обсяг пам'яті, необхідний для зберігання одного піксела. У палітрі зображення використовується  $2^{16}$  кольорів, отже, один піксел може бути зіставлений з будь-яким із  $2^{16}$  можливих номерів палітри кольорів. Тому мінімальний обсяг пам'яті, для одного піксела дорівнюватиме  $\log_2 2^{16} = 16$  бітів. Мінімальний обсяг пам'яті, достатній для зберігання всього зображення дорівнюватиме

$$16 * 256 * 256 = 2^4 * 2^8 * 2^8 = 2^{20} : 2^3 = 2^{17} \text{ байтів} = 2^{17} : 2^{10} = 2^7 \text{ Кбайтів} = 128 \text{ Кбайтів},$$

що відповідає пункту номер 1.

**8. Скільки секунд знадобиться модему, що надсилає повідомлення зі швидкістю 28800 бітів/с, щоб передати кольорове растрове зображення розміром  $640 \times 480$  пікселів, за умови, що колір кожного піксела кодується трьома байтами?**

Визначимо обсяг зображення у бітах

$$3 \text{ байти} = 3 * 8 = 24 \text{ біти},$$

$$V = I * X * Y = 640 * 480 * 24 \text{ біти} = 7372800 \text{ бітів}$$

Знайдемо час для передачі зображення:  $7372800 : 28800 = 256$  секунд

**9. Скільки секунд знадобиться модему, що передає повідомлення зі швидкістю 14400 бітів/с, щоб передати колірний растр розміром  $800 \times 600$  пікселів, за умови, що в палітрі 16 мільйонів кольорів?**

Для кодування 16 млн кольорів потрібно 3 байти або 24 біти (графічний режим True Color). Загальна кількість пікселів у зображенні  $800 \times 600 = 480000$ .

Оскільки 1 піксел має 3 байти, то  $480000 \text{ пікселів} * 3 = 1\,440\,000$  байтів або  $11\,520\,000$  бітів.  $11\,520\,000 : 14400 = 800$  секунд.

**10. Сучасний монітор дозволяє формувати на екрані 16777216 різних кольорів. Скільки бітів пам'яті займає 1 піксел?**

Один піксел кодується комбінацією двох знаків «0» та «1». Потрібно дізнатися про довжину коду піксела

$$2^x = 16777216, \log_2 16777216 = 24 \text{ біти.}$$

**11. Монітор працює з 16-кольоровою палітрою в режимі 640 \* 400 пікселів. Для кодування зображення потрібно 1250 Кбайтів. Скільки сторінок відеопам'яті він займає?**

Сторінка – ділянка відеопам'яті, що містить інформацію про один образ екрана. У відеопам'яті можуть розміщуватися одночасно кілька сторінок, тому, щоб дізнатися кількість сторінок, потрібно поділити обсяг відеопам'яті для всього зображення на обсяг пам'яті на одну сторінку. К – число сторінок

$$K = V_{\text{зобр}} / V_{1\text{c}}$$

$$V_{\text{зобр}} = 1250 \text{ Кб за умовою}$$

Для цього обчислимо обсяг відеопам'яті для однієї сторінки зображення з 16-колірною палітрою і роздільною здатністю 640 \* 400.

$$V_{1\text{c}} = 640 * 400 * 4, \text{ де } 4 - \text{глибина кольору } (2^4 = 16)$$

$$V_{1\text{c}} = 1024000 \text{ бітів} = 128000 \text{ байтів} = 125 \text{ Кб}$$

$$K = 1250 : 125 = 10 \text{ сторінок}$$

**12. Сторінка відеопам'яті складає 16000 байтів. Монітор має роздільну здатність 320 \* 400 пікселів. Скільки кольорів у палітрі?**

*Розв'язання*

$V = I * X * Y$  – обсяг однієї сторінки,  $V = 16000$  байтів = 128000 бітів за умовою. Знайдемо глибину кольору  $I$ .

$$I = V / (X * Y).$$

$$I = 128000 / (320 * 400) = 1.$$

Визначимо тепер скільки кольорів на панелі.  $K = 2^I$ , де  $K$  – кількість кольорів,  $I$  – глибина кольору.  $K = 2$

**13. Сканується кольорове зображення розміром 10 x10 см. Роздільна здатність сканера 600 dpi та глибина кольору 32 біти. Який обсяг матиме отриманий графічний файл?**

Роздільна здатність сканера 600 dpi (dot per inch – точок на дюйм) означає, що на відрізку довжиною 1 дюйм сканер здатний розрізнити 600 точок. Перекладемо роздільну здатність сканера з точок на дюйм у точки на сантиметр

$$600 \text{ dpi} : 2,54 \approx 236 \text{ пікселів/см } (1 \text{ дюйм} = 2.54 \text{ см})$$

Отже, розмір зображення у точках становитиме  $2360 \times 2360$  пікселів (помножили на 10 см).

Загальна кількість точок зображення

$$2360 \times 2360 = 5569600.$$

Обсяг файлу

$$32 \text{ біти} \times 5569600 = 178227200 \text{ бітів} \approx 21 \text{ Мбайт}$$

**14. Обсяг відеопам'яті дорівнює 256 Кб. Кількість використовуваних кольорів – 16. Обчисліть варіанти роздільної здатності монітора за умови, що кількість сторінок зображення може бути 1, 2 або 4**

Якщо число сторінок дорівнює 1, то формулу  $V = I * X * Y$  можна виразити як

$$256 * 1024 * 8 \text{ бітів} = X * Y * 4 \text{ біти},$$

(оскільки використовується 16 кольорів, то глибина кольору дорівнює 4 біти.)  
Тобто,  $512 * 1024 = X * Y$ ;  $524288 = X * Y$ .

Співвідношення між висотою та шириною екрана для стандартних режимів не різняться між собою та дорівнюють 0,75. Отже, щоб знайти  $X$  і  $Y$ , треба розв'язати систему рівнянь

$$\begin{cases} XY = 524288 \\ X / Y = 0,75 \end{cases}$$

Виразимо  $X = 524288 / Y$ , підставимо в друге рівняння, отримаємо

$$Y^2 = 524288 * 3 / 4 = 393216.$$

Знайдемо  $Y \approx 630$ ;  $X = 524288 / 630 \approx 830$ .

Варіантом роздільної здатності може бути  $630 \times 830$ .

Якщо кількість сторінок дорівнює 2, то одна сторінка обсягом  $256 : 2 = 128$  Кбайтів, тобто,  $128 * 1024 * 8 \text{ бітів} = X * Y * 4 \text{ бітів}$ , тобто,  $256 * 1024 = X * Y$ ;  $262144 = X * Y$ .

Розв'язуємо систему рівнянь  $\begin{cases} XY = 262144 \\ Y / X = 0,75 \end{cases}$

$$X = 262144 / Y; Y^2 = 262144 * 3 / 4 = 196608; Y = 440, X = 600.$$

Варіантом роздільної здатності може бути  $600 \times 440$ .

Якщо кількість сторінок дорівнює 4, то

$$256 : 4 = 64; 64 * 1024 * 2 = X * Y; 131072 = X * Y;$$

Розв'язуємо систему рівнянь  $\begin{cases} XY = 131072 \\ Y / X = 0,75 \end{cases}$

$$X = 131072 / Y; Y^2 = 131072 * 3 / 4 = 98304; Y \approx 310, X \approx 420.$$

**15. Обсяг сторінки відеопам'яті – 125 Кбайтів. Монітор працює з 16-кольоровою палітрою. Яка роздільна здатність екрана?**

Оскільки глибина кольору дорівнює 4 ( $2^4 = 16$ ), маємо  $V = 4 * X * Y$ . У формулі обсягу відеопам'яті обсяг виражений в бітах, а в умові завдання дано в Кбайтах, тому обидві частини рівності треба подати в байтах.

$125 * 1024 = (X * Y * 4) / 8$  (ділимо праворуч на 8 – переводимо в байти, множимо зліва на 1024 – переводимо в байти. Далі розв'язуємо рівняння  $4 * X * Y = 125 * 1024 * 8$ .

$$X * Y = 125 * 1024 * 2 = 250 * 1024 = 256000$$

У парі роздільної здатності екрана зустрічається число 640, Розділимо отримане число на 640

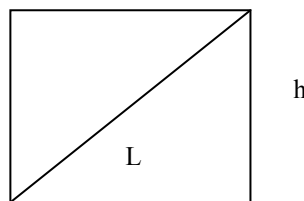
$$256000 : 640 = 400.$$

**16. Які графічні режими роботи монітора може забезпечити відеопам'ять обсягом 1 Мб?**

Відеопам'ять обсягом 1 Мб може забезпечити такі графічні режими:

- $640 \times 480$  (при глибині кольору 4, 8, 16, 24 біти);
- $800 \times 600$  (при глибині кольору 4, 8, 16 бітів);
- $1024 \times 768$  (при глибині кольору 4, 8 бітів);
- $1280 \times 1024$  (при глибині кольору 4 біти).

**17. Визначити максимально можливу роздільну здатність екрана для монітора з діагоналлю 15" і розміром точки екрана 0,28 мм**



Завдання зводиться до знаходження числа точок шириною екрана. Виразимо розмір діагоналі в сантиметрах. Враховуючи, що 1 дюйм = 2,54 см, маємо:  $2,54 \text{ см} \times 15 = 38,1 \text{ см}$ .

Визначимо співвідношення між висотою та шириною екрана для режиму екрана  $1024 \times 768$  пікселів:  $768 : 1024 = 0,75$ .

Визначимо ширину екрана. Нехай ширина екрана дорівнює  $L$ , а висота –  $h$ ,

$$h : L = 0,75, \text{ тоді } h = 0,75L.$$

За теоремою Піфагора маємо

$$L^2 + (0,75 L)^2 = 38,1^2 \quad L^2 + 0,5625 L^2 = 1451,61 \quad L \approx 30,5 \text{ см}$$

Кількість точок по ширині екрана дорівнює  $305 \text{ мм} : 0,28 \text{ мм} = 1089$ . Отже, максимально можливою роздільною здатністю екрана монітора є  $1024 \times 768$ .

**18. Скільки кольорів буде використано, якщо для кожного кольору піксела взяти 2 рівні градації яскравості, 64 рівні яскравості кожного кольору?**

Усього для кожного піксела використовується набір із трьох кольорів (червоний, зелений, синій) зі своїми рівнями яскравості (0 – горить, 1 – не горить). Отже,

$$K = 2^3 = 8 \text{ кольорів,}$$

$$K = 64^3 = 262144 \text{ кольорів.}$$

**19. Заповніть в таблиці кольорів на 24-бітвовій глибині кольору в шістнадцятковому форматі.**

При глибині кольору в 24 біти на кожен із кольорів виділяється по 8 бітів, тобто, для кожного з кольорів можливі 256 рівнів інтенсивності ( $2^{28} = 256$ ). Ці рівні задані двійковими кодами (мінімальна інтенсивність – 00000000, максимальна інтенсивність – 11111111). У двійковому поданні виходить таке формування кольорів:

Назва кольору	Інтенсивність		
	червоний	Зелений	Синій
чорний	00000000	00000000	00000000
червоний	11111111	00000000	00000000
зелений	00000000	11111111	00000000
синій	00000000	00000000	11111111
білий	11111111	11111111	11111111

Перевівши в шістнадцяткову систему числення маємо:

Назва кольору	Інтенсивність		
	червоний	зелений	синій
чорний	00	00	00
червоний	FF	00	00
зелений	00	FF	00
синій	00	00	FF
білий	FF	FF	FF

**20. Автоматична камера формує зображення розміром  $200 \times 256$  пікселів. Для кодування кольору кожного пікселя використовується однакова кількість бітів, коди пікселів записуються у файл один за одним без проміжків. Обсяг файлу з зображенням не може перевищувати 65 Кбайтів без урахування розміру заголовка файлу. Яку максимальну кількість кольорів можна використовувати на панелі?**

Першочергове завдання – визначити глибину кольору даного растрового зображення, а потім скористатися формулою Хартлі для отримання остаточної відповіді.

Із постановки завдання відомі:

- габарити растрового зображення, виражені в пікселях ( $200 \times 256$ );
- загальна інформаційна вага зображення, виражена в Кб (65 Кбайтів).

Необхідно звернути увагу на такий фрагмент тексту: «Не може перевищувати 65 Кбайтів». В умові не йдеться про те, що розмір файлу строго дорівнює 65 Кбайтів, а лише про те, що він не перевищує цієї величини.

Потрібно прагнути до того, щоб кожен піксел кодувався якомога більшим цілим числом бітів, отже, розмір вихідного графічного файлу в наступних розрахунках будемо приймати найбільшим, тобто 65 Кбайтів.

Пам'ять на піксел = загальний обсяг пам'яті / кількість пікселів.

Для спрощення наступних математичних викладок зробимо таке:

1. Розкладемо всі задані натуральні числа на прості множники (факторизація числа).
2. Перекладемо одиниці виміру інформації з [Кбайтів] у [біти].

$$200 = 2 \times 2 \times 2 \times 5 \times 5 = 2^3 \times 5^2$$

$$256 = 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 \times 2 = 2^8$$

В 1-ому Кбайті 8192 або  $2^{13}$  бітів.

Підставляємо розкладені величини у формулу

$$I = \frac{5 \cdot 13 \cdot 2^{13} [\text{бит}]}{2^3 \cdot 5^2 \cdot 2^8} = \frac{5 \cdot 13 \cdot 2^{13} [\text{бит}]}{2^{11} \cdot 5^2} = \frac{4 \cdot 13}{5} = \frac{52}{5} = 10.4, [\text{бит}]$$

Але, глибина кольору має виражатися цілим числом бітів, тому округляємо «вниз» до найближчого цілого:  $10.4 \approx 10$  [бітів].

# ТЕСТ

1. Галузь діяльності, в якій комп'ютери використовуються як інструмент для синтезу (створення) зображень, так і для обробки візуальної інформації, отриманої з реального світу

**Комп'ютерна графіка**

Розпізнавання зображення

Обробка зображення

Векторна графіка

2. Графіка, яка оперує з двовимірним масивом (матрицею) пікселів.

Інтерактивна графіка

**Растрова графіка**

Фрактальна графіка

Векторна графіка

3. Об'єкт, окремі елементи якого успадковують властивості батьківських структур.

Примітив

**Фрактал**

Сплайн

Вікно

4. Розділ комп'ютерної графіки, коли користувач має можливість динамічно керувати вмістом зображення, його формою, розміром і кольором на поверхні екрана за допомогою інтерактивних пристроїв управління.

Пасивна графіка.

Растрова графіка

**Інтерактивна графіка**

Фрактальна графіка

5. Автор першої програми з формування векторних графічних зображень.

**Сазерленд**

Глушков

Брезенхем

Олександров



6. Найменший логічний елемент двовимірного цифрового зображення в растровій графіці
- точка
  - зерно люмінофора
  - піксел (правильний)
  - растр

7. Різновид користувальницького інтерфейсу, у якому елементи інтерфейсу (меню, кнопки, значки, списки і т. п.), показані користувачеві на екрані, виконані у вигляді графічних зображень.

Меню

**Графічний інтерфейс користувача**

Список

Інтерактивний екран

8. Елемент об'ємного зображення, що містить значення елемента растра в тривимірному просторі.

**Воксель**

Тексель

Піксел

Піраміда

9. Растрове зображення, що накладається на поверхню полігону, з яких складаються 3D-моделі, для додання їй кольору, забарвлення або ілюзії рельєфу

Растр

**Текстура**

Фактура

Воксель

10. Послідовність паралельних один одному відеорядків на екрані електронно-променевої трубки або іншого устаткування, що виникає в результаті роботи системи розгортки.

**Растр**

Вікно

Інтерфейс

Текстура

11. Розподіл екрану 1024\*768

**SVGA**

VGA

MVGA

SXGA

12. Найменші, неподільні з точки зору прикладних програм, графічні елементи, які використовуються як базові для побудови більш складних зображень.

***Графічні примітиви***

Графічні команди

Текстури

Об'єкти

13. Чи може множина пов'язаних між собою графічних елементів, які визначають об'єкт для ідентифікації, бути примітивом ?

**Може**

Не може

Може, за умови, що реалізується відеокартою

Може, якщо складові графічні елементи неподільні

14. Примітив, для генерації якого в графічному пристрої є відповідний апаратний блок, називається

Логічним

**Фізичним**

Структурним

Текстурним

15. Видимість, колір, яскравість, динамічні характеристики, тип і товщина ліній визначають для примітива

**Візуальні властивості**

Статус

Режим занесення у відеопам'ять

Ядро

16. Множина примітивів, які мають однакові візуальні властивості і статус, і ідентифікуються одним ім'ям, називається:
- Графічним елементом
  - Графічним об'єктом**
  - Графічним сегментом
  - Графічним курсором
17. Максимальна абсолютна похибка, як має місце при формуванні відрізків прямих за алгоритмом Петуха-Обідника, дорівнює
- Кроку дискретизації
  - Півкроку дискретизації**
  - Нулю
  - Півтора кроки дискретизації
18. У методі оцінювальної функції при формуванні крокової траєкторії оцінюється:
- Знак оцінювальної функції**
  - Значення оцінювальної функції
  - Модуль оцінювальної функції
  - Вид оцінювальної функції
19. В алгоритмі Петуха-Обідника і Брезенхема початкові значення оцінювальної функції :
- Збігаються
  - Мають різні значення
  - Не збігаються**
  - Завжди від'ємні
20. При реалізації колового інтерполювання крокова траєкторія завжди проходить через кінцеву точку дуги
- Не проходить
  - Проходить
  - Проходить за умови використання режиму «доводки»**
  - Приходить, якщо оцінювальна функція від'ємна

21. При реалізації кругової інтерполяції і використанні чотирикрової орієнтації руху для формуванні  $\frac{1}{4}$  кола в першому квадранті необхідно виконати:

**2R кроків**

R кроків

R/2 кроків

R+ 2 кроків

22. При використанні аналітичної форми завдання кривої можна сформулювати:

Замкнуті криві

**Незамкнуті криві**

Криві спірального типу

Кола великого радіуса

23. Застосування векторної графіки порівняно з растровою ...

- не змінює способи кодування зображення
- збільшує обсяг пам'яті, необхідної для зберігання зображення
- не впливає на обсяг пам'яті, необхідної для зберігання зображення, і на трудомісткість редагування зображення
- **скорочує обсяг пам'яті, необхідної для зберігання зображення, і полегшує редагування останнього (правильний)**

24. Графіка з поданням зображення у вигляді послідовності точок зі своїми координатами, з'єднаними між собою кривими, які описуються математичними рівняннями, називається

- фрактальною
- растровою
- **векторною (правильно)**
- прямолінійною

25. Для формування відрізка прямої за алгоритмом Петуха-Обідника необхідно виконати:

**БП інтерполяційних тактів;**

МП інтерполяційних тактів

БП-МП інтерполяційних тактів

БП+МП інтерполяційних тактів

26. Ефект Рунге-Мерей виникає за умови, що кількість вузлових точок більша:

5

2

10

4

27. Степінь полінома Лагранжа дорівнює:

Кількості вузлових точок

3

5

**На одиницю менше кількості вузлових точок**

28. Кількість тактів в алгоритмі Петуха-Обідника дорівнює:

меншому приросту

**більшому приросту**

різниці більшого та меншого приростів

кількості діагональних кроків

29. При збільшенні кількості вузлових точок при інтерполюванні за методом Лагранжа:

Форма кривої не змінюється

Форма кривої змінюється тільки у вузлових точках

**Форма кривої змінюється за межами вузлових точок**

Крива набуває синусоїдальної форми

30. При сплайновій інтерполяції задаються:

Похідна в точках «склеювання»;

Координати точок «склеювання»

**Похідна та координати в усіх базових точках;**

Рівняння кривої

31. Мінімальний степінь сплайна:

3

2

5

1

32. Сплайн є:

**Кусково-поліноміальною функцією**

Неперервною спадною функцією

Синусоїдальною функцією

Зростаючою функцією

33. При сплайнової інтерполяції на кожному підінтервалі необхідно визначити таку кількість невідомих:

1

2

3

4

34. У методі відсікання шляхом ділення відрізка навпіл використовують операцію

Множення

Ділення

**Зсуву**

Піднесення до степеня

35. Метод Коена-Сазерленда дозволяє зменшити

**Кількість «довгих» операцій**

Кількість операцій додавання

Кількість операцій піднесення до степеня

Кількість матричних операцій

36. Кількість ітерацій в методі відсікання шляхом ділення відрізка навпіл дорівнює

2

$\log_2 N$ , де  $N$  – довжина відрізка

$2^N$

1

37. Однією з основних функцій графічного редактора є:

Генерація та зберігання коду зображення

**Створення зображень**

Перегляд та виведення вмісту відеопам'яті  
Сканування зображень

38. Якість зображення визначається кількістю точок, з яких воно складається, і це називається:
- Колірна здатність
  - Графічна розгортка
  - Роздільна розгортка
  - Роздільна здатність**
39. Кольорове зображення на екрані формується за рахунок змішування таких базових кольорів:
- Синій, жовтий, червоний
  - Червоний, зелений, синій**
  - Жовтий, червоний, чорний
  - Білий, зелений, червоний
40. Відеоадаптер – це:
- Пристрій, що управляє роботою графічного дисплея**
  - Електронний, енергозалежний пристрій для зберігання інформації про графічне зображення
  - Програма, що розподіляє ресурси відеопам'яті
  - Дисплейний процесор
41. Графічна інформація на екрані монітора подається у вигляді:
- Світлового зображення
  - Растрового зображення**
  - Кольорового зображення
  - Векторного зображення
42. Графічне зображення, подане у пам'яті комп'ютера як послідовність ліній, називається:
- Фрактальним
  - Векторним**
  - Лінійним
  - Растровим

43. Який з наведених графічних редакторів є векторним?

Adobe Photoshop

Paint

PhotoPaint

**Corel Draw**

44. У колірній моделі CMY описує реальні поліграфічні фарби за допомогою кольорів:

**Блакитний, пурпурний, жовтий**

Білий, жовтий, зелений

Червоний, синій, зелений

Чорний, червоний, зелений

45. Кольорові зображення формуються:

Глибиною кольору

Палітрою кольорів

**Двійковим кодом кольору**

Кількістю кольорів екрана

46. Діапазон кольорів, який може бути відтворений будь-яким способом, називається:

Насиченість

Перехід

**Колірне охоплення**

Яскравість

47. Застосування векторної графіки порівняно з растровою:

Не впливає на обсяг пам'яті, необхідний для зберігання зображення  
та на трудомісткість редагування зображення

Збільшує обсяг пам'яті, необхідної для зберігання зображення,  
та спрощує процес редагування зображення

**Зменшує обсяг пам'яті, необхідної для зберігання зображення,  
та полегшує редагування зображення**

Не змінює спосіб кодування зображення



48. Всі сучасні комп'ютерні відеодисплеї здатні відображати інформацію лише в:  
**растровому форматі**  
фрактальному форматі  
анімаційному форматі  
векторному форматі
49. Тривимірна графіка – розділ комп'ютерної графіки, сукупність прийомів та інструментів (як програмних, так і апаратних), призначених для:  
Зображення чорно-білих об'єктів  
**Зображення об'ємних об'єктів**  
Зображення плоских об'єктів  
Зображення кольорових об'єктів
50. Виберіть найпростіший графічний редактор:  
Paint NET  
Gimp  
**Paint**  
Inkscape
51. Яку дію можна виконати лише за допомогою растрового графічного редактора?  
Змінити масштаб зображення  
Змінити яскравість та контрастність зображення  
**Копіювати фрагмент зображення**  
Повернути зображення на задану кількість градусів
52. Графічне зображення, подане в пам'яті комп'ютера як опис сукупності точок із зазначенням їх координат і відтінку кольору, називається:  
Фрактальним  
Векторний  
Лінійним  
**Растровим**

53. Виберіть формат, який не є графічним:

BMP

GIF

**COM**

JPG

54. Які атрибути надаються об'єктам у растровій графіці?

Розмір створюваних об'єктів

Положення щодо координат

**Товщина ліній та колір заповнення**

Положення щодо краю листа

55. Пікселі на екрані утворюють сітку з горизонтальних та вертикальних стовпців, яку називають:

Координатна площина

Відеопам'ять

Растр

**Матриця**

56. Який вид графіки з'явився першим?

Ділова графіка

**Наукова графіка**

Анімаційна графіка

Ілюстративна графіка

57. Інструментами в графічному редакторі є ...

Лінія, коло, прямокутник

Виділення, копіювання, вставка

**Олівець, пензель, гумка**

Набори кольорів

58. Примітивами у графічному редакторі називаються ...

Набори кольорів

Олівець, пензель, гумка

**Лінія, коло, прямокутник**

Виділення, копіювання, вставка

59. Для отримання рухомого зображення використовується:

Ділова графіка

**Анімаційна графіка**

Наукова графіка

Ілюстративна графіка

60. Палітрою в графічному редакторі є ...

Олівець, пензель, гумка

Лінія, коло, прямокутник

**Набори кольорів**

Виділення, копіювання, вставка

61. Графічний редактор – прикладна програма, яка може бути використана для:

**Створення графічних зображень**

Написання музичного твору

Проведення обчислень

Написання твору

62. Виберіть рядок, у якому перелічені формати графічних файлів,

**\*.gif, \*.jpg, \*.png, \*.tif**

\*.txt, \*.doc, \*.rtf

\*.exe, \*.com

\*.wav, \*.mp3, \*.wma

63. Невеликий розмір файлів є перевагою:

Фрактальної графіки

Растрової графіки

Будь-якого виду графіки

**Векторної графіки**

64. Вкажіть формат файлу для редагування у Photoshop або ImageReady:

CDR

JPEG

BMP

**PSD**

65. Великий розмір файлів є недоліком:

Фрактальної графіки

Будь-якого виду графіки

Растрової графіки

**Векторної графіки**

66. Метафайловий формат для графічних файлів (векторних та растрових), що містять ілюстрації та текст з великим набором шрифтів та гіпертекстовими посиланнями з метою передачі їх по мережі в стислому вигляді – це:

BMP

CDR

PSD

**PDF**

67. Logic

BMP

CDR

PSD

**PDF**

## ЛІТЕРАТУРА

1. Романюк О. Н., Кательніков Д. І., Косовець О. П. Веб-дизайн і комп'ютерна графіка : навчальний посібник. Вінниця : ВНТУ, 2007. 103 с.
2. Романюк О. Н. Комп'ютерна графіка : навчальний посібник. Вінниця : ВДТУ, 2001. 129 с.
3. Роджерс Д., Адамс Дж. Математические основы машинной графики ; пер. с англ. М. : Мир, 2001. 604 с.
4. Маценко В. Г. Комп'ютерна графіка : навч. пос. Чернівці : Рута, 2009. 343 с.
5. Самодумкин С. А., Степанова М. Д., Колб Д. Г. Практикум по компьютерной графике : учебно-методическое пособие. В 3 ч. ; под ред. проф. В. В. Голенкова. Минск : БГУИР, 2007.
6. Авхадиев Ф. Г., Губайдуллина Р. К., Насибуллин Р. Г. Учебно-методическое пособие по численным методам анализа. Казань : Казанский (Приволжский) федеральный университет, 2019. 113 с.
7. Мосин В. Г. Математические основы компьютерной графики : монография. Самара : СГАСУ, 2005. 227 с.
8. Херн Д., Бейкер М. Компьютерная графика и стандарт OpenGL. М. : Издательский дом «Вильямс», 2005. 1168 с.
9. Хилл Ф. OpenGL. Программирование компьютерной графики. СПб. : Питер, 2002. 1088 с.
10. Цисарж В. В., Марусин Р. И. Математические методы компьютерной графики. К. : Факт, 2004. 464 с.
11. Василюк А. С., Мельникова Н. І. Комп'ютерна графіка : навчальний посібник. Львів : Видавництво Львівської політехніки, 2016. 308 с.
12. Шикин Е. В. Компьютерная графика. Динамика, реалистические изображения. М. : Диалог–МИФИ, 1997. 288 с.
13. Основы комп'ютерної графіки : навч. посібник ; за ред. В. С. Ходокова. К. : Центр навч. літератури, 2004. 392 с.
14. Горобець С. М. Основы комп'ютерної графіки ; за ред. М. В. Левківського. К. : Центр навч. літератури, 2006. 232 с.
15. Новожилова М. В., Мироненко В. В. Комп'ютерна графіка : навчально-методичний посібник. Х. : ХНУБА, 2015. 70 с.
16. Холодна З. Б., Яловкіна Є. Б. Комп'ютерна графіка та моделювання : навчальний посібник. Х : НТУ «ХП», 2007.
17. Коцюбинский А. О., Грошев С.В. Компьютерная графика : практическое пособие. М. : ТЕХНОЛОДЖИ-3000, 2011. 752 с.
18. Эйнджел Э. Интерактивная компьютерная графика. Вводный курс на базе OpenGL М. : Вильямс, 2001. 92 с.
19. Никулин Е. А. Компьютерная геометрия и алгоритмы машинной графики. СПб. : БХВ–Петербург, 2003. 560 с.
20. Залевський В. Й., Залевський С. В. Курс лекцій з диференціальної геометрії для студентів фізико-математичного факультету. Київ, 2013. 81 с.

*Електронне навчальне видання  
комбінованого використання  
Можна використовувати в локальному та мережному режимах*

**Романюк Олександр Никифорович  
Майданюк Володимир Павлович**

**ПРАКТИКУМ  
для самостійної роботи студентів  
з навчальної дисципліни  
«Комп'ютерна графіка»**

*Практикум*

Рукопис оформлено *О. Романюком*

Редактор *В. Дружиніна*

Оригінал-макет виготовлено *О. Ткачуком*

Підписано до видання 15.10.2022.

Гарнітура Times New Roman.

Зам. № P2022-082.

Видавець та виготовлювач –  
Вінницький національний технічний університет,  
Редакційно-видавничий відділ.

ВНТУ, ГНК, к. 114.

Хмельницьке шосе, 95, м. Вінниця, 21021.

Тел. (0432) 65-18-06.

**press.vntu.edu.ua;**

*Email: irvc.vntu@gmail.com.*

Свідоцтво суб'єкта видавничої справи  
серія ДК № 3516 від 01.07.2009 р.