

**ФАКУЛЬТЕТ АВТОМАТИКИ, ТЕЛЕМЕХАНІКИ ТА ЗВ'ЯЗКУ**  
**Кафедра „Обчислювальна техніка та системи управління”**

**ОСНОВИ ПРОЕКТУВАННЯ АЛГОРИТМІВ**  
**МЕТОДИЧНІ ВКАЗІВКИ**  
**до лабораторних робіт з дисципліни**  
***"КОМП'ЮТЕРНА ТЕХНІКА ТА ПРОГРАМУВАННЯ"***

**Харків 2011**

Методичні вказівки розглянуто та рекомендовано до друку на засіданні кафедри «Обчислювальна техніка та системи управління» 28 вересня 2009 р, протокол № 2.

Призначені для студентів факультету УПП та відповідають робочій програмі дисципліни «Комп'ютерна техніка та програмування».

Укладачі:

доценти С.Є. Бантюков,  
А.Ф. Карачаров,  
асистенти С.Є. Бантюкова,  
О.А.Кошева,  
старш. викл. О.Є. Пенкіна

Рецензент

проф. Г.І. Загарій

ОСНОВИ ПРОЕКТУВАННЯ АЛГОРИТМІВ

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт з дисципліни

*"КОМП'ЮТЕРНА ТЕХНІКА ТА ПРОГРАМУВАННЯ"*

Відповідальний за випуск Пенкіна О.Є.

Редактор Третьякова К.А.

---

Підписано до друку 19.10.09 р.

Формат паперу 60x84 1/16 . Папір писальний.

Умовн.-друк.арк. 1,25. Обл.-вид.арк. 1,5.

Замовлення № Тираж 200. Ціна

---

Видавництво УкрДАЗТу, свідоцтво ДК № 2874 від. 12.06.2007 р.

Друкарня УкрДАЗТу,  
61050, Харків - 50, майд. Фейербаха, 7

## РОБОТА 1

### ПРОЕКТУВАННЯ АЛГОРИТМІВ ЛІНІЙНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

**Мета роботи** – вивчення способів опису алгоритмів, методики проектування схем алгоритмів лінійних обчислювальних процесів.

#### **Завдання та порядок виконання**

1 Вивчити навчальний матеріал і підготувати відповіді на контрольні запитання.

2 Скласти схему алгоритму обчислення функцій відповідно до варіанта завдання.

#### **Контрольні запитання**

1 Назвати етапи розв'язання задач на ЕОМ.

2 Визначити поняття “алгоритм” і назвати його властивості.

3 Визначити поняття “програма” розв'язання задачі на ЕОМ.

4 Дати перелік способів опису алгоритмів.

5 Пояснити правила використання символів для схем алгоритмів.

6 Визначити поняття “лінійний обчислювальний процес”.

7 Дати перелік типів вказівок, що складають лінійний обчислювальний процес.

#### **Зміст звіту**

1 Номер роботи, її назва, визначення мети.

1 Стислі відповіді на контрольні запитання.

3 Алгоритми обчислення функцій та стислий їх опис.

4 Висновки з роботи.

## Навчальний матеріал

1 *Електронні обчислювальні машини* (ЕОМ) – це високопродуктивні засоби обробки інформації, призначені для розв'язання широкого кола різноманітних задач.

Розв'язання задачі на ЕОМ **включає** такі етапи:

- постановка задачі і вибір числового способу її розв'язання;
- розробка алгоритму (алгоритмізація);
- складання програми алгоритмічною мовою;
- реалізація програми на ЕОМ, включаючи дії з введення, відлагодження і виконання програми на ЕОМ;
- аналіз отриманих результатів.

ЕОМ є всього лише автоматом, що швидко і точно виконує приписи, складені людиною. Розробка таких приписів, тобто проектування ходу розв'язання задачі – невід'ємна частина діяльності, пов'язана із використанням ЕОМ. На початковому етапі ці приписи подаються у вигляді алгоритму.

З точки зору неформальних уявлень *алгоритм* – це система точних і зрозумілих приписів про склад і послідовність виконання кінцевого числа дій, необхідних для розв'язання задач даного типу.

Властивості алгоритму:

- детермінованість (визначеність) – однозначність результату процесу при заданих вхідних даних;
- дискретність – роз'єднання алгоритму на окремі елементарні дії, можливість виконання яких людиною або машиною не викликає сумнівів;
- масовість – можливість вибору вхідних даних з деякої безлічі даних;
- результативність – одержання результату або повідомлення про неможливість одержання результату при заданих вхідних даних.

За алгоритмом складається програма розв'язання задачі.

*Програма обчислювальної машини* – це опис алгоритму розв'язання задачі якою-небудь алгоритмічною

мовою, реалізуючи яку ЕОМ перетворює вхідні дані в результат розв'язання задачі.

## 2 Способи опису алгоритмів:

- словесний опис послідовності дій;
- операторна схема, в якій використовується рядковий символний запис з індексами для вказування заданої послідовності виконання дій;
- *схема алгоритму* - графічне зображення обчислювального процесу.

Елементами схем алгоритму є геометричні фігури (символи), кожна з яких визначає певну дію, та лінії потоку, що визначають послідовність виконання дій. Вид дії та дані (операнди), над якими виконуються дії, записуються всередині символів процесу традиційним способом (формули, відношення і т.ін.). Схеми алгоритмів є універсальним способом документування алгоритмів, бо їхній вигляд не залежить від того, якою мовою у подальшому вони будуть реалізовані. Ще одна перевага полягає у високій наочності схем. При побудові алгоритмів вибирається різноманітна глибина деталізації окремих операцій. На схемах алгоритмів стрілками позначаються тільки лінії потоку, що мають злам і напрямки, які вказують знизу вгору, праворуч - ліворуч. Розмір ( $a$ ) фігури алгоритму вибирається з ряду 10, 15, 20 мм. Ці розміри можна збільшити на число, кратне 5. Розмір  $b=1.5*a$ .

Для зручності читання схем алгоритмів символам процесу привласнюють порядкові номери. Номери символів процесу проставляються так, щоб їх можна було читати зліва направо і згори вниз незалежно від напрямку потоку.

При зображенні схем алгоритмів слід керуватися єдиною системою програмної документації, до якої входять держстандарти зі схем алгоритмів та програм.

3 При складанні схем алгоритмів обчислювальних процесів виділяють такі типові структури алгоритмів (процесів):

- лінійну;
- розгалужену;
- циклічну.

*Лінійний обчислювальний процес* характеризується тим, що кроки, на які він розбивається, виконуються послідовно в тому порядку, в якому вони подані. У практиці програмування лінійні алгоритми в чистому вигляді практично не зустрічаються.

У схемі лінійного обчислювального процесу символи визначення змінних, розрахунків проміжних і підсумкових змінних, виведення результатів обчислення розміщують послідовно.

4 Об'єктами опису в схемах алгоритмів є константи, змінні величини, вирази (арифметичні і логічні).

*Константа* – це величина, що має постійне значення, яке не залежить від операції, що виконується в алгоритмі. Наприклад, числа 7, -1.34, 0.00556 можуть бути константами.

*Змінна* – це величина, значення якої може змінюватися у процесі реалізації алгоритму. Змінна має своє власне ім'я – *ідентифікатор*. Значення змінної зберігається у пам'яті ЕОМ у машинному слові, позначеному цим ім'ям. В обчислювальних операціях використовується значення змінної, що зберігається у пам'яті ЕОМ. Змінна величина перед початком обчислювальних операцій повинна бути визначена, тобто їй повинно бути задане конкретне значення операціями введення або привласнення.

Результатом реалізації алгоритму теж є змінна, значення якої зберігається у пам'яті ЕОМ.

Розглянемо основні блоки, що використовуються у схемах алгоритмів, у таблиці 1.

Таблиця 1 – Найменування, умовні позначення символів та функції, які відображаються ними

№	Найменування	Графічне зображення	Функції
1	Процес		Виконання операції або групи операцій
2	Рішення		Перевірка умови (рішення)
3	Модифікатор		Початок циклу
4	Введення/виведення		Введення даних, виведення результатів
5	Підпрограма		Використання створених раніше та описаних окремо алгоритмів
6	Документ		Виведення до друку результатів на папір
7	З'єднувач (сторінковий)		Розрив лінії потоку
8	З'єднувач (міжсторінковий)		Розрив лінії потоку з переходом на іншу сторінку
9	Пуск/зупинка		Початок, кінець, зупинка

Приклад 1

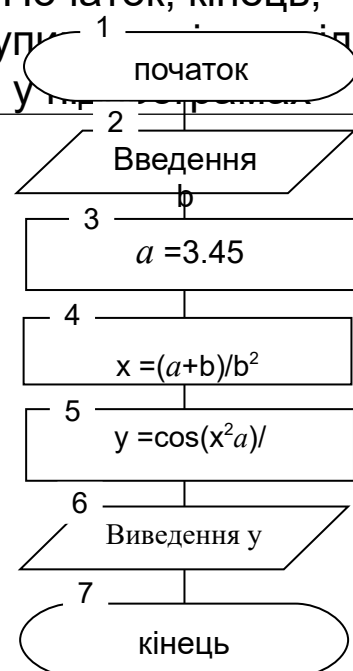


Рисунок 1

Скласти схему алгоритму обчислення функції  $y = \frac{\cos(x^2 a)}{\sqrt{x+a}}$ , якщо  $x = \frac{a+b}{b^2}$ ,  $a=3.45$ .  $b$  – будь-яке число більше 1.

При складанні схеми алгоритму (рисунок 1) скористуємось способом обчислення частинами.

Блоки процесу 2, 3 визначають змінні, блоки 4, 5, 6, 7 – розрахунок, блок 8 – виведення результатів. Всі символи алгоритму розміщені послідовно. Значення величин  $x$ ,  $y$  зберігаються у комірках пам'яті з відповідними іменами.

## Варіанти індивідуальних завдань

### 1 рівень

- 1  $m=3.2a+6.45b-c$ ,  $a=3.1$ ,  $c=2.5$ .
- 2  $y=z-d+\sqrt{d^2+1}+d^2+1$ ,  $z=6.3$ .
- 3  $y=x+c-mZ$ ,  $x=7.1$ ,  $m=4.2$ .
- 4  $x=3y+2py+sy$ ,  $y=2.3$ ,  $p=1.5$ .
- 5  $n=am^2+47a^2m^5$ ,  $m=1.7$ .

### 2 рівень

- 6  $t=a^2+mb-k$ ,  $k=m^3+a-b$ ,  $b=-6.66$ .
- 7  $x=0.05+\sin(a^2yz)$ ,  $z=3a+8a^2+a^3$ ,  $y=0.02$ .
- 8  $n=m^3+\sqrt{am}+\sin(am)$ ,  $m=b-0.65a$ ,  $b=0.35$ .
- 9  $s=c+b+b^2a$ ,  $c=0.5a^2+a+1$ ,  $b=2.5$ .
- 10  $f=\sqrt{\sin(s)+xa}$ ,  $s=ax^2+1$ ,  $a=6.2$ .

### 3 рівень

- 11  $k=89-\frac{165}{|b+a|}$ ,  $b=a^2$ ,  $a=7c^2+c+4$ .
- 12  $x=\sin(a^2b)+\frac{ba^3}{9}$ ,  $a=1.25-\cos(b^2)$ ,  $b=0.75c$ .
- 13  $k=\cos(a^2m)+\sqrt{a+m}$ ,  $m=\frac{a}{0.5b}$ ,  $b=0.53$ .



$$14 \quad x = \cos(m+a) - \sqrt{ma} + am, \quad a = \cos(m-5) + s, \quad s = m^2 c.$$

$$15 \quad p = 2\sin(s+1) + \sqrt{s+1} + \ln(s+1), \quad s = abc, \quad c = 6.31a - b.$$

## РОБОТА 2

### ПРОЕКТУВАННЯ АЛГОРИТМІВ РОЗГАЛУЖЕНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

**Мета роботи** – вивчення методики проектування алгоритмів розгалужених обчислювальних процесів.

#### **Завдання та порядок виконання**

- 1 Вивчити теоретичний матеріал.
- 2 Підготувати відповіді на контрольні запитання.
- 3 Скласти схеми алгоритмів згідно з варіантами завдання.

#### **Контрольні запитання**

- 1 Визначити поняття «розгалужений обчислювальний процес».
- 2 Визначити поняття «логічне відношення».
- 3 Перелічити операції відношення.
- 4 Визначити поняття «логічний вираз».
- 5 Які логічні операції використовуються у логічних виразах?

#### **Зміст звіту**

- 1 Номер роботи, її назва та мета.
- 2 Відповіді на контрольні запитання.
- 3 Схеми алгоритмів індивідуальних завдань та стислий їх опис.
- 4 Висновки до роботи.

## Навчальний матеріал

*Розгалуженим називають алгоритм*, в якому вибір дії залежить від виконання певних умов та значень вхідних даних або проміжних результатів.

Алгоритми розгалужених обчислювальних процесів являють собою структури, що мають декілька варіантів обчислень. Кожний варіант поданий окремою обчислювальною гілкою, вибір гілки здійснюється управляючою частиною алгоритму.

1 Управляюча частина – символи «рішення» (інша їх назва – логічні або умовні), з'єднані так, щоб для означеного набору вхідних даних або проміжних результатів гарантувалося виконання дій за єдиною гілкою. Вибір варіанта задається логічним відношенням або логічним виразом.

*Логічне відношення* - послідовний запис констант, змінних, арифметичних виразів, об'єднаних операціями відношення

$> =$ ,  $>$ ,  $=$ ,  $<>$ ,  $<$ ,  $< =$

*Логічний вираз* - послідовний запис логічних відношень, об'єднаних знаками логічних операцій:

логічне множення або операція «І» позначається знаком  $\wedge$ ,

логічне додавання або операція «АБО» позначається знаком  $\vee$ ,

логічне заперечення або операція «НІ» позначається знаком  $\neg$ .

Загальний вигляд схеми розгалуженого процесу приведений на рисунку 2. Залежно від результату перевірки умови виконується Дія 1 або Дія 2. Кожна з дій може являти собою складову структуру з будь-яких символів.

Окремим випадком розгалуження є структура

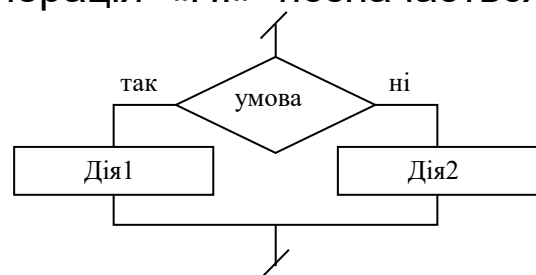


Рисунок 2

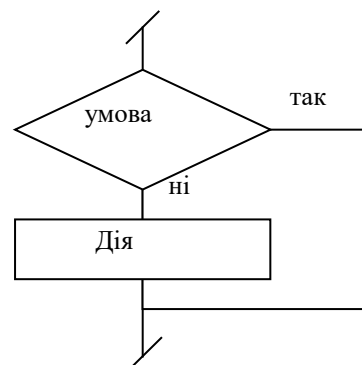


Рисунок 3

"обхід", в якій одна гілка не містить жодних дій (рисунк 3).

Узагальненням розгалуження є структура "множинний вибір", в якій одна з дій обирається залежно від значення управляючої змінної (рисунк 4).

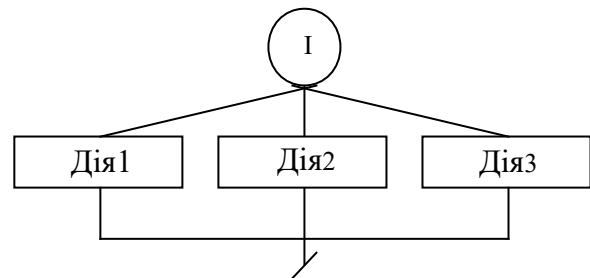


Рисунок 4

### Приклад 1

Скласти схему алгоритму обчислення значення функції:

$$y = \begin{cases} 2x + 3, & \text{якщо } x < 0 \\ 4x - 7, & \text{якщо } x \geq 0. \end{cases}$$

Схема алгоритму наведена на рисунку 5. У вихідних даних указані дві умови ( $x < 0 \wedge x \geq 0$ ), але в алгоритмі достатньо записати одну з них ( $x < 0$ ). Цей блок має два виходи: вихід на блок 4, якщо  $x < 0$ , та вихід на блок 5, якщо  $x \geq 0$ .

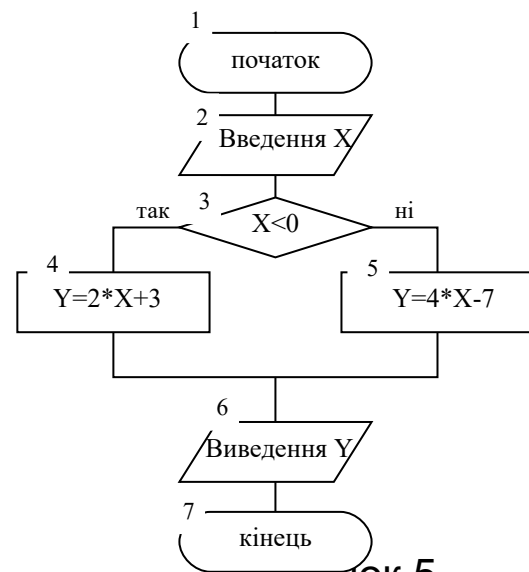


Рисунок 5

### Приклад 2

Скласти схему алгоритму обчислення функції Z:

Схема алгоритму наведена на рисунку 6.

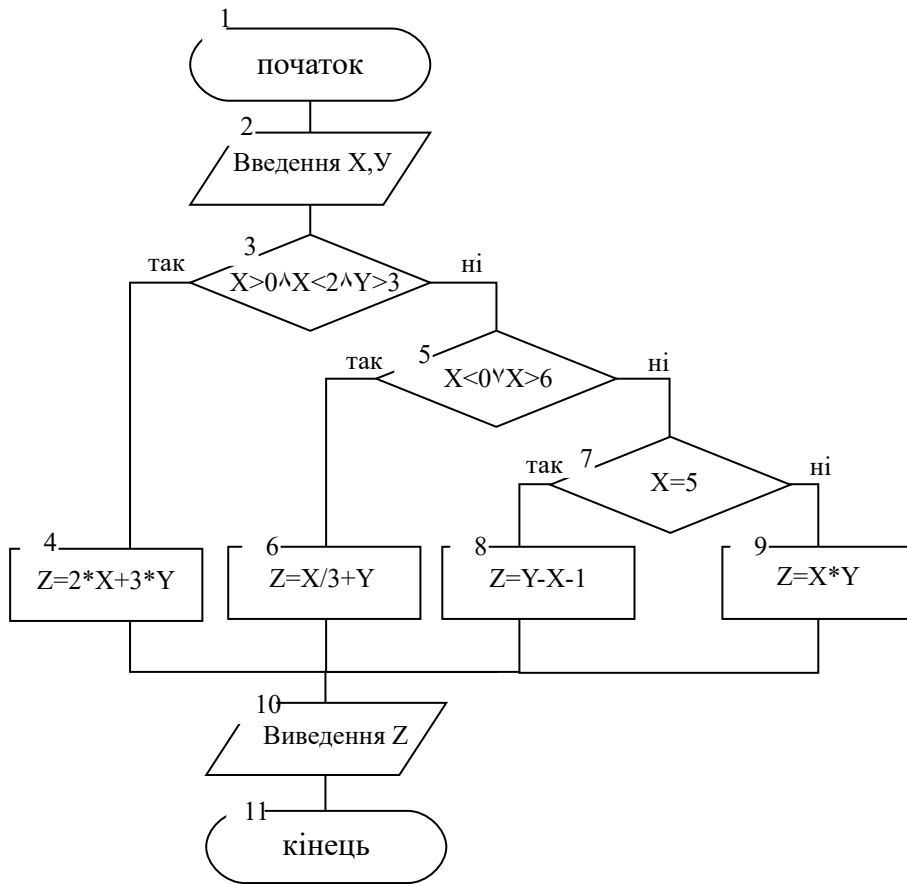
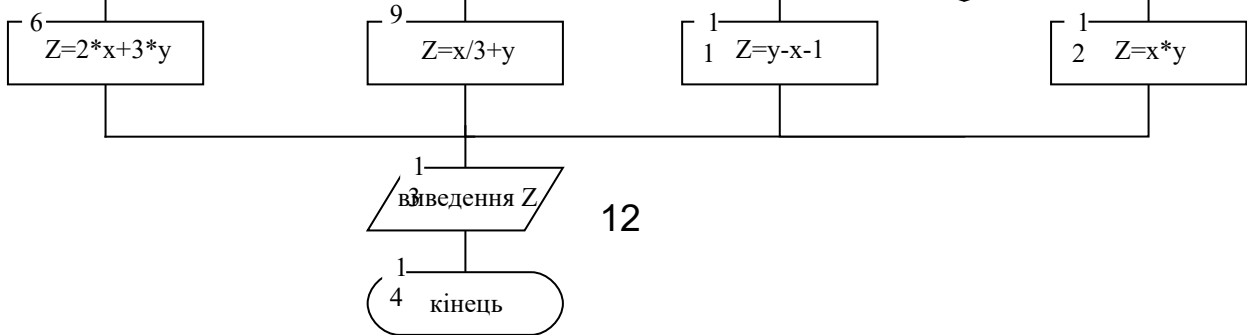


Рисунок 6

Управляюча частина схеми, яка забезпечує вибір одного з багатьох шляхів обчислення функції  $Y$ , подана трьома блоками «рішення» (блоки 3, 4, 5).

Логічний вираз, який виступає в межах як одного логічного блоку, декілька виражень, що містять за допомогою логічних операцій.

Наприклад,  $A < X < B$  треба вважати як  $A < X \wedge X < B$ . На рисунку 7 наведена схема алгоритму обчислення функції  $Z$ , в якому логічний вираз відношення. Блоки 3, 4, 5 реалізують логічне додавання.



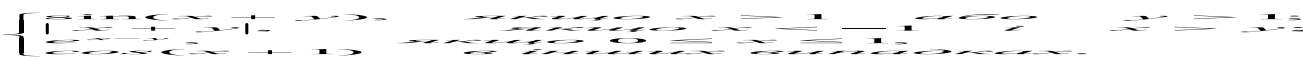
## Рисунок 7

### Варіанти індивідуальних завдань

#### 1 рівень

1

Z=



x,y – довільні значення

$$2 \ y = \begin{cases} |x^3|, & \text{якщо } x < ab \text{ і } x < -2; \\ \sin(x^a + b), & \text{якщо } x > ab \text{ або } x > 2; \\ a + b + 1, & \text{якщо } 0 < x < 1 \text{ і } x = ab; \\ x - (a + b)^2 & \text{в інших випадках.} \end{cases}$$

x,a,b – довільні значення

$$3 f = \begin{cases} c + x, & \text{якщо } 1 \leq x \leq 3; \\ c - x, & \text{якщо } 3 < x \leq 10 \text{ і } c > 0; \\ |c|^x, & \text{якщо } x > 10; \\ x & \text{в інших випадках.} \end{cases}$$

**c, x – довільні значення**

$$4 w = \begin{cases} e^v, & \text{якщо } v > x \text{ або } v > y; \\ (x + y), & \text{якщо } v = 10 \text{ і } y > x > 15; \\ v^{x+y}, & \text{якщо } v = x; \\ 1 & \text{в інших випадках.} \end{cases}$$

**v, x, y – довільні значення**

$$5 t = \begin{cases} s + 2z, & \text{якщо } s + z = 2 \text{ або } s + z = 1; \\ s - z^2, & \text{якщо } s + z > 5 \text{ і } z > 0; \\ |s + z|, & \text{якщо } 3 < s + z \leq 5; \\ a + s & \text{в інших випадках.} \end{cases}$$

**a, s, z – довільні значення**

## 2 рівень

$$6 y = \begin{cases} a + b + c, & \text{якщо } a + b > c; \\ (a + b) - c, & \text{якщо } a + b < c \text{ і } c > 5; \\ |\sin(b)|, & \text{якщо } a + b = c; \\ a^{bc} & \text{в інших випадках.} \end{cases} \quad z = \begin{cases} 1, & \text{якщо } y < 0; \\ 2, & \text{якщо } 0 \leq y < 5; \\ 3, & \text{якщо } 5 \leq y < 10; \\ 4, & \text{якщо } y \geq 10. \end{cases}$$

**a, b, c – довільні значення**

$$7 t = \begin{cases} z + x - \sin(y), & \text{якщо } z + x > \sin(y); \\ x - z \sin(y), & \text{якщо } z + x = \sin(y); \\ \ln|xyz|, & \text{якщо } z + x < \sin(y). \end{cases} \quad s = \begin{cases} \end{cases}$$

**t – 1, якщо  $0 < zt < 5$ ;**

**z, якщо  $zt > 10$  або  $zt < -3$ ;**

**t, якщо  $zt = 0$ ;**

**0, в інших випадках.**

**y, z, x – довільні значення**

$$8 \ m = \begin{cases} \sin^2(x) + \cos(x), & \text{якщо } x > 0; \\ \frac{x+y}{a^2+2} + \sin\left(\frac{x+y}{a^2+2}\right), & \text{якщо } x < 0 \text{ і } a \geq 0; \\ x + |a - y| & \text{в інших випадках} \end{cases}$$

$$n = \begin{cases} |m+c|, & \text{якщо } m > 10 \text{ або } m = 4; \\ m-c, & \text{якщо } m < 3; \\ m, & \text{якщо } 5 < m < 8; \\ mc & \text{в інших випадках.} \end{cases}$$

$x, y, a, c$  – довільні значення

$$9 \ x = \begin{cases} t+q, & \text{якщо } t > 10 \text{ і } t = q; \\ t+1, & \text{якщо } t \leq -10; \\ t, & \text{якщо } t = 5; \\ 1+tq & \text{в інших випадках} \end{cases} \quad y =$$

$$\begin{cases} \sin(x) + 1, & \text{якщо } x = 1; \\ \ln|x+1|, & \text{якщо } x = 2; \\ \frac{1}{x}, & \text{якщо } x = 3; \end{cases}$$

$t, q$  – довільні значення

$$10 \ k = \begin{cases} x+y^2+2, & \text{якщо } x = y+2; \\ x+y+2, & \text{якщо } x > y+2 \text{ і } y = 3; \\ \sin(x) + \sin(2), & \text{якщо } 0 < x < y+2; \\ x + \ln|y+2y| & \text{в інших випадках.} \end{cases}$$

$$Z = \begin{cases} 2k, & \text{якщо } k < 2; \\ 3k, & \text{якщо } 2 \leq k \leq 6; \\ 4k, & \text{якщо } 6 < k < 10; \\ 5k, & \text{якщо } k \geq 10. \end{cases}$$

$x, y$  – довільні значення

### 3 рівень

$$11 \ a = \begin{cases} x \sin(y), & \text{якщо } x > y; \\ y \sin(x), & \text{якщо } x \leq y. \end{cases} \quad | = \begin{cases} \ln(a^2+1), & \text{якщо } 15 \leq a < 20; \\ 2a+7, & \text{якщо } 10 \leq a < 15; \\ 1-ab, & \text{якщо } 8 < a < 10; \\ \frac{4a}{7(a+b)^2+5}, & \text{якщо } 0 \leq a < 8 \text{ і } a > b; \\ a+b & \text{в інших випадках.} \end{cases}$$

$$p = \begin{cases} t+a, & \text{якщо } t > 0 \text{ або } a > 0; \\ t-a, & \text{якщо } t < 0 \text{ і } t < 0; \\ |a| & \text{в інших випадках,} \end{cases} \quad b = \ln|a+xy|, \quad t = 3l - 1.$$

$x, y$  – довільні значення

$$12 \quad a = b \cos(b)$$

$$z = \begin{cases} 25y + a, & \text{якщо } y > a \text{ і } a > 10; \\ y + 3a, & \text{якщо } a < y < 20; \\ \cos(ya), & \text{якщо } y = a; \\ 1 & \text{в інших випадках.} \end{cases}$$

$$v =$$



$$w = \begin{cases} v + 1, & \text{якщо } v + 1 \leq 0; \\ \frac{1}{v^2 + 1}, & \text{якщо } v + 1 > 0. \end{cases}$$

$y, b$  – довільні значення

$$13 \quad x = \begin{cases} t^3 + 5, & \text{якщо } t < -2; \\ t - 1, & \text{якщо } -2 \leq t < 0; \\ \sin(t), & \text{якщо } 0 \leq t < 5; \\ \ln(t), & \text{якщо } t \geq 5. \end{cases} \quad z = \begin{cases} e^x, & \text{якщо } x < 0; \\ x^3, & \text{якщо } x \geq 0. \end{cases}$$

$$y = \begin{cases} a + z, & \text{якщо } a + z > 0 \text{ і } z < 0; \\ az, & \text{якщо } z > 5 \text{ або } a + z < -3; \\ a - x & \text{в інших випадках.} \end{cases}$$

$$f = \begin{cases} \frac{1}{(ay)^2 + 1}, & \text{якщо } y = -4; \\ 2xy, & \text{якщо } xy > 0 \text{ і } x > 0; \\ \frac{y}{x^4 + 1} & \text{в інших випадках.} \end{cases}$$

$a, t$  – довільні значення

$$14 \quad b = t^3 + t - 1 \quad c = \begin{cases} 1, & \text{якщо } b < 0; \\ 2, & \text{якщо } b \geq 0. \end{cases} \quad d = \begin{cases} 1, & \text{якщо } a \leq 1; \\ 3, & \text{якщо } 1 < a \leq 3; \\ 5, & \text{якщо } a > 3. \end{cases}$$

$$a = \begin{cases} b^3, & \text{якщо } b > 0 \text{ і } c > 0 \text{ або } b > 10; \\ b + c, & \text{якщо } b < 0 \text{ і } c = 0; \\ \frac{1}{(bc)^2 + 5}, & \text{якщо } b = 0 \text{ або } -5 < c < -1; \\ \frac{1}{(b+1)^2 + 3} & \text{в інших випадках.} \end{cases}$$

$$f = \begin{cases} abd, & \text{якщо } ab < d + 1; \\ \frac{ab}{d^4 + 1}, & \text{якщо } ab = d + 1; \\ \frac{ad}{b^2 + 2}, & \text{якщо } ab > d + 1. \end{cases} \quad t \text{ – довільні значення}$$



$$15 \quad x = \begin{cases} t+1, & \text{якщо } t < 1; \\ t, & \text{якщо } t = 1; \\ t-1, & \text{якщо } t > 1. \end{cases} \quad y = \begin{cases} a^3 + x, & \text{якщо } a < x < 10 \text{ і } a < 10; \\ a + \sin(x), & \text{якщо } 10 < x \leq 15 \text{ і } x = a; \\ \ln|xa|, & \text{якщо } x = 20 \text{ або } 25 < a \leq 30; \\ e^{x-a} & \text{в інших випадках.} \end{cases}$$

$$p = \begin{cases} 1+z, & \text{якщо } z = 0 \text{ і } y = 0; \\ z+y, & \text{якщо } zy < 0; \\ 25 & \text{в інших випадках.} \end{cases} \quad z = a^2 + y^3 - 5.$$

$t, a$  – довільні значення.

## РОБОТА 3

### ПРОЕКТУВАННЯ АЛГОРИТМІВ ПРОСТИХ АРИФМЕТИЧНИХ ЦИКЛІЧНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ

**Мета роботи** – вивчення та засвоєння методики проектування алгоритмів арифметичних циклічних обчислювальних процесів.

#### Завдання та порядок виконання

- 1 Підготувати відповіді на контрольні запитання.
- 2 Скласти схеми алгоритмів згідно з варіантами завдань.

#### Контрольні запитання

- 1 Визначити поняття «цикл», «циклічний обчислювальний процес».
- 2 Визначити поняття «параметр циклу».
- 3 Що необхідно виконати, щоб організувати арифметичний цикл?
- 4 Визначити поняття «рекурентний вираз».
- 5 Навести приклади рекурентних алгоритмів.

6 Які переваги дає застосування рекурентних алгоритмів?

### **Зміст звіту**

- 1 Номер роботи, її назва.
- 2 Стислий зміст навчального матеріалу та відповіді на контрольні запитання.
- 3 Схеми алгоритмів та стислий їх опис.
- 4 Висновки до роботи.

### **Навчальний матеріал**

У практиці інженерних розрахунків доводиться виконувати багаторазове обчислення за однаковими математичними залежностями при різних значеннях вхідних даних. Такі обчислення, що повторюються, називають *циклами*.

*Циклічним називається алгоритм*, який містить послідовність операцій, що виконуються багаторазово. Використання циклів дозволяє значно зменшити схему алгоритму та розмір відповідної програми.

За способом задання кількості повторень тіла циклу можна виділити арифметичні та ітераційні цикли.

В *ітераційних циклах* кількість повторень, як правило, не може бути визначена до початку його виконання і залежить від заданої умови виходу з циклу.

*Арифметичними (регулярними)* називають цикли, кількість повторень яких може бути явно задана в умові задачі або при необхідності розрахована до початку його виконання. У таких циклах присутня змінна, яка називається параметром циклу. Параметр циклу має бути визначеним межами та кроком його змінювання, які задаються константами, змінними або виразами. Арифметичні цикли з одним параметром називаються простими. Умова закінчення арифметичного циклу може перевірятися у різних місцях алгоритму.

*Цикл з після-умовою (ЦИКЛ-ДО)* (рисунок 8). Перевірка умови закінчення циклу здійснюється у кінці циклу після

виконання певних дій (тіла циклу). У зв'язку з цим тіло циклу завжди виконується хоча б один раз.

**Цикл з перед-умовою (ЦИКЛ-ПОКИ)** (рисунок 9). Умова перевіряється на початку циклу до виконання тіла циклу, тому тіло циклу може взагалі не виконуватися.

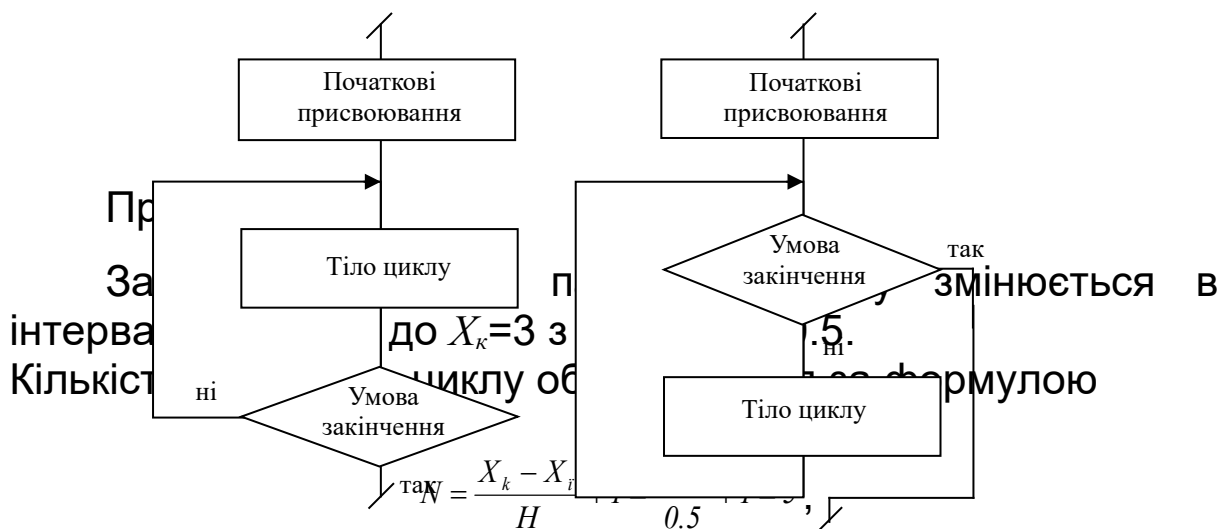


Рисунок 8

Рисунок 9

де  $X_{П}$  – початкове значення параметра циклу;

$X_{К}$  – кінцеве значення параметра циклу;

$H$  – крок зміни параметра.

Для організації простого арифметичного циклу необхідно виконати такі дії:

- задати початкове значення параметра циклу;
- перевірити, чи входить значення параметра циклу до інтервалу змінення (умова закінчення циклу). Якщо значення параметра не перевищує кінцевого значення, то виконуються наступні обчислення (процес виконується перший раз або повторюється), в іншому випадку здійснюється вихід з циклу;
- здійснити задані обчислення та, якщо потрібно, друкування результатів;
- змінити параметр циклу на величину кроку;
- повернутися до етапу перевірки умови закінчення циклу.

## Приклад 2

Скласти схему алгоритму обчислення значень функції  $Y=\sin(XA)-B$ .

Параметр циклу  $X$  змінюється в інтервалі від  $X_{\Gamma}$  до  $X_{\kappa}$  з кроком  $H_x$ ,

де  $X_{\Gamma}=C$ ,  $X_{\kappa}=D$ ,  $H_x=H$ ,  $A=9.63$ ,  $B=5.1$ ;

$H$ ,  $C$ ,  $D$  – довільні числа ( $C < D$ ).

Схема алгоритму задачі являє собою простий циклічний обчислювальний процес з неявно заданою кількістю повторень.

У схемі алгоритму, що наведена на рисунку 10, цикл організований з використанням блока 5 «модифікатор». У цьому блоці визначені початкове, кінцеве значення параметра циклу  $X$  та крок його зміни. Обчислення значення функції виконується у блоці 6.

На рисунку 11 наведена схема цього алгоритму з блоком «рішення». Її блоки 5, 6 та 9 відповідають блоку 5 схеми рисунка 10.

Призначення символів:

блоки 2–4 – формування вхідних даних;

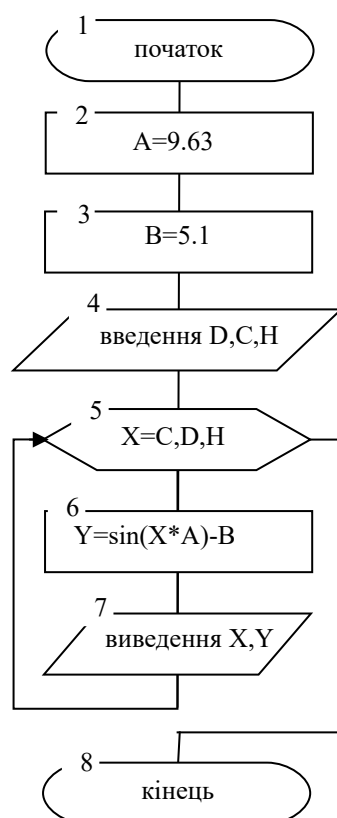
блок 5 – задання початкового значення параметра  $X$ ;

блок 6 – перевірка умови закінчення циклу;

блок 7 – обчислення функції;

блок 8 – друкування обчисленого значення функції та поточного значення параметра циклу;

блок 9 – обчислення наступного значення параметра циклу.



20

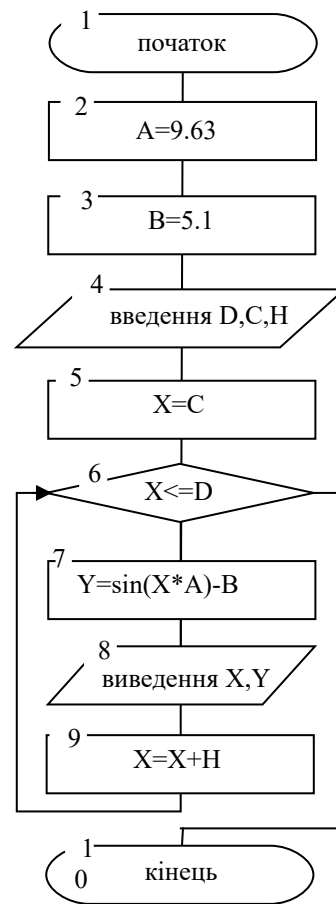


Рисунок 10

Рисунок 11

*Рекурентний вираз* – це вираз, який описує будь-який член послідовності чисел. Обчислення за рекурентними виразами реалізується циклічними обчислювальними процесами.

Наприклад, формула  $X=X+1$  означає: до складу  $X$  додати 1 та результат записати у  $X$ , тобто рекурентний вираз пов'язує між собою послідовно обчислені значення. Вхідними даними для наступного кроку є результати попереднього.

Рекурентні вирази використовуються для обчислення суми або добутку кінцевого числа даних. Для того щоб обчислити суму деякого числа даних, необхідно виконати дії:

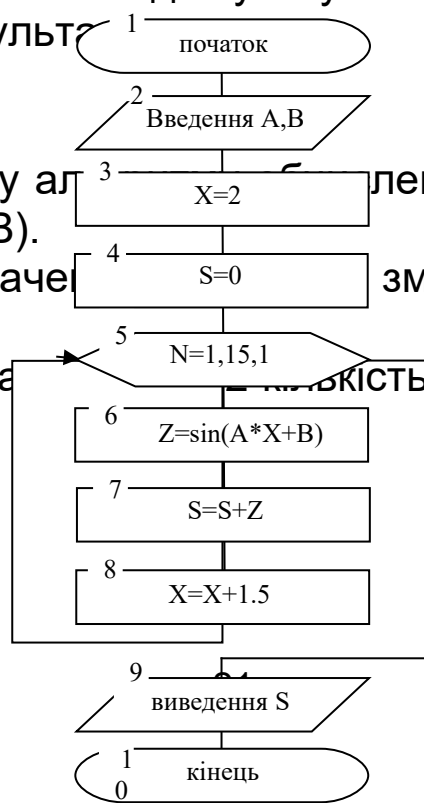
- сформулювати початкові дані;
- визначити початковий стан суматора, в якому буде здійснюватися накопичування суми;
- організувати цикл накопичування суми шляхом додавання нових значень до суми усіх попередніх;
- вивести результат

Приклад 3

Скласти схему алгоритму обчислення суми 15 значень функції  $Z=\sin(AX+B)$ .

Початкове значення  $X=2$ ; кількість повторень задана в умові задачі.  $A, B$  - довільні числа.

В алгоритмі наведеному змінна  $N=1,15,1$  - кількість повторень задана в умові задачі.



## Рисунок 12

Призначення символів:  
блоки 2–3 – формування вхідних даних;  
блок 4 – підготовка (обнулення) суматора;  
блоки 5–8 – організація циклу з параметром N (кількість обчислень функції Z); розрахунок значень функції Z; додавання отриманих значень; збільшення X на розмір кроку;  
блок 9 – друкування значення суми.

### Варіанти індивідуальних завдань

1 рівень

$$1 \quad z = \cos(x), \quad x = 24e^t, \quad t \in [3;9], \quad h_t = 1.5$$

2

$$m = 2.49x^3 - \frac{b}{x^2 + s^2}, \quad s = 3.63 \sin(a + x), \quad x \in [-1; a/2],$$

3

$$t = 2m - \sin^2(x) + \frac{3}{2x}; \quad m = 36x^2; \quad x \in [0.2; 1.6]; \quad h_x =$$

4

$$p = 3b^2 - \frac{\ln|a-c|}{a+2b}; \quad b = 0.75a^2 + c; \quad a \in [-3; 3]; \quad h_a = 0.5;$$

5

$$z = \cos(ay^2 + 1); \quad y = b + d^2 - e^{-x}; \quad x \in [-3; b]; \quad b = 2;$$

2 рівень

$$6 \quad n = 3; \quad e = \sum_{i=1}^n i^2; \quad f = (e - n)!.$$

$$7 \quad n = 5; \quad a = (10 - n)!; \quad t = \sum_{k=1}^{20} \sin(a+k).$$

8

$$k = n!; \quad n = 2; \quad z = \sum_{i=1}^k (a+b)^i; \quad p = \prod_{i=1}^k (a+b)^i; \quad a = 2;$$

$$9 \quad c = \frac{m!}{n!(m-n)!}; \quad m = \sum_{i=1}^5 i; \quad n = \prod_{j=1}^3 j.$$

10

$$r = \cos\left(\sum_{i=1}^n \ln(i) + \prod_{j=1}^m \sin(j^3)\right); \quad f = 3; \quad n = f!; \quad m = (n-2)!.$$

3 рівень

11

$$y = b^2 \sin(\cos(x + m - n)); \quad x \in [100; 200]; \quad h_x = b^3 + b^2 +$$

$$m = \prod_{k=p}^r (a-b)^k; \quad a = 4; \quad b = 2; \quad t1 = 3; \quad t2 = 5; \quad p =$$

12

$$z = ax + by + c; \quad y = x^3 + x^2 + 1; \quad x = 2; \quad a = (x + 5)!;$$

$$b = \sum_{i=1}^n i^3 + \frac{1}{y!}; \quad c = \prod_{j=3}^5 (a+b)^j.$$

13

$$t = x! + y! + n; \quad x = \sum_{i=1}^4 i; \quad y = \prod_{i=1}^3 i; \quad n = \sum_{k=1}^5 (k-1).$$

$$14 \quad z = a \cos(a + b); \quad a \in [a1; a2]; \quad h_a = 50;$$

$$a1 = 5!; \quad a2 = (n + 5)!; \quad b = \sum_{i=1}^{a2-a1} i; \quad n = 1.$$

15

$$f = \frac{\sum_{i=1}^n (a+b)^i + \prod_{j=j1}^{j2} (ab)^j}{(n+4)!} - \sum_{k=1}^3 (a+b)^k \prod_{j=4}^6 b^j; \quad n = 4;$$

$$a = 0.3; \quad b = 1.7; \quad j1 = 3; \quad j2 = 6$$

## РОБОТА 4

### ПРОЕКТУВАННЯ АЛГОРИТМІВ ІТЕРАЦІЙНИХ ЦИКЛІЧНИХ ОБЧИСЛЮВАЛЬНИХ ПРОЦЕСІВ



**Мета роботи** – вивчення поняття “ітераційний цикл”, принципів його побудови і набуття практичних навичок розробки схем циклічних ітераційних алгоритмів.

### **Завдання і порядок виконання**

- 1 Вивчити теоретичний матеріал.
- 2 Підготувати відповіді на контрольні запитання.
- 3 Скласти схеми алгоритмів розв’язання задач трьох рівнів за заданими варіантами.

### **Контрольні запитання**

- 1 Дайте визначення ітераційного циклу.
- 2 Укажіть порядок побудови ітераційного алгоритму.
- 3 Як організується вихід з циклу в ітераційному алгоритмі?
- 4 Поясніть необхідність використання в ітераційних циклах рекурсивних співвідношень.
- 5 Як варто розуміти умову "...обчислити із заданою точністю" ?
- 6 Укажіть, для яких задач застосовуються ітераційні цикли.
- 7 Наведіть приклад ітераційного алгоритму.

### **Зміст звіту**

- 1 Номер роботи, її назва, визначення мети.
- 2 Короткий зміст теоретичного матеріалу і відповіді на контрольні запитання.
- 3 Схеми алгоритмів за варіантами завдань і короткий опис першої з них.
- 4 Висновки з роботи.

### **Навчальний матеріал**

Поряд з арифметичними циклами в інженерній практиці використовуються ітераційні циклічні обчислювальні процеси.

Ітераційний цикл – це циклічний обчислювальний процес, в якому кількість повторень тіла циклу заздалегідь

невідомо і залежить від умови досягнення шуканого результату. В ітераційних алгоритмах необхідно забезпечити обов'язкове виконання умови виходу з циклу, тобто збіжність ітераційного процесу.

### Приклад 1

Скласти схему алгоритму обчислення значення функції  $Y$ , яка подана сумою елементів нескінченного числового ряду, що сходиться, такого вигляду:

$$y = \frac{1}{1+a} + \frac{2}{1+a^2} + \frac{3}{1+a^3} + \frac{4}{1+a^4} + \dots + \frac{i}{1+a^i} + \dots$$

для значення  $A > 1.0$ .

Так як числовий ряд нескінченний, то для практичних розрахунків обмежуються деякою кількістю елементів, виходячи з вимоги заданої точності  $\varepsilon$  обчислення суми  $Y$ .

Числовий ряд, що сходиться, – це ряд величин (елементів ряду), значення кожної з яких менше значення попередньої величини цього ряду. Сума елементів такого ряду – це скінченна величина і її можна знайти. Практично обчислення суми елементів припиняють на черговому елементі, що за своїм значенням менше, ніж задана точність  $\varepsilon$ . Усіма наступними елементами (які виявляються меншими за  $\varepsilon$ ), знехтуємо. Якщо значення елементів змінюються так, як показано на рисунку 13, то до суми будуть включені тільки чотири перші елементи. П'ятий елемент і наступні за ним до суми не включаються.

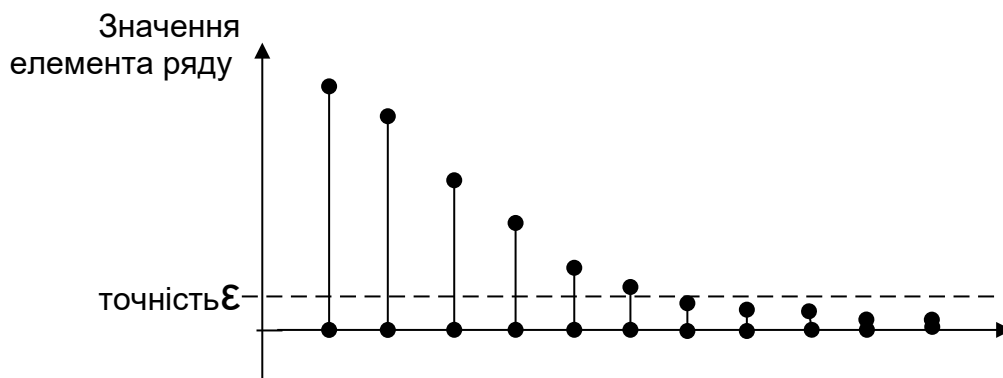


Рисунок 13 – Залежність значення елемента ряду, що сходиться, від його номера

На рисунку 14 показано алгоритм знаходження суми  $Y$ . Накопичення суми елементів виконується у символі 7 за допомогою рекурсивної залежності

$$Y = Y + Z,$$

$Z$  – значення чергового обчисленого доданка суми.

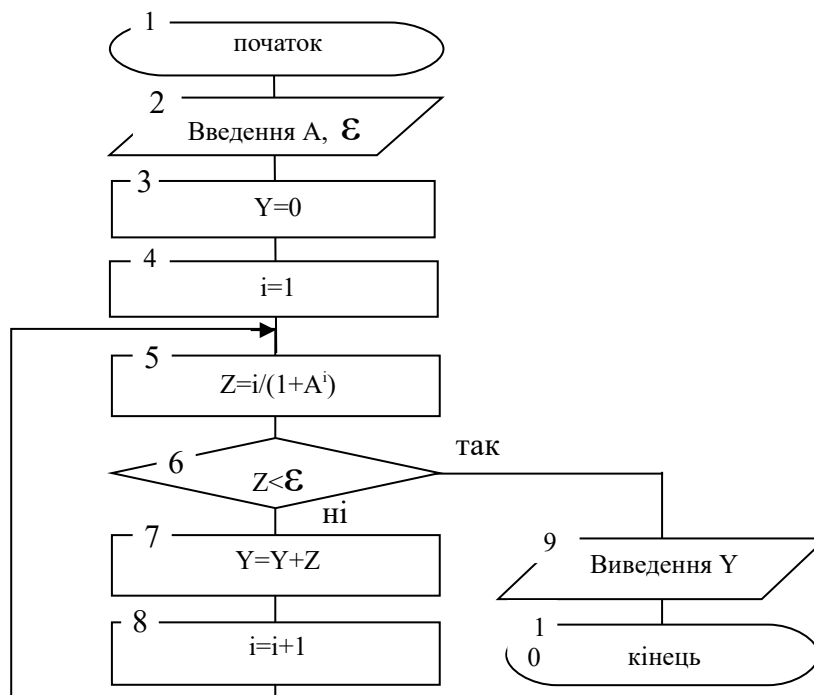


Рисунок 14 – Схема алгоритму ітераційного обчислювального процесу до прикладу 1

Кожний наступний доданок відрізняється від попереднього за видом залежності (змінюється чисельник, показник степеня у знаменнику). Щоб відстежити ці зміни, в алгоритмі використана допоміжна змінна  $i$ . Вона вказує номер доданка, що обчислюється, і за її значенням можна визначити поточний доданок  $Z=i/(1+A^i)$ .

Блок 6 – перевірка: чи слід даний доданок додати до суми. Це варто робити, якщо доданок  $>\varepsilon$  і його значенням не можна знехтувати. Перевірка рядів, у яких можуть існувати негативні елементи, виконується за модулем значення елемента. Блоки 3 і 4 задають початкові значення суми  $Y$  і допоміжної величини  $i$ .

Отже, ітераційні алгоритми для обчислення сум нескінченних рядів будуються у такому порядку:

- вводяться необхідні вхідні дані;
- задається початкове значення суми і допоміжних змінних, якщо це необхідно;
- обчислюється значення поточного елемента;
- виконується порівняння елемента суми із заданою точністю  $\varepsilon$ ;
- якщо елемент не менший за величину  $\varepsilon$ , то він додається до накопиченої суми, змінюються значення допоміжних змінних, після чого обчислюється черговий елемент і цикл повторюється;
- якщо елемент менший за  $\varepsilon$ , то нагромадження суми елементів ряду припиняється і виводиться отриманий результат.

При побудові ітераційного обчислювального процесу неприпустимо використання символу “модифікатор”, тому що в ітераційному процесі немає керуючої змінної (параметра) циклу.

Вихід з ітераційного циклу може бути виконаний на будь-якому кроці при досягненні заданої умови точності. Маємо на увазі, якщо в приведеному прикладі величина  $A$  буде вводиться різною, то різним буде значення чергового доданка. Таким чином, кількість повторень при одній і тій же точності спрогнозувати заздалегідь неможливо.

## Приклад 2

Скласти схему алгоритму обчислення з точністю  $\varepsilon = 10^{-5}$  суми елементів ряду

$$Y = B + \frac{A}{2X+1} - \frac{A^2}{6X+2} + \frac{A^3}{24X+3} - \frac{A^4}{120X+4} + \dots + \frac{(-1)^{i+1} A^i}{i! * (2X+i)}$$

виведення на друк всіх елементів, що входять до суми, і визначення їх кількості.

В умові поставлена задача визначити кількість елементів ряду. Допоміжна змінна  $i$  вказує на порядковий номер поточного елемента  $i$ , якщо для останнього, включеного до суми елемента, виведемо значення  $i$ , то це буде кількість елементів ряду.

У даному алгоритмі проглядається типова послідовність дій (рисунок 15).

Блоки 2-4 виконують введення вхідних даних і задають початкові значення суми  $Y$  і допоміжної змінної  $i$ . Тут  $Y$  привласнюється значення першого елемента суми, який не укладається у загальну формулу елементів ряду; величині  $i$  привласнюється значення 2, що дозволяє зручніше зробити загальний початок для елементів ряду.

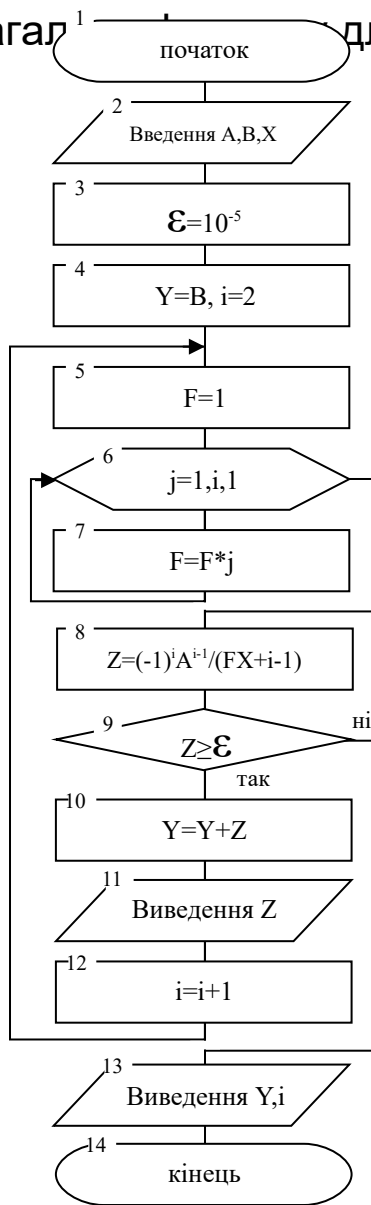


Рисунок 15 – Схема алгоритму ітераційного обчислювального процесу до прикладу 2

Блоки 5-8 необхідні для обчислення чергових елементів ряду. З них блоки 5-7 реалізують обчислення факторіала  $F=i!$  і далі, у блоці 8 значення факторіала використовується у вигляді змінної  $F$ .

Блок 11 необхідний для друкування значення чергового елемента. Завдяки тому що цей символ знаходиться усередині циклу, друкування буде багаторазовим, за кількістю повторень циклу, тобто за кількістю елементів ряду.

Блок 13 виводить  $Y$ –суму елементів ряду та  $i$  – номер останнього елемента.

В інших постановках задач з використанням ітераційних циклів обчислення чергового елемента і його наступна обробка можуть бути більш складними, однак основна послідовність дій буде тією ж самою.

## Варіанти індивідуальних завдань

### 1 рівень

Обчислити суми елементів рядів, що сходяться, із заданою точністю  $\varepsilon = 10^{-5}$ .

$$1 \quad y = 2 + \sum_{n=1}^{\infty} \frac{x^{n+1}}{n!}.$$

$$2 \quad y = 1 + \sum_{n=1}^{\infty} (-1)^n \frac{n!}{x^n}.$$

$$3 \quad y = x + \sum_{n=1}^{\infty} \frac{(\ln x)^n}{n!}.$$

$$4 \quad y = a + \sum_{n=1}^{\infty} \frac{x^{2n}}{(2n)!}.$$

$$5 \quad y = \sum_{n=1}^{\infty} (-1)^{n-1} \frac{x^n}{n}.$$

## 2 рівень

Обчислити суми елементів рядів, що сходяться, із заданою точністю  $\mathcal{E} = 10^{-5}$  і кількість елементів ряду

$$6 \quad y = 0.5 + \sum_{n=1}^{\infty} \frac{(3n)!}{x^{2n+1}}.$$

$$7 \quad y = \sum_{n=1}^{\infty} \frac{(x-1)^n}{nx^n}.$$

$$8 \quad y = 1 + \sum_{n=1}^{\infty} \frac{(x \ln a)^n}{n!}.$$

$$9 \quad y = a^2 + \sum_{n=2}^{\infty} (-1)^n \frac{(n-1)!}{x^n}.$$

$$10 \quad y = x + \sum_{n=2}^{\infty} \frac{(n+1)x}{n!a^n}.$$

## 3 рівень

Обчислити суми елементів рядів, що сходяться, із заданою точністю  $\mathcal{E} = 10^{-5}$ , кількість елементів ряду, а також кількість елементів, які  $> \mathcal{E}$  і  $< 10^* \mathcal{E}$ .

$$11 \quad y = 2 + \sum_{n=0}^{\infty} \frac{1}{a^n + \sin^{n+2}(x)}.$$

$$12 \quad y = 1 + \sum_{n=1}^{\infty} (-1)^n \frac{x^{2n}}{(2n)!}.$$

$$13 \quad y = \sum_{m=2}^{\infty} (-1)^{m-1} \frac{x^{2m-1}}{(2m)! - 1}.$$

$$14 \quad y = \sum_{n=0}^{\infty} (-1)^n \frac{(n+1)!}{x^{2n+1}}.$$

$$15 \quad y = b + \sum_{n=0}^{\infty} \frac{x^{2n+1}}{(2n+1)!}.$$

## РОБОТА 5

### ПРОЕКТУВАННЯ СХЕМ АЛГОРИТМІВ ВИЗНАЧЕННЯ НАЙБІЛЬШОГО ТА НАЙМЕНШОГО ЗНАЧЕННЯ ФУНКЦІЇ

**Мета роботи** – вивчення та освоєння методики проектування алгоритмів пошуку найбільшого (MAX) та найменшого (MIN) значення функції.

### **Завдання та порядок виконання**

1 Вивчити навчальний матеріал і підготувати відповіді на контрольні запитання.

2 Побудувати схеми алгоритмів задач відповідно до варіанта завдання.

### **Контрольні запитання**

1 Визначити поняття найбільшого (MAX) та найменшого (MIN) значення функції.

2 Який тип алгоритму використовується при знаходженні MAX та MIN значення функції?

### **Зміст звіту**

1 Номер роботи, її назва, визначення мети.

2 Короткі відповіді на контрольні запитання.

3 Висновки за результатами роботи.

### **Навчальний матеріал**

Знаходження MIN та MAX значення функції припускає, що функція визначена на заданому інтервалі та необхідно визначити найбільше (MAX) чи найменше (MIN) серед них. Оскільки аналітичне вираження функції незмінне на інтервалі визначення, то різні її значення утворюються за рахунок зміни параметрів (аргументів) цієї функції. Тому процедура визначення MIN, MAX значення функції виконується у циклі.

Перед початком циклу за визначенням MIN чи MAX значення функції  $Y=f(x)$  обчислюється перше значення  $Y_1$  функції, яке приймається за MIN чи MAX, тобто  $Y_{\min}=Y_1$  чи  $Y_{\max}=Y_1$ . Далі для обчислення наступного значення функції заміняємо значення параметра циклу та, якщо нове



значення параметра припустимо, обчислюємо відповідне значення функції. Отримане значення функції  $Y_2$  залежно від умови задачі порівнюємо при знаходженні MIN з  $Y_{\min}$ , при знаходженні MAX – з  $Y_{\max}$ .

Якщо  $Y_2 < Y_{\min}$ , то  $Y_{\min}$  приймає значення  $Y_2$  ( $Y_{\min} = Y_2$ ), а якщо  $Y_2 \geq Y_{\min}$ , то  $Y_{\min}$  залишає своє значення, після чого здійснюється перехід до обчислення чергового значення параметра. Після зміни параметра циклу процес повторюється.

При знаходженні MAX  $Y_2$  порівнюється з  $Y_{\max}$ . Якщо  $Y_2 > Y_{\max}$ , то  $Y_{\max}$  приймає значення  $Y_2$  ( $Y_{\max} = Y_2$ ), інакше  $Y_{\max}$  зберігає своє значення і після зміни параметра циклу процес повторюється.

Все це можна описати математичною залежністю:

$$MAX: Y_{\max} = \begin{cases} Y_i, & \text{якщо } Y_i > Y_{\max} \\ Y_{\max}, & \text{якщо } Y_i \leq Y_{\max} \end{cases};$$

$$MIN: Y_{\min} = \begin{cases} Y_i, & \text{якщо } Y_i < Y_{\min} \\ Y_{\min}, & \text{якщо } Y_i \geq Y_{\min} \end{cases}$$

де  $Y_i$  – чергове значення функції.

Вихід з циклу здійснюється після досягнення параметром верхньої межі інтервалу. Слід відмітити, що при розв'язанні таких задач йдеться не про MIN чи MAX функції, а про MIN чи MAX серед обчислених значень цієї функції. Це пояснюється тим, що комп'ютер обчислює дискретні значення функції при відповідних дискретних значеннях параметра та дійсний MIN чи MAX може бути між ними (рисунок 16). Підвищити точність визначення MIN чи MAX можна за рахунок зменшення кроку зміни параметра циклу.

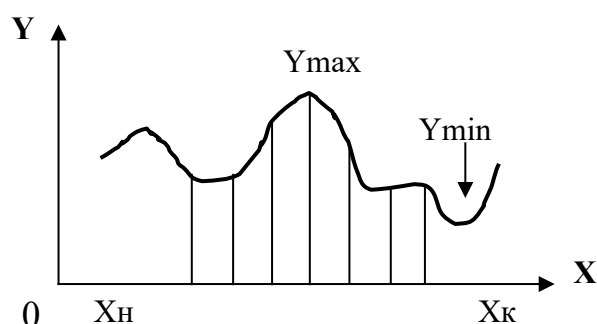


Рисунок 16

### Приклад 1

Скласти схему алгоритму пошуку MIN значення функції  $Y = |A| e^{(Bx)}$ , при зміні параметра X від 0 до 4 з кроком  $h=0.5$

Алгоритм пошуку MIN значення цієї функції приведений на рисунку 17:

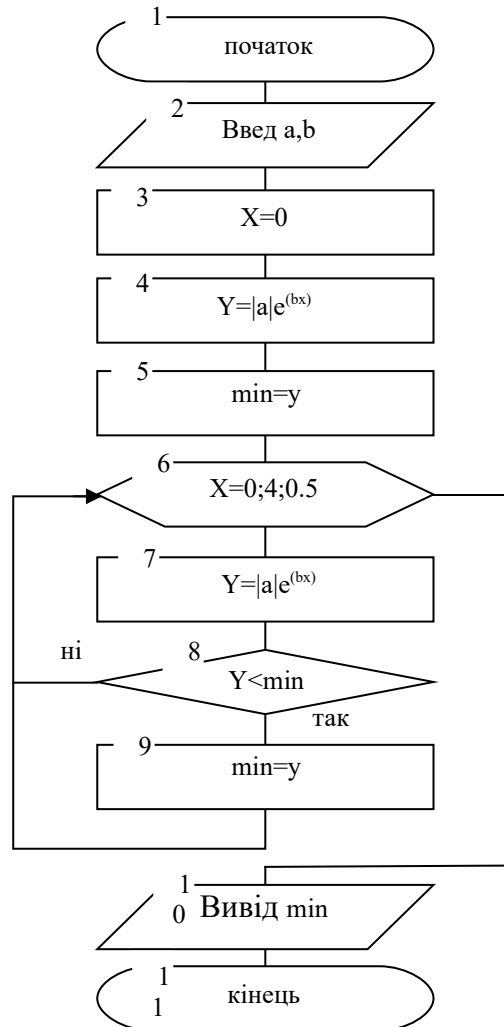


Рисунок 17

- блок 2 – введення змінних a, b;
- блок 3 – надання початкового значення параметру циклу x;
- блок 4 – обчислення першого значення функції;
- блок 5 – MIN присвоюється значення першого значення функції;
- блок 6 – блок керування циклом;

блок 7 – обчислення чергового значення функції;

блок 8 – порівняння чергового значення функції з MIN; якщо чергове значення функції менше, ніж MIN, то MIN приймає це значення (блок 9) та здійснює перехід до чергового значення параметра. У протилежному випадку MIN зберігає своє значення та здійснює перехід до чергового значення параметра;

блок 10 – друкування MIN значення функції;

блок 11 – вихід з алгоритму.

## Варіанти індивідуальних завдань

### 1 рівень

1 Скласти схему алгоритму пошуку MIN значення функції:

$$Y = A * e^{-B * X} * \sin(W * X + F), \quad x \in [0, T], \quad Hx = H.$$

2 Скласти схему алгоритму пошуку MAX значення функції:

$$Y = A * e^{-B} + C * e^{-D}, \quad B \in [W, F], \quad Hb = H.$$

3 Скласти схему алгоритму пошуку MAX значення функції:

$$Y = \cos(C * Z) * (X + Y), \quad C \in [0, \pi], \quad Hc = \pi/6.$$

4 Скласти схему алгоритму пошуку MIN значення функції:

$$Y = \frac{A * X^{-C}}{B + K}, \quad C \in [-3, 5], \quad Hc = 0,5.$$

5 Скласти схему алгоритму пошуку MAX значення функції:

$$Y = X * \sin(-B * X), \quad X \in [\pi/12, \pi], \quad Hx = \pi/12.$$

### 2 рівень

6 Визначити значення аргументу, при якому досягається MAX значення функції:

$$Y = \frac{A * X^A + B * X}{C}, \quad X \in [M, K], \quad H_x = H.$$

7 Визначити значення аргументу, при якому досягається MIN значення функції:

$$Y = \frac{A * e^B}{B + A}, \quad A \in [F, Q], \quad H_a = H.$$

8 Визначити значення аргументу, при якому досягається MIN значення функції:

$$Y = A * X^{-C * X}, \quad X = \frac{A + C}{R}, \quad A \in [K, L], \quad H_a = 1.$$

9 Визначити значення аргументу, при якому досягається MAX значення функції:

$$Y = \sqrt{|B|} * X^{-D * C}, \quad C = 2 * \sqrt{|B|}, \quad B \in [M, K], \quad H_b = H$$

10 Визначити різницю між MIN та MAX значеннями функції:

$$Y = \frac{A * X + C * X^D}{2 + \sin(B * X)}, \quad X \in [M, K], \quad H_x = H.$$

3 рівень

11 Визначити, чи є 4 значення функції MAX:

$$Y = B * E^{-\sin(F)}, \quad F \in [0; 0,1], \quad H_f = 0,01.$$

12 Визначити найбільше від'ємне значення функції:

$$Y = \frac{\cos(A * X)}{K}, \quad X \in [R, Q], \quad H_x = H.$$

13 Визначити, яким за рахунком є MAX значення функції:

$$Y = \frac{\sqrt{|F * K|}}{X^A}, \quad B \in [R, F], \quad Hb = H.$$

14 Визначити найменше додатне значення функції

$$Y = \text{LOG}(1 + e^{C+Z}), \quad C \in [2, 10], \quad Hc = 1.$$

15 Визначити, яким за рахунком є найбільше від'ємне значення функції:

$$Y = (X + C) * \text{SIN}(B * X), \quad X \in [-\pi/3, \pi/2], \quad H = \pi/12.$$

## РОБОТА 6

### ПРОЕКТУВАННЯ АЛГОРИТМІВ ПОШУКУ І ВИБІРКИ ЕЛЕМЕНТІВ МАСИВУ

**Мета роботи** – вивчення методики проектування алгоритмів пошуку і вибірки елементів масиву в пам'яті з довільним доступом.

#### **Завдання та порядок виконання**

1 Вивчити навчальний матеріал і підготувати відповіді на контрольні запитання.

2 Розробити схему алгоритму пошуку елементів із заданою ключовою ознакою відповідно до завдань 1-15 розділу 6.

### **Контрольні запитання**

- 1 Визначити поняття "пам'ять з довільним доступом".
- 2 Визначити поняття "масив даних".
- 3 Визначити поняття "розмір, розмірність масиву".
- 4 Перелічити типи алгоритмів обробки масивів.
- 5 Перелічити типи алгоритмів пошуку і вибірки.
- 6 Сутність алгоритму послідовного перебору.

### **Зміст звіту**

- 1 Номер роботи, назва, визначення мети.
- 2 Короткі відповіді на контрольні запитання.
- 3 Схеми алгоритмів пошуку, вибірки і короткий їх опис.
- 4 Висновки з роботи.

### **Навчальний матеріал**

1 *Масив* – упорядкована послідовність однорідних елементів, що мають ім'я. Позиція елемента в масиві може визначатися декількома вимірами і вказується списком індексів, що записуються з ім'ям масиву в дужках.

*Індекс* – константа, змінна чи вираз. Індеси в списку розділяють комами.

*Елементом масиву* називають змінну з індексом. Значення змінній з індексом привласнюють за такими ж правилами, що й простим змінним.

Кількість індексів – *розмірність масиву*.

Кількість елементів масиву – *розмір масиву*.

$A_i, Z_9, S_{j+2}, S_{k-3}$  – елементи одновимірних масивів;

$A_{ij}, Z_{9j}, S_{2^*i,j+2}$  – елементи двовимірних масивів.

В алгоритмах обробки масивів присутні оператори обробки елементів масивів і величин відповідних індексів.

2 Процеси обробки масивів даних доцільно подавати типовими алгоритмами. З урахуванням спільності й повторюваності виділимо і розглянемо чотири типи алгоритмів:

- алгоритми керування ;

- алгоритми переміщення;
- алгоритми зміни значення величини;
- алгоритми структурного перетворення.

*Алгоритми керування* визначають послідовність виконання команд програми. Залежно від значення спеціальних ознак під дією команд відбувається розгалуження, повторення процедур обробки, звертання до підпрограм і т.д.

У результаті виконання *алгоритмів переміщення* відбувається введення, виведення, зміна місця розміщення масивів, їхнє дублювання і т.д.

*Алгоритми зміни значення* величини формують масиви елементів, що є результатом виконання арифметичних і логічних операцій. Операндами при виконанні обчислень виступають елементи, що належать одному чи декільком масивам.

У результаті виконання *алгоритмів структурного перетворення* може змінитися відносно розташування елементів усередині масиву, розмір і склад масивів. Значення величин елементів залишаються без зміни. Алгоритми такого типу подані алгоритмами пошуку, вибірки і сортування.

3 *Алгоритмами пошуку і вибірки* реалізують процедури вибірки одиночних елементів чи елементів групи, формування масивів елементів, значення яких задовольняють визначені умови (ключові ознаки), скорочення масивів, вибірки екстремальних величин. Умови вибірки задаються виразами арифметичного чи логічного типу.

*Ключова ознака* – одна чи декілька характеристик, що відрізняють один елемент від іншого (порядковий номер елемента в масиві, значення самого елемента і т.д.).

4 Найпростіше реалізувати пошук елемента в масиві, якщо ключова ознака збігається з номером місця. Тоді пошук елемента в пам'яті з довільним доступом буде зведений до простого задання адреси елемента.

Якщо елементи в пам'яті розташовані довільно і ключова ознака не є адресою, то для пошуку елемента

можна застосувати алгоритм прямого перебирання. Для цього послідовним переглядом всіх елементів масиву перевіряють відповідність заданого значення ключової ознаки і ознаки  $X_i$  елемента  $i$ .

Якщо відповідність буде встановлено, то  $i$ -й елемент буде знайденим.

### Приклад 1

Масив  $X$  складається з  $N$  числових констант  $X_i$ . Розробити схему алгоритму формування масиву  $B$ , що складається з позитивних чисел масиву  $X$ . Ключова ознака – числове значення елемента  $i$ .

В алгоритмі (рисунок 18) перевірку відповідності ознак виконує блок 7.

Блок 8 формує нову послідовність елементів, що утворюють масив  $B$  з елементами  $B_j$ . Кінцеве значення змінної  $j$  відповідає розмірності масиву  $B$ . У блоках 5, 6, 8 обчислюються значення індексів.

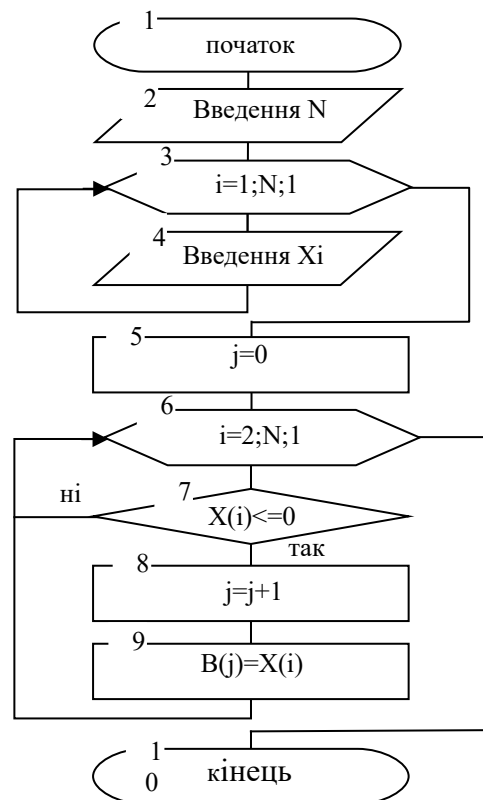
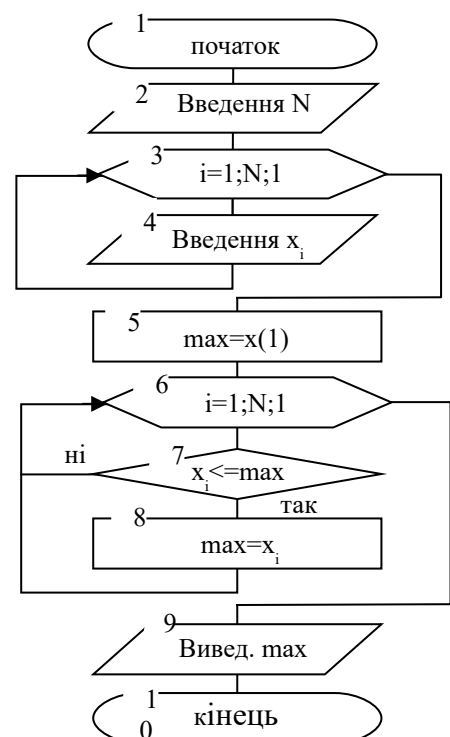


Рисунок 18

### Приклад 2

Масив  $X$  складається з  $N$  числових констант  $X_i$ . Розробити схему алгоритму пошуку елемента  $X_i$  з найбільшим (MAX) значенням ключової ознаки.

В алгоритмі (рисунок 19) при пошуку елемента з максимальним значенням ознаки кожного елемента масиву порівнюються із змінною MAX, якій у блоці 5 присвоєне значення ознаки будь-





якого елемента (у прикладі – елемента 1). Блок 8 значення ознаки змінної MAX замінить, якщо ознака елемента  $X(i) > MAX$ .

Після перегляду всіх елементів масиву значення змінної MAX буде відповідати найбільшому значенню  $X_i$ .

Рисунок 19

## Варіанти індивідуальних завдань

### 1 рівень

Масив X містить M однорідних елементів. Скласти схему алгоритму визначення номера i першого елемента  $X(i)$  масиву X, ключова ознака якого задовольняє умову завдань 1-5.

1  $X(i) > 0$ .

2  $X(i) < 0$ .

3  $X(i) = 0$ .

4  $X(i) = MIN$  (найменше значення ознаки).

5  $X(i) \geq -0.12$ .

### 2 рівень

Масив X містить M однорідних елементів. Скласти схему алгоритму формування масиву Z з елементами  $Z(j)$ , ключова ознака яких задовольняє умову завдань 6-10. Визначити розмір масиву Z.

6  $Z(j) > 0$ .

7  $Z(j) < 0$ .

8  $Z(j) > 13$ .

9  $Z(j) > -0.12$ .

10  $4 \leq Z(j) \leq 12$ .

### 3 рівень

Масив X містить M однорідних елементів. Скласти схему алгоритму формування масивів Z і S з елементами відповідно  $Z(j)$ ,  $S(k)$ , ключові ознаки яких задовольняють умову завдань 11-15. Обчислити суму елементів масивів Z, S, визначити їхній розмір.

- 11  $Z(j) > 0, S(k) < 0.$       12  $1 \leq Z(j) < 5, 5 \leq S(k) < 10.$   
13  $-3 \leq Z(j) < -1, -1 \leq S(k) \leq 3.$     14  $Z(j) > -5, -9 \leq S(k) < -5.$   
15  $-13 \leq Z(j) < 0, S(k) \geq -13.$

## РОБОТА 7

### ПРОЕКТУВАННЯ АЛГОРИТМІВ ОБРОБКИ ОДНОВИМІРНИХ МАСИВІВ

**Мета роботи** – вивчення методики проектування схем алгоритмів обчислення суми, добутку кінцевої кількості елементів одновимірних масивів, введення значень елементів.

#### Завдання та порядок виконання

- 1 Вивчити навчальний матеріал і підготувати відповіді на контрольні запитання.
- 2 Скласти схему алгоритму відповідно до варіанта завдання.

#### Контрольні запитання

- 1 Опишіть алгоритми введення і виведення елементів масиву.
- 2 Яким чином у схемі алгоритму розрахунку суми кінцевої кількості елементів обчислити їх кількість ?
- 3 Опишіть послідовність розрахунку добутку елементів одновимірного масиву.
- 4 Яким чином визначити середнє арифметичне значення кінцевого числа елементів масиву ?
- 5 Які процедури повинні бути в алгоритмі формування нових масивів ?

#### Зміст звіту

- 1 Номер роботи, назва, визначення мети.
- 2 Короткі відповіді на контрольні запитання.

3 Схеми алгоритмів обробки масиву та їх короткий опис.

4 Висновки з роботи.

### Навчальний матеріал

1 Схеми алгоритмів обчислення суми і добутку кінцевої кількості елементів масиву являють собою циклічні алгоритми.

Параметр циклу - порядковий номер елемента.

Для одержання суми необхідно виділити визначену комірку пам'яті ЕОМ. Схема, що наведена на рисунку 20, дозволяє одержати суму 20 елементів масиву K.

i - порядковий номер елемента масиву.

У блоках 2, 3 організовано введення значень елементів одновимірного масиву з клавіатури в режимі "діалогу".

Користувач на запитання ЕОМ (блок 3) вводить послідовно числа масиву, починаючи з i=1. Режим введення підтримується схемою циклічного алгоритму.

У блоці 6 нагромаджується сума S при послідовному перегляді масиву. Початковий стан S – нуль, що визначається блоком 4. Середнє арифметичне значення SR кінцевої кількості елементів масиву визначається за формулою  $SR = \frac{S}{K}$ , де S – сума елементів, K – кількість елементів.

Блок 7 обчислює значення SR після визначення суми 20 елементів масиву.

Схема, що наведена на рисунку 21, дозволяє одержати добуток елементів масиву K.

Початковий стан P – одиниця, що визначається блоком 4. Середнє геометричне значення PR кінцевої кількості елементів масиву визначається за формулою  $PR = \sqrt[20]{P \cdot K_1 \cdot K_2 \cdot \dots \cdot K_{20}}$ , де P – початковий стан, K – кількість елементів масиву.

Блок 7 обчислює значення PR після визначення добутку 20 елементів масиву.

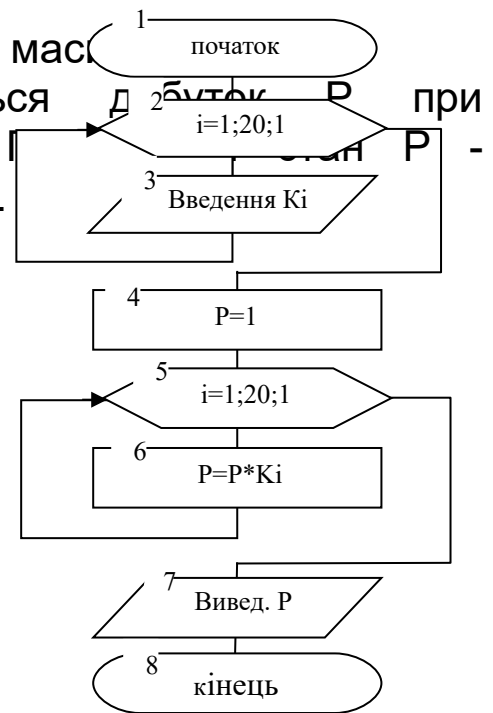
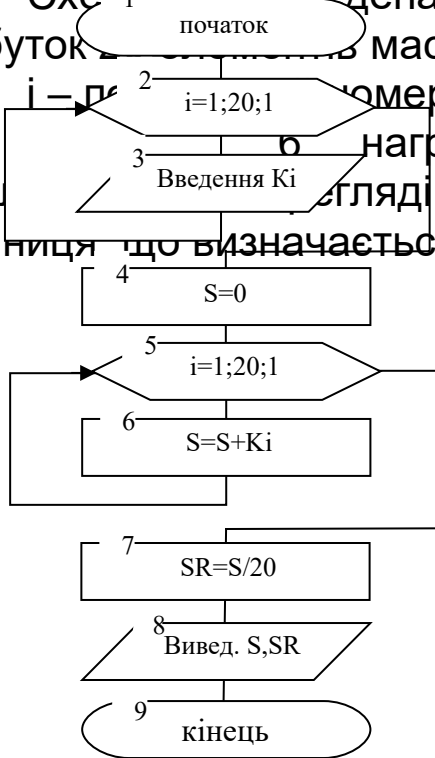


Рисунок 20

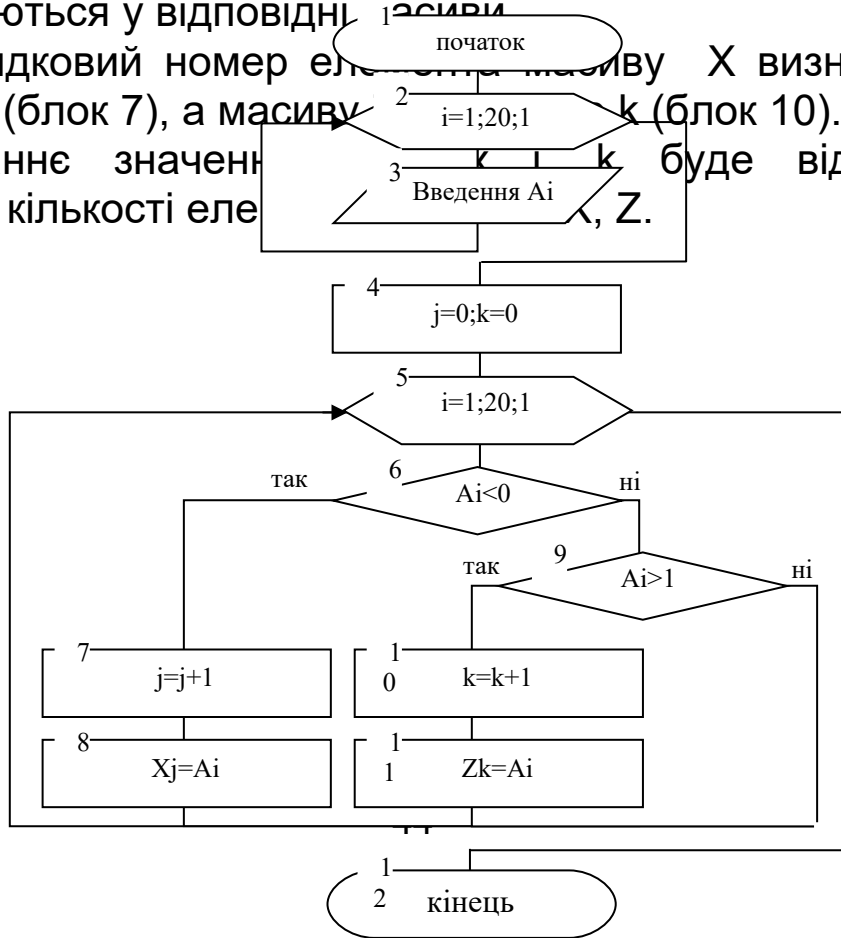
Рисунок 21

2 У схеми алгоритмів формування нових масивів необхідно вводити процедури визначення номера елемента масиву, що формується. Схема алгоритму, наведена на рисунку 22, дозволяє з елементів масиву А створити два масиви Х, Z. Масив Х складається з негативних елементів масиву А, масив Z – з елементів із ключовою ознакою  $A_i > 12$ .

Послідовним переглядом елементів масиву А знаходяться елементи, що задовольняють умови пошуку (блоки 6, 9). Блоками 8, 11 елементи масиву А переписуються у відповідні масиви

Порядковий номер елементів масиву Х визначається змінною j (блок 7), а масиву Z (блок 10).

Останнє значення k буде відповідати загальній кількості елементів масиву Z.



## Рисунок 22

### Варіанти індивідуальних завдань

Скласти схему алгоритму обробки одновимірного масиву відповідно до варіанта завдання.

#### 1 рівень

Масив  $M$  містить  $K$  елементів  $M_i$ ,  $i = 1, K$ .

1 Визначити кількість негативних елементів і обчислити їхнє середнє арифметичне значення.

2 Обчислити суму елементів з непарними порядковими номерами.

3 Обчислити добуток елементів з парними порядковими номерами.

4 Обчислити середнє арифметичне елементів  $M_i > 1$  і  $M_i < 15$ .

5 Обчислити загальну кількість негативних елементів і  $M_i > 20$ .

#### 2 рівень

Масив  $A$  містить 30 елементів  $A_i$ ,  $i = 1, 30$ . Скласти схему алгоритму формування масиву  $T$  з елементами  $T_i$ .

$$6 \quad T_i = A_i + \frac{C}{D}, \quad \text{де } C = \sum_{i=1}^{30} A_i, \quad D = \prod_{i=1}^{10} A_i.$$

Визначити середнє арифметичне позитивних елементів масиву.

$$7 \quad T_i = \begin{cases} \sum_{l=1}^{30} A_l + A_i / 2, & \text{якщо } A_i \leq 0; \\ \prod_{l=1}^{15} A_l - 2A_i, & \text{якщо } A_i \geq 0; \\ 0 & \text{в інших випадках.} \end{cases}$$

Визначити кількість нульових елементів у масиві А.

$$8 \quad T_i = \begin{cases} A_i + \sum_{l=1}^5 A_l, & \text{якщо } 0 \leq A_i \leq 5; \\ 2A_i + \prod_{l=1}^{15} A_l, & \text{якщо } 5 < A_i < 10; \\ 0 & \text{в інших випадках.} \end{cases}$$

Визначити суму негативних елементів з парними індексами.

$$9 \quad T_i = \begin{cases} A_i + \sum_{l=12}^{30} A_l, & \text{якщо } 0 \leq A_i < 10; \\ 2 + \prod_{l=15}^{25} A_l, & \text{якщо } 10 < A_i < 15; \\ 0 & \text{в інших випадках.} \end{cases}$$

Визначити добуток всіх елементів  $A_i > 50$ .

$$10 \quad T_i = \begin{cases} A_i - \sum_{l=1}^{26} A_l, & \text{якщо } -5 \leq A_i \leq 5 \quad \text{і} \quad A_i = 10; \\ A_i + \prod_{l=7}^{12} A_l, & \text{якщо } 10 < A_i < 15 \quad \text{або} \quad A_i = 20; \\ 0 & \text{в інших випадках.} \end{cases}$$

Визначити середнє арифметичне елементів з парними індексами.

### 3 рівень

Дано масиви К і М з елементами  $K_i$  і  $M_i$  відповідно,  $i=1,20$ .

Скласти схему алгоритму формування масиву Т з елементами  $T_i$ .

$$11 \quad T_i = K_i + \frac{\sum_{i=1}^{15} K_i + \sum_{i=4}^{12} M_i}{\prod_{i=3}^{11} K_i + \prod_{i=2}^{10} M_i}.$$

Визначити різницю між добутком і середнім арифметичним позитивних елементів.

$$12 \quad T_i = \frac{K_i}{M_i} + \frac{\sum_{i=5}^{12} K_i + \prod_{i=1}^8 M_i}{\sum_{i=8}^{16} (K_i * M_i)}.$$

Визначити значення першого максимального елемента і його порядковий номер.

$$13 \quad T_i = \begin{cases} K_i + \sum_{i=1}^{10} M_i, & \text{якщо } 0 \leq K_i \leq 5 \text{ і } M_i \leq 0; \\ K_i + \prod_{i=1}^7 M_i, & \text{якщо } K_i \geq 0 \text{ і } M_i \geq 0; \\ 0 & \text{в інших випадках.} \end{cases}.$$

Визначити кількість і суму всіх елементів  $K_i > 0$ .

$$14 \quad T_i = \begin{cases} M_i + \sum_{i=1}^{26} K_i, & \text{якщо } 0 < M_i \leq 10; \\ M_i - \prod_{i=1}^{27} M_i, & \text{якщо } 10 < M_i < 15 \text{ або } K_i = 20; \\ 0 & \text{в інших випадках.} \end{cases}$$

Визначити різницю між максимальним і мінімальним елементами.

$$15 \quad T_i = \begin{cases} M_i + \sum_{i=2}^{23} (K_i * M_i), & \text{якщо } -5 \leq K_i \leq 5 \text{ або } M_i > 0; \\ M(i) + \prod_{i=3}^{12} (K_i + M_i), & \text{якщо } 5 \leq K_i \leq 10 \text{ або } M_i < 0; \\ 0 & \text{в інших випадках.} \end{cases}$$

Визначити різницю між номерами максимального і мінімального елементів.

## РОБОТА 8

### ПРОЕКТУВАННЯ АЛГОРИТМІВ ОБРОБКИ ДВОВИМІРНИХ МАСИВІВ

**Мета роботи** – вивчення методики проектування схем алгоритмів обчислення координат положення елементів, суми і добутку кінцевої кількості елементів двовимірних масивів.

#### Завдання та порядок виконання

1 Вивчити навчальний матеріал і підготувати відповіді на контрольні запитання.

2 Скласти схему алгоритму відповідно до варіанта завдання.

### **Контрольні запитання**

1 Визначити розмір і розмірність масиву  $A$  з елементами  $A_{ij}$ , якщо  $i=1,15$ ;  $j=1,20$ .

2 Етапи визначення добутку елементів у непарних рядках двовимірного масиву.

3 Етапи визначення суми елементів за парними стовпцями двовимірного масиву.

4 Етапи визначення добутку елементів по кожному рядку двовимірного масиву.

5 Етапи визначення суми елементів по кожному стовпцю двовимірного масиву.

### **Зміст звіту**

1 Номер роботи, її назва, визначення мети

2 Короткі відповіді на контрольні запитання.

3 Схеми алгоритмів обробки масиву (згідно з варіантами завдань) і їхній опис.

4 Висновки з роботи.

### **Навчальний матеріал**

1 Масив, що має два індекси (рядків і стовпців), називають двовимірним. Нехай є масив  $B$  з елементами  $B_{ij}$ ,  $i=1,10$ ;  $j=1,15$ . Розмірність цього масиву дорівнює двом, тому що він має два індекси:  $i$  – індекс рядків,  $j$  – індекс стовпців. Розмір масиву  $B$  дорівнює 150 (добуток кількості рядків на кількість стовпців).

Робота з масивом складається з трьох етапів:

1 Введення даних у масив.

2 Обробка даних відповідно до умови конкретної задачі.



### 3 Виведення результатів чи зміненого масиву.

Основні операції над масивом визначені на рівні елементів. Операції над елементами масивів визначаються типом елементів. Для двовимірних масивів процедури введення-обробки-виведення елементів будуються на основі вкладених циклічних обчислювальних алгоритмів.

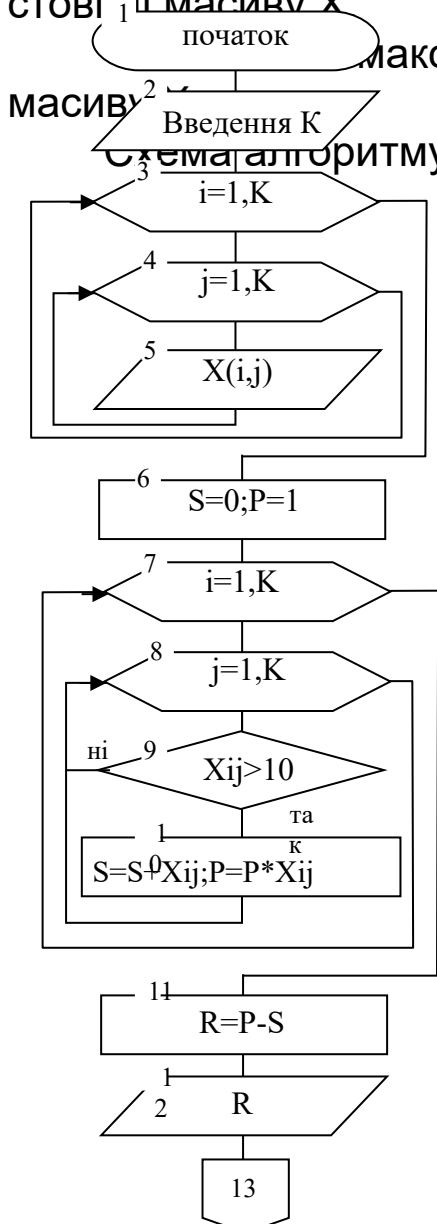
2 Принципи побудови алгоритмів обробки двовимірних масивів розберемо на прикладі розв'язання задачі.

Задано двовимірний масив  $X$  числових елементів ( $K$  рядків і  $K$  стовпців).

Виконати такі дії з їх обробки:

- визначити різницю між добутком і сумою елементів  $> 10$  у масиві  $X$ ;
- визначити середнє арифметичне значення елементів кожного рядка і записати їх в одновимірний масив  $Y$ ;
- визначити кількість негативних значень у кожному стовбі  $1$  і масиву  $X$ .

Схема алгоритму приведена на рисунку 23.



Введення  $K$  кількості рядків та стовпців масиву  $X$

Організація циклу за рядками масиву  $X$

Організація циклу за стовпцями масиву  $X$

Введення елемента масиву  $X$

Початкове значення суми та добутку

Організація циклу за рядками масиву **X**

Організація циклу за стовпцями масиву **X**

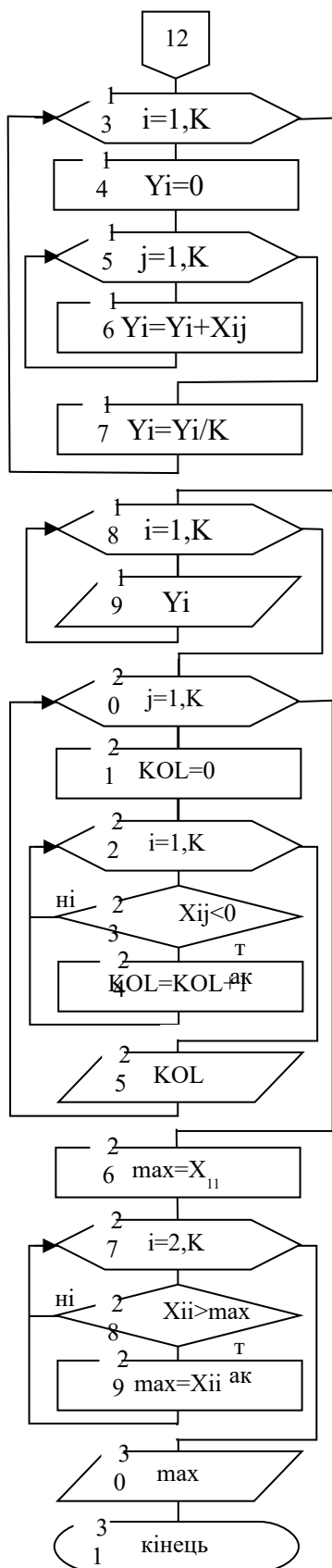
Якщо **X(i,j)** більше 10,

то на цю величину збільшується **S** та  
домножується **P**

Обчислюється різниця між добутком та  
сумою

Друкування різниці **R**

Рисунок 23 – Схема алгоритму (початок)



Організація циклу за рядками масиву **X**

Обнулення елемента масиву **Y**

Організація циклу за стовпцями масиву **X**

Накопичення елементів рядка масиву **X**

Визначення **Y(i)** як середнього арифметичного

Друкування масиву **Y**

Організація циклу за стовпцями масиву **X**

Початкове значення кількості

Організація циклу за рядками стовпчика

Якщо значення елемента негативне, то визначення кількості негативних елементів

Друкування кількості негативних значень у кожному стовпці масиву **X**

Початкове значення **max** дорівнює значенню першого елемента діагоналі

Організація циклу по діагоналі масиву **X**

якщо черговий елемент більше за **max**, значення **max** змінюється на **X(i,i)**

Друкування найбільшого з елементів діагоналі

Рисунок 23 – Схема алгоритму (закінчення)

Розглянувши приведену схему алгоритму, можна виділити основні закономірності побудови алгоритмів обробки двовимірних масивів.

1 Для визначення будь-якої величини У РЯДКАХ масиву необхідно першим відкрити модифікатор РЯДКІВ.

2 Для визначення будь-якої величини У СТОВПЦЯХ масиву необхідно першим відкрити модифікатор СТОВПЦІВ.

3 Для визначення будь-якої величини У КОЖНОМУ РЯДКУ масиву необхідно :

- а) відкрити модифікатор рядків;
- б) задати початкове значення шуканої величини;
- в) відкрити модифікатор стовпців.

4 Для знаходження якої-небудь величини У КОЖНОМУ СТОВПЦІ масиву необхідно :

- а) відкрити модифікатор стовпців;
- б) задати початкове значення шуканої величини;
- в) відкрити модифікатор рядків.

Виведення шуканої величини для 3-го і 4-го пунктів здійснюється між закриттям модифікатора рядків і закриттям модифікатора стовпців.

## **Варіанти індивідуальних завдань**

### 1 рівень

Дано масив А з елементами  $A_{ij}$ ,  $i=1,5$ ;  $j=1,10$ . Скласти схему алгоритму визначення:

- 1) середнього арифметичного позитивних елементів;
- 2) добутку негативних елементів;
- 3) кількості елементів, що  $>1$  і  $<15$ , тобто  $1 < A_{ij} < 15$ ;
- 4) різниці між добутком всіх елементів і сумою негативних елементів;
- 5) середнього арифметичного негативних елементів масиву.

### 2 рівень

Дано масив  $A$  з елементами  $A_{ij}$ ,  $i=1,10$ ;  $j=1,15$ . Скласти схему алгоритму визначення:

- 6) суми негативних елементів у парних рядках масиву;
- 7) добутку позитивних елементів у непарних стовпцях масиву;
- 8) різниці між кількістю негативних і позитивних елементів у парних рядках масиву;
- 9) середнього арифметичного негативних елементів у непарних стовпцях;
- 10) різниці між сумою і добутком елементів  $>10$  у парних рядках масиву.

### 3 рівень

Дано масив  $B$  з елементами  $B_{ij}$ ,  $i=1,N$ ,  $j=1,M$ . Скласти схему алгоритму для визначення:

- 11) добутку негативних елементів у кожному непарному рядку масиву;
- 12) суми позитивних елементів у кожному непарному стовпці масиву;
- 13) середнього арифметичного негативних елементів у кожному парному рядку масиву;
- 14) різниці між середніми арифметичними позитивних і негативних елементів у кожному непарному стовпці масиву;
- 15) різниці між сумою елементів  $>10$  у кожному парному рядку і сумою елементів  $<10$  у кожному непарному стовпці масиву.

## РОБОТА 9

### ПРОЕКТУВАННЯ АЛГОРИТМІВ СОРТУВАННЯ

**Мета роботи** – вивчення методики проектування алгоритмів сортування масивів у пам'яті з довільним доступом.

### Завдання та порядок виконання

1 Вивчити навчальний матеріал і підготувати відповіді на контрольні запитання.

2 Скласти схему алгоритму сортування масиву  $X$ , що містить  $M=60$  елементів відповідно до варіанта завдання.

### **Контрольні запитання**

1 Типи алгоритмів обробки масивів.

2 Характеристика алгоритмів структурного перетворення.

3 Визначити поняття "ключова ознака".

4 Сутність алгоритмів внутрішнього і зовнішнього сортування.

5 Сутність алгоритму вставлення.

### **Зміст звіту**

1 Номер роботи, її назва, визначення мети.

2 Короткі відповіді на контрольні запитання.

3 Схеми алгоритмів сортування масиву та їх короткий опис.

4 Висновки з роботи.

### **Навчальний матеріал**

1 Алгоритми сортування змінюють відносно розташування елементів оброблюваного масиву без зміни їхнього значення. Найбільш загальним видом сортування можна вважати упорядкування масиву (розташування елементів у порядку збільшення або зменшення значення ключової ознаки).

Організація сортування залежить від розміру масиву, виду пам'яті, що використовується для його збереження. Умовно масив вважається невеликим, якщо його розмір не перевищує ємності оперативної пам'яті, виділеної для роботи з ним. Оперативна пам'ять допускає звертання до будь-якого елемента масиву шляхом безпосереднього задання адреси (*пам'ять з довільним доступом*). Масиви невеликих обсягів сортуються методами внутрішнього сортування.

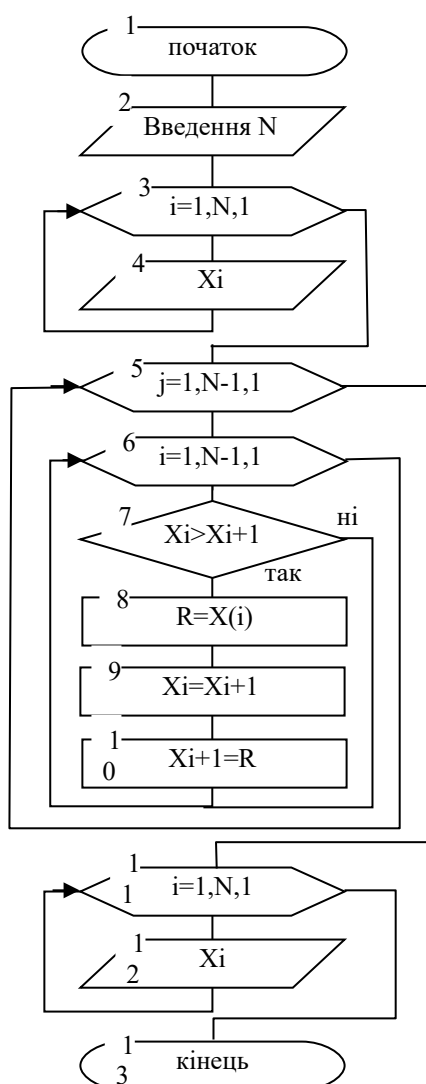
Для збереження великих масивів, розмір яких істотно перевищує ємність оперативної пам'яті, використовується

зовнішня пам'ять (магнітна стрічка, магнітні диски). Такі масиви сортуються методами зовнішнього сортування.

2 *Метод бульбашки* (рисунок 24). Зовнішній цикл: установлюємо кількість проходжень; внутрішній цикл: установлюємо кількість порівнянь. Послідовно порівнюємо  $i$ -й елемент ( $i=1, N$ ) з наступними, починаючи від  $i+1$ -го до кінця. Якщо знайдено менший елемент, його міняють місцями з  $i$ -м. У результаті кожного проходження менші («більш легкі») переміщуються до початку масиву: «спливають вгору, як бульбашки повітря».

### Приклад 1

Елементи масиву  $X$  нумеруються у пам'яті в порядку їхнього розташування числами натурального ряду  $i=1, 2, \dots, N$ . Скласти схему алгоритму сортування масиву в порядку збільшення значення ключової ознаки.



$X_i$  – ключова ознака елемента  $i$  (його числове значення).

Ознака кожного елемента, починаючи з  $i=2$ , послідовно порівнюється з ознакою попередніх елементів  $j=i-1, i-2, \dots, 1$  (блок 5). Значення ознаки елемента  $i$  попередньо переписується у додатковий елемент пам'яті  $R$  (блок 8).

Після перегляду всіх елементів вони будуть розставлені так, що ключова ознака  $X_i$  елемента з меншим номером буде менша ключової ознаки елемента  $i+1$  і більше ключової ознаки елемента  $i-1$ .

Рисунок 24 – Алгоритм вставлення  
**Варіанти індивідуальних завдань**

60 елементів масиву  $X$  розподілені на три групи:

$X(1) - X(20)$  – елементи групи 1;

$X(21) - X(40)$  – елементи групи 2;

$X(41) - X(60)$  – елементи групи 3.

1 рівень

Скласти схему алгоритму сортування 3-х груп елементів масиву  $X$ . Елементи розташувати в порядку відповідно до варіанта завдання.

Ключова ознака  $X(i)$  – значення елемента  $i$ .

1) 1 група – убунання, 2,3 – зростання;

2) 1,3 – убунання, 2 – зростання;

3) 1,2 – убунання, 3 – зростання;

4) 1 – убунання, 2,3 – зростання;

5) 1,2 – убунання, 3 – зростання.

2 рівень

Скласти схему алгоритму сортування 3-х груп елементів масиву  $X$ . Елементи розташувати в порядку відповідно до варіанта завдання. Послідовний перегляд почати з передостанніх елементів груп:  $N_1=19$ ,  $N_2=39$ ,  $N_3=59$ .

6) 1 група – убунання, 2,3 – зростання;

7) 1,3 – убунання, 2 – зростання;

8) 1,2 – убунання, 3 – зростання;

9) 1 – убунання, 2,3 – зростання;

10) 1,2 – убунання, 3 – зростання.

3 рівень

Скласти схему алгоритму сортування масиву  $X$  в умовах відповідно до завдань 1 - 5 і формування масиву  $B$ . Елементи масиву  $B$ :

11) сума відповідних пар елементів груп 2,3;

12) сума відповідних пар елементів груп 1,3;



- 13) сума відповідних пар елементів груп 1,2;
- 14) добуток відповідних пар елементів груп 2,3;
- 15) добуток відповідних пар елементів груп 1,2.