



**Міністерство освіти і науки, молоді та спорту України  
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«НАЦІОНАЛЬНИЙ ГІРНИЧИЙ УНІВЕРСИТЕТ»**



**МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ТА ІНДИВІДУАЛЬНІ  
ЗАВДАННЯ  
ДО ЛАБОРАТОРНИХ РОБІТ  
З ДИСЦИПЛІНИ  
„АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ”**

**ЧАСТИНА 2**

**Дніпропетровськ  
2011**



**Міністерство освіти і науки, молоді та спорту України  
ДЕРЖАВНИЙ ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД  
«НАЦІОНАЛЬНИЙ ГІРНИЧИЙ УНІВЕРСИТЕТ»**

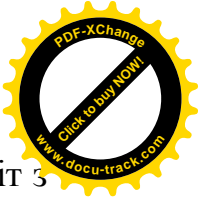
---

**МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ТА ІНДИВІДУАЛЬНІ  
ЗАВДАННЯ  
ДО ЛАБОРАТОРНИХ РОБІТ  
З ДИСЦИПЛІНИ  
„АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ”**

**ЧАСТИНА 2**

**ДЛЯ СТУДЕНТІВ НАПРЯМУ 6.050101**

**Дніпропетровськ  
ДВНЗ «НГУ»  
2011**



Методичні рекомендації та індивідуальні завдання до лабораторних робіт з дисципліни “Алгоритмізація та програмування” частина 2 для студентів напрямку 6.050101/ Упоряд.: М.О. Алексеєв, Л.М. Коротенко, О.С. Шевцова, – Д.: НГУ, 2011.– 79 с.

Упорядники:

М.О. Алексеєв , доктор техн. наук, проф. (л. р. № 1 – 4);

Л.М. Коротенко, канд. техн. наук, доц. (л. р. № 5 – 9);

О.С. Шевцова, асист. (л. р. № 10 – 14).

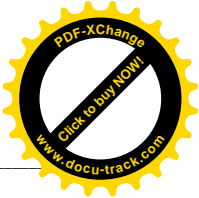
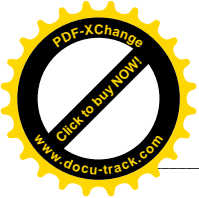
Затверджено методичною комісією з напрямку 050101 Комп’ютерні науки (протокол № 1 від 30.09.11) за поданням кафедри програмного забезпечення комп’ютерних систем (протокол № 2 від 29.09.11).

Відповідальний за випуск завідувач кафедри програмного забезпечення комп’ютерних систем, д.т.н., проф. М.О. Алексеєв.



## ЗМІСТ

Лабораторна робота № 1. Лінійний обчислювальний процес.....	4
Лабораторна робота № 2. Розгалужений обчислювальний процес.....	8
Лабораторна робота № 3. Розгалужений обчислювальний процес. Кускові функції.....	13
Лабораторна робота № 4. Циклічний обчислювальний процес. Табулювання функції.....	17
Лабораторна робота № 5. Циклічний обчислювальний процес. Скінченні суми та добуки.....	20
Лабораторна робота № 6. Циклічний обчислювальний процес. Нескінченні суми.....	24
Лабораторна робота № 7. Одновимірні масиви.....	27
Лабораторна робота № 8. Двовимірні масиви (матриці).....	33
Лабораторна робота № 9. Обробка символічної інформації.....	38
Лабораторна робота № 10. Функції.....	44
Лабораторна робота № 11. Функції з параметрами.....	49
Лабораторна робота № 12. Робота з текстовими файлами.....	55
Лабораторна робота № 13. Структури.....	63
Лабораторна робота № 14. Динамічні масиви.....	70
Додаток 1 .....	76
Додаток 2 .....	77
Список літератури .....	78



# ЛАБОРАТОРНА РОБОТА № 1

## ЛІНІЙНИЙ ОБЧИСЛЮВАЛЬНИЙ ПРОЦЕС

---

### 1. Мета роботи

Дослідити роботу оператора присвоювання мови програмування C++, засвоїти методику складання, налагодження та розрахунку програм лінійних обчислювальних процесів на персональній електронно-обчислювальній машині (ПЕОМ).

### 2. Постановка завдання

Скласти таблицю символічних імен, схему алгоритму та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.

#### Теоретичні відомості

У програмі передбачено використовувати наведені нижче стандартні функції мови програмування C++:

абсолютне значення цілого числа  $abs(X)$ ;

абсолютне значення числа з плаваючою комою  $fabs(X)$ ;

корінь з числа  $sqrt(X)$ ;

експоненту числа  $exp(X)$ ;

синус  $sin(X)$ ;

арксинус  $asin(X)$ ;

косинус  $cos(X)$ ;

арккосинус  $acos(X)$ ;

тангенс  $tan(X)$ ;

арктангенс  $atan(X)$ ;

$X$  у степені  $Y$   $pow(X, Y)$ .

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

– скласти таблицю символічних імен;

– скласти схему алгоритму або псевдокод (за вказівкою викладача);

– скласти програму розв'язання задачі мовою C++.

3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

3.4. Оформлення звіту.

Звіт повинен містити:



- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

#### 4. Приклад

Скласти таблицю символічних імен та програму для обчислення значення

$$\gamma = e^{tg^2 x} + \frac{\sqrt{|t-z|}}{\cos^3 \pi^2 + e^{\pi z^2}} + 2,3 * 10^{1,6},$$

якщо  $p = 3,142$  ;  $t = 2 \cdot 10^3$  ;  $x = 14$  ;  $z = -1,24$  .

#### Розв'язання.

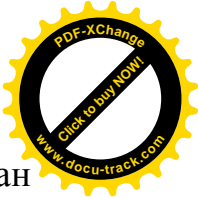
*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
t	t	змінна	double	Уводиться з клавіатури
x	x	змінна	double	Уводиться з клавіатури
z	z	змінна	double	Уводиться з клавіатури
$\pi$	pi	змінна	double	Уводиться з клавіатури
$\gamma$	y	змінна	double	результат

#### Приклад програми мовою C++

для компілятора Visual C++

```
# include <iostream.h> // підключення бібліотеки потоків ввід/вивід
# include <math.h> // підключення бібліотеки математичних дій
int main()
{
    double pi,t,x,z,y; // ідентифікація змінних
    pi=3.142;
    t=2*pow(10,3); // завдання змінним значень з умови
    x=14;
    z=-1.24;
    y=exp(pow(tan(x),2))+sqrt(fabs(t-z))/(
    pow(cos(pow(pi,2)),3)+
    exp(pi)*pow(z,2))+2.3*pow(10,1.6); // підрахунок результату
    cout << "Результат:" << endl; // вивід тексту на екран
```



```

cout <<"y = " << y << endl; // вивід значення змінної y на екран
return 0;
}

```

**5. Варіанти завдань**

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обчислення необхідного значення за формулою відповідного варіанта.

1. Період коливань маятника завдовжки  $l$  обчислюється за формулою

$$t = 2\pi \sqrt{\frac{l}{g}},$$

де  $g$  - прискорення вільного падіння ( $9,81 \text{ м/с}^2$ ). Знайти період

коливань маятника.

2. Сила тяжіння між тілами масою  $m_1$  і  $m_2$ , що перебувають на відстані  $r$  одне

від одного,  $F = \gamma \frac{m_1 m_2}{r^2}$ , де гравітаційна постійна  $\gamma = 6.673 \cdot 10^{-11} \text{ м}^3 / (\text{кг} \cdot \text{с}^2)$ .

Знайти силу тяжіння  $F$ .

3. Периметр  $p$  правильного багатокутника, описаного біля кола радіуса  $r$ ,

$$p = 2nr \text{tg}\left(\frac{\pi}{n}\right),$$

де  $n$  – кількість кутів. Знайти периметр  $p$ .

4. Енергія, що випромінюється чорним тілом на хвилі завдовжки  $\lambda$  при

температурі  $\tau$ ,  $E = \frac{2\pi c \cdot h \lambda^{-5}}{e^{ch/\beta \cdot \lambda \tau} - 1}$ , де  $c = 2,997924 \cdot 10^8$  – швидкість світла;

$h = 6,626 \cdot 10^{-34} \text{ Дж/с}$  – постійна Планка;  $\beta = 1,38 \cdot 10^{-23} \text{ Дж/град}$  - постійна Больцмана. Знайти енергію  $E$ , випромінену чорним тілом.

5. Об'єм циліндра з радіусом основи  $R$  і заввишки  $h$ :  $V = \pi R^2 h$ . Площа його

бічної і повної поверхонь відповідно  $S_{\sigma} = 2\pi R h$ ;  $S_n = 2\pi R h + 2\pi R^2$ .

Знайти  $V$ ,  $S_{\sigma}$  і  $S_n$ .

6. Знайти довжину кола, площу круга і об'єм кулі з радіусом  $R$ . При

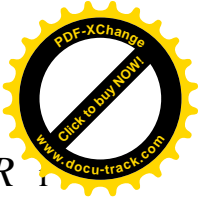
обчисленні використати формули  $l = 2\pi R$ ;  $S = \pi R^2$ ;  $V = \frac{4}{3} \pi \cdot R^3$ .

7. Визначити швидкість різання круглошліфувального верстата за формулою

$$V = d_1 z \pi \frac{d_3}{d_2},$$

де  $d_1$  – діаметр шківа двигуна;  $d_2$  – діаметр робочого вала;

$d_3$  – діаметр різального інструмента;  $z$  – частота обертання.



8. Знайти загальну поверхню та об'єм круглого конуса, що має радіус  $R$  і довжину твірної  $L$ . При обчисленні застосувати формули  $S = \pi R^2 + \pi RL$ ;  $V = \frac{1}{3} \pi R^2 H$ , де  $H$  – висота конуса, яку визначити за формулою  $H = \sqrt{L^2 - R^2}$ .
9. Знайти площу сегмента і сектора. При обчисленні застосувати формули  $S_{сек} = \frac{\pi r^2 \alpha}{360^\circ}$ ;  $S_{сег} = \frac{r^2}{2} \left( \frac{\pi \alpha}{180^\circ} - \sin \alpha \right)$ , де  $r$  – радіус кола,  $\alpha$  – центральний кут, град.
10. Дано гіпотенузу  $c$  і катет прямокутного трикутника  $a$ . Знайти другий катет  $b$  і радіуси  $R$  та  $r$  описаного і вписаного кіл відповідно. При обчисленні застосувати формули  $r = \frac{2S}{a + b + c}$  і  $R = \frac{abc}{4S}$ , де  $S$  – площа трикутника.
11. Знайти відстань між двома точками з координатами  $(x_1, y_1)$  і  $(x_2, y_2)$ . За допомогою формули  $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ .
12. Знайти периметр і площу прямокутного трикутника, якщо відомі довжини двох катетів.
13. Знайти периметр і площу рівнобічного трикутника, якщо відома довжина його сторони.
14. Знайти площу кільця і площу частини кільця з центральним кутом  $\varphi$  (у градусах). При обчисленні застосувати формули  $S = \pi(R^2 - r^2)$ ;  $S = \frac{\varphi \pi}{360^\circ} (R^2 - r^2)$ .
15. Знайти периметр і площу паралелограма зі сторонами  $a$ ,  $b$  і кутом  $\alpha$  між ними. При обчисленні застосувати формулу  $S = ab \sin \alpha$ .
16. Знайти внутрішній кут  $\alpha$  і суму внутрішніх кутів правильного опуклого багатокутника. При обчисленні застосувати формули  $\alpha = \pi \frac{n - 2}{n}$ ;  $S = \pi(n - 2)$ .
17. Знайти об'єм і площу поверхні прямого паралелепіпеда зі сторонами  $a$ ,  $b$  і  $c$ .
18. Знайти середню лінію і площу трапеції, якщо відомі її основи і висота.
19. Дано координати вершин трикутника  $A(x_1, y_1)$ ,  $B(x_2, y_2)$  і  $C(x_3, y_3)$ . Знайти середини його сторін. При обчисленні застосувати формули  $x = \frac{(x_1 + x_2)}{2}$ ;  $y = \frac{(y_1 + y_2)}{2}$ , де  $M(x, y)$  – середина відрізка  $AB$ , заданого точками  $A(x_1, y_1)$ ,  $B(x_2, y_2)$ .





20. Відомі координати вершин трикутника  $A(x_1, y_1)$ ,  $B(x_2, y_2)$  і  $C(x_3, y_3)$ . Знайти його периметр. При обчисленні застосувати формулу відстані

між двома точками:  $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ .

21. На площині задані пряма  $Ax + By + c = 0$  і точка  $M$  з координатами  $(x_1, y_1)$ . За формулою:  $d = \frac{|Ax_1 + By_1 + C|}{\sqrt{A^2 + B^2}}$  знайти відстань  $d$  від точки до прямої.

22. Відомі вектори  $\vec{a}(x_1, y_1)$  та  $\vec{b}(x_2, y_2)$  і кут  $\varphi$  між ними, град. Знайти скалярний добуток векторів за формулою  $(\vec{a}, \vec{b}) = |\vec{a}| \cdot |\vec{b}| \cos \varphi$ , якщо

$$|\vec{a}| = \sqrt{x_1^2 + y_1^2}.$$

23. Відомі два вектори  $\vec{a}(x_1, y_1)$  і  $\vec{b}(x_2, y_2)$ . Знайти кут  $\varphi$  між ними.

При обчисленні застосувати формули  $(\vec{a}, \vec{b}) = |\vec{a}| \cdot |\vec{b}| \cos \varphi$ ;

$$(\vec{a}, \vec{b}) = x_1x_2 + y_1y_2; |\vec{a}| = \sqrt{x_1^2 + y_1^2}.$$

24. На площині задані прямі  $y = k_1x + b_1$  і  $y = k_2x + b_2$ . Знайти кут  $\varphi$  між ними.

При обчисленні застосувати формулу  $tg \varphi = \frac{k_2 - k_1}{1 + k_1k_2}$ .

25. Знайти кути трикутника, сторони якого задані рівняннями прямих ліній:  $y = k_1x + b_1$ ;  $y = k_2x + b_2$  і  $y = k_3x + b_3$ . При обчисленні застосувати

формулу  $tg \varphi = \frac{k_2 - k_1}{1 + k_1k_2}$ , де  $k_1$  і  $k_2$  – коефіцієнти прямих, заданих рівняннями  $y = k_1x + b_1$ ;  $y = k_2x + b_2$ , а  $\varphi$  – кут між ними.

## ЛАБОРАТОРНА РОБОТА №2 РОЗГАЛУЖЕНИЙ ОБЧИСЛЮВАЛЬНИЙ ПРОЦЕС

### 1. Мета роботи

Дослідити роботу оператора умови мови програмування C++, засвоїти методику складання, налагодження та розрахунку програм розгалужених обчислювальних процесів на ПЕОМ.



## 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою С++ відповідно до заданого варіанта. Налагодити та розв’язати задачу на ПЕ-ОМ.

### Теоретичні відомості

Для організації умовних та безумовних переходів у програмі мовою С++ використовуються оператори `if - else`, `switch` та `goto`. Під час розв’язання задач даної лабораторної роботи доречно застосувати оператор умови `if - else` з таким синтаксисом:

```
if (перевірка_умови) оператор_1; else оператор_2;
```

Якщо умова у дужках набуває значення „True”, виконується оператор\_1, а якщо „False” – оператор\_2. Коли ж замість одного необхідно виконати кілька операторів, то їх об’єднують у фігурні дужки. В операторі `if` службове слово `else` може бути відсутнім, якщо немає потреби в іншій гілці умови.

В операторі `if - else` безпосередньо після ключових слів `if` та `else` повинні виконуватися інші оператори. Якщо хоча б один з них – оператор `if`, його називають укладеним.

У блоці „перевірка умови” можуть бути використані складні логічні умови, які реалізуються за допомогою операцій булевої алгебри:

Операція	Призначення	Приклад
!	Логічне НІ	! в
<	Менше	а < в
<=	Менше або =	а <= в
>	Більше	а > в
>=	Більше або =	а >= в
==	Дорівнює	а == в
!=	Не дорівнює	а != в
&&	Логічне І	а && в
	Логічне АБО	а    в

## 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою С++.



### 3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

### 3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

## 4. Приклад

Скласти таблицю символічних імен та програму для розв'язання задачі.

Дано три числа:  $a, b, c$ . з'ясувати правильність твердження  $a > b$  та  $a < c$ .

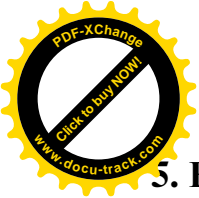
### Розв'язання.

*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
a	a	Змінна для порівняння	int	Уводиться з клавіатури
b	b	Змінна для порівняння	int	Уводиться з клавіатури
c	c	Змінна для порівняння	int	Уводиться з клавіатури

### Текст програми мовою C++ для компіляторів Borland C

```
#include <iostream.h>
int main()
{
    int a, b, c;
    cout << "Уведіть a, b, c: " << endl;
    cin >> a >> b >> c; // уведення даних
    if ((a > b) && (a < c)) // якщо умова виконається,
        cout << "a>b і a<c - Правильно!" << endl; // на екрані
        // відобразиться «a>b і a<c - Правильно!»
    else // якщо умова не виконається,
        cout << "Неправильно a>b і a<c" << endl; // на екрані
        // відобразиться «Неправильно a>b і a<c»
    return 0;
}
```

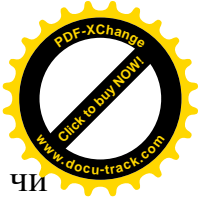


## 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обчислення необхідного значення за формулою відповідного варіанта:

1. Відомі два числа. Замінити друге число нулем, якщо воно більше першого, і залишити незмінним, якщо це не так.
2. Знайти найменше з трьох відомих чисел.
3. Знайти найбільше з трьох відомих чисел.
4. Відомі три числа. Від'ємне з них число залишити без зміни, а інші – піднести до квадрата.
5. Відомі три числа  $a$ ,  $b$  і  $c$ . З'ясувати правильність умови  $a < b < c$ . Відповідь вивести на екран у текстовій формі: «Правильно» або «Неправильно».
6. Відомі два числа  $x$  і  $y$  ( $x \neq y$ ), менше з яких замінити їх півсумою, а більше – їхнім подвоєним добутком.
7. Відомі три числа  $a$ ,  $b$  і  $c$ . Якщо виконується умова  $a > b > c$ , подвоїти кожне з цих чисел, а в протилежному разі замінити їх значення модулями.
8. Відомі два числа  $x$  і  $y$ . Якщо  $x$  і  $y$  – від'ємні, то кожне число замінити його модулем; якщо від'ємне одне з них, кожне число збільшити на 0,5; якщо  $x$  і  $y$  – невід'ємні, кожне число збільшити в 10 разів.
9. Дано позитивні числа  $x$ ,  $y$ ,  $z$ . З'ясувати, чи існує трикутник із довжинами сторін  $x$ ,  $y$ ,  $z$ . Відповідь вивести на екран у вигляді тексту.
10. Визначити, чи матиме квадратне рівняння  $ax^2 + bx + c = 0$  хоча б один дійсний розв'язок. Відповідь вивести на екран як текст.
11. Знайти найбільше і найменше з двох відомих чисел.
12. Продаж книг у книгарні здійснюється за допомогою ЕОМ. Скласти програму, що запитує вартість книг; суму грошей, внесену покупцем; а потім визначає належну здачу (якщо грошей внесено більше); друкує «Спасибі за покупку!», якщо здачі не потрібно, або видає повідомлення про нестачу внесеної суми.
13. Відомі числа  $x$  і  $y$ . Поміняти їх місцями так, щоб виконувалося співвідношення  $x \geq y$ .
14. Відомі координати трьох точок  $A(x_1, y_1)$ ,  $B(x_2, y_2)$  і  $C(x_3, y_3)$ . З'ясувати, чи лежать ці точки на одній прямій. Відповідь вивести на екран як текст.
15. Відомі координати точки  $A(x, y)$ . З'ясувати, чи належить точка  $A$  кільцю з центром на початку координат із радіусами: зовнішнім  $R$  і внутрішнім  $r$  ( $R > r$ ).
16. Відомі координати точки  $A(x_1, y_1)$ . З'ясувати, чи належить точка  $A$  колу одиничного радіуса з центром у точці  $B(x_2, y_2)$ .



17. На площині задані пряма  $y = kx + b$  і точка  $A(x, y)$ . З'ясувати, чи належить точка  $A$  цій прямій.
18. На площині задані прямі  $y = k_1x + b_1$  і  $y = k_2x + b_2$ . Визначити взаємне їх розташування на площині. Указівки: умова паралельності двох прямих  $k_1 = k_2$ ; умова перпендикулярності двох прямих  $1 + k_1k_2 = 0$ .
19. Відомі координати вершин трикутника  $A(x_1, y_1)$ ,  $B(x_2, y_2)$  і  $C(x_3, y_3)$ . Визначити, чи трикутник рівнобедрений. При обчисленні застосувати формулу відстані між двома точками  $A(x_1, y_1)$  і  $B(x_2, y_2)$ :  
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$
20. Відомі координати вершин трикутника  $A(x_1, y_1)$ ,  $B(x_2, y_2)$  і  $C(x_3, y_3)$ . Визначити, чи трикутник рівнобічний. При обчисленні застосувати формулу відстані між двома точками  $A(x_1, y_1)$  і  $B(x_2, y_2)$ :  
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$
21. Сторони трикутника задані рівняннями прямих  $y = k_1x + b_1$ ,  $y = k_2x + b_2$  і  $y = k_3x + b_3$ . Визначити, чи трикутник прямокутний. При обчисленні застосувати умову перпендикулярності прямих  $1 + k_1k_2 = 0$ , де  $k_1$  і  $k_2$  – коефіцієнти прямих, заданих рівняннями  $y = k_1x + b_1$ ;  $y = k_2x + b_2$ .
22. На площині задані коло радіуса  $R$  із центром у точці  $A(a, b)$  і точка  $M(x, y)$ . З'ясувати, не лежить точка  $M$  цьому колу, колу чи лежить поза колом.
23. Відомі координати вершин чотирикутника  $A(x_1, y_1)$ ,  $B(x_2, y_2)$ ,  $C(x_3, y_3)$  і  $D(x_4, y_4)$ . Визначити, чи цей чотирикутник – ромб. При обчисленні застосувати формулу відстані між двома точками  $A(x_1, y_1)$  і  $B(x_2, y_2)$ :  
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$
24. Відомі координати вершин чотирикутника  $A(x_1, y_1)$ ,  $B(x_2, y_2)$ ,  $C(x_3, y_3)$  і  $D(x_4, y_4)$ . Визначити, чи цей чотирикутник – квадрат. При обчисленні застосувати формулу відстані між двома точками  $A(x_1, y_1)$  і  $B(x_2, y_2)$ :  
$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}.$$
25. Відомі два вектори  $\vec{a}(x_1, y_1)$  і  $\vec{b}(x_2, y_2)$ . Обчислити довжину цих векторів і визначити, у скільки разів вектор  $\vec{a}$  довший або коротший від вектора  $\vec{b}$ .



## ЛАБОРАТОРНА РОБОТА № 3 РОЗГАЛУЖЕНИЙ ОБЧИСЛЮВАЛЬНИЙ ПРОЦЕС. КУСКОВІ ФУНКЦІЇ

---

### 1. Мета роботи

Дослідити дії оператора умови мови програмування C++, засвоїти методику складання, налагодження та розрахунку програм розгалужених обчислювальних процесів на ПЕОМ щодо кускових функцій.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.

#### Теоретичні відомості

У програмі для розв'язання задач визначення кускових функцій можуть бути використані оператори умови `if-else` (див. л.р. № 2).

Кускова функція – це функція, що на різних проміжках аргументу  $x$  задається різними формулами. Залежно від того, якому проміжку належить аргумент функції  $x$ , і має визначатися формула, за якою знаходиться значення функції в заданій точці.

Наприклад:

$$y = \begin{cases} \sin x, & \text{при } x \leq 0; \\ \operatorname{arctg} x, & \text{при } 0 < x \leq 8; \\ \lg x, & \text{при } 8 < x \leq 32; \\ \frac{1}{x}, & \text{в інших випадках.} \end{cases}$$

Якщо задати значення аргументу  $x = -4$ ,  $y$  розраховуватиметься за формулою  $\sin(-4)$ ; якщо  $x = 5$ ,  $y$  набуде значення  $\operatorname{arctg}(5)$ ; якщо  $x$  дорівнюватиме 20,  $y = \lg(20)$ , та коли  $x$  матиме значення більше 32,  $y$  визначатиметься за формулою  $\frac{1}{x}$ .

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв'язання задачі мовою C++.

3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

3.4. Оформлення звіту.



Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

#### 4. Приклад

Написати програму для обчислення значення функції

$$y = \begin{cases} \sin x, & \text{при } x < 0; \\ \frac{2}{x} + 3x, & \text{при } x > 0; \\ x, & \text{при } x = 0. \end{cases}$$

**Розв'язання.**

*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
x	x	Змінна	double	Уводиться з клавіатури
y	y	Змінна	double	Результат

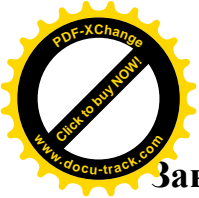
**Текст програми мовою C++**

**для компіляторів Borland C**

```
#include <iostream.h>
#include <math.h>
int main()
{
    double x, y;
    cout << "Задайте x: ";
    cin >> x;
    if (x<0) y = sin(x); // при x < 0 y дорівнює sin(x)
    else
        if (x>0) y = 2/x + 3*x; // при x > 0 y дорівнює 2/x + 3*x
        else y = 2; // в інших випадках y дорівнює 2
    cout << " y = " <<y << endl;
    return 0; }
```

#### 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.



**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обчислення значень функції відповідно до заданого варіанта.

$$1. y = \begin{cases} \sin x, & \text{при } x \leq 0; \\ \arctg x, & \text{при } 0 < x \leq \pi/4; \\ \log_2 x, & \text{при } \pi/4 < x \leq 32; \\ 1/x, & \text{в інших випадках.} \end{cases}$$

$$2. z = \begin{cases} \ln|x|, & \text{при } x < -\pi; \\ \sin x + \cos 2x, & \text{при } -\pi \leq x < \pi; \\ x^3 + 1, & \text{при } \pi \leq x < 10; \\ \frac{x+1}{x^2+8}, & \text{при } 10 \leq x < 100; \\ \ln x, & \text{в інших випадках.} \end{cases}$$

$$3. z = \begin{cases} \arctg \frac{x}{y}, & \text{при } y \neq 0, |x| > |y|; \\ \arcsin \frac{x}{y}, & \text{при } y \neq 0, |x| \leq |y|; \\ 0, & \text{в інших випадках.} \end{cases}$$

$$4. z = \begin{cases} \frac{1}{x} + \frac{1}{y}, & \text{при } x < -10, y < -5; \\ \frac{x-y}{x+y}, & \text{при } -10 \leq x < 0, \\ & -5 \leq y < 0; \\ & \bullet \text{при } 0 \leq x < 2\pi, \\ \frac{\sin x}{\cos y}, & 0 \leq y < \pi/2; \\ \ln(x^2 + y^2), & \text{в інших випадках.} \end{cases}$$

$$5. z = \begin{cases} \sin \frac{x+y}{y}, & \text{при } y \neq 0, |x| > |y|; \\ \arccos \frac{x}{y}, & \text{при } y \neq 0, |x| \leq |y|; \\ \pi, & \text{в інших випадках.} \end{cases}$$

$$6. y = \begin{cases} 0, & \text{при } x \leq 0; \\ 1/x, & \text{при } 0 < x \leq 1; \\ x^2, & \text{при } 1 < x \leq 4; \\ 14 + \log_2 x, & \text{при } x > 4. \end{cases}$$

$$7. z = \begin{cases} \frac{2}{x} - \frac{4}{y}, & \text{при } x < -20, y < -10; \\ \frac{x-y-2}{x+y}, & \text{при } -20 \leq x < 0, \\ & -10 \leq y < 0; \\ \frac{\sin x + \cos x}{\cos y}, & \text{при } 0 \leq x < 2\pi, \\ & 0 \leq y < \pi/2; \\ \log_4(x^2 + y^2), & \text{в інших випадках.} \end{cases}$$

$$8. y = \begin{cases} \cos x, & \text{при } x \leq 0; \\ \arcsin x, & \text{при } 0 < x \leq \pi/2; \\ \log_4 x, & \text{при } \pi/2 < x \leq 64; \\ 1/x^2, & \text{в інших випадках.} \end{cases}$$

$$9. y = \begin{cases} 0, & \text{при } x < 0; \\ 1/(x^2 + 1), & \text{при } 0 \leq x \leq 1; \\ x^3, & \text{при } 1 < x \leq 4; \\ 62 + \log_8 x, & \text{при } x > 4. \end{cases}$$

$$10. z = \begin{cases} x + y, & \text{при } x < 0, y \leq 0; \\ \arctg \frac{x}{y}, & \text{при } 0 \leq x < 10, \\ 0, & \text{в інших випадках.} \end{cases}$$

$$11. z = \begin{cases} \sin x + \sin y, & \text{при } x < 0, y > 0; \\ \cos x + \cos y, & \text{при } x > 0, y < 0; \\ \frac{\sin x + \cos y}{\cos^2 x + \sin^2 y}, & \text{в інших випадках.} \end{cases}$$





$$y = \frac{\text{sign } k \cdot \text{sign } l \cdot \delta_{kl}}{\text{sign}(k+l)\text{sign}(k-l)\delta_{k+l,k-l}},$$

де

12..

$$\text{sign } x = \begin{cases} -1, & \text{при } x < 0; \\ 0, & \text{при } x = 0; \\ 1, & \text{при } x > 0; \end{cases}$$

$$\delta_{nm} = \begin{cases} 1, & \text{при } n = m; \\ 0, & \text{в інших випадках.} \end{cases}$$

$$13. z = \begin{cases} \ln x, & \text{при } x < -\pi; \\ \sin x - \cos x, & \text{при } -\pi \leq x < \pi; \\ x + 1, & \text{при } \pi \leq x < 10; \\ \frac{x+3}{x^2+8}, & \text{при } 10 \leq x < 100; \\ \log_2 x, & \text{в інших випадках.} \end{cases}$$

$$14. z = \begin{cases} \text{ctg } \frac{x}{y}, & \text{при } y \neq 0, |x| > |y|; \\ \arccos n \frac{x}{y}, & \text{при } y \neq 0, |x| \leq |y|; \\ -1, & \text{в інших випадках.} \end{cases}$$

$$15. y = \begin{cases} \cos x, & \text{при } x \leq 0; \\ \text{tg } x, & \text{при } 0 < x \leq \frac{\pi}{4}; \\ \log 10x, & \text{при } \frac{\pi}{4} < x \leq 32; \\ \frac{1}{x^2}, & \text{в інших випадках.} \end{cases}$$

$$16. y = \begin{cases} x^2 + a & a > b, b > 2 \\ \max(a, b) & a < b, b < 2 \\ \sin^3 x^2 + \sqrt[3]{a^b - \text{tg} b} & a = b \\ \min(a, b) & \text{в інших випадках} \end{cases}$$

$$17. z = \begin{cases} \text{ctg } \frac{x}{y}, & \text{при } y \neq 0, |x| > |y|; \\ \arccos n \frac{x}{y}, & \text{при } y \neq 0, |x| \leq |y|; \\ -1, & \text{в інших випадках.} \end{cases}$$

$$18. z = \begin{cases} \frac{1}{x^2} + \frac{1}{y^2}, & \text{при } x < -5, y < -1; \\ \frac{x-y}{x+y}, & \text{при } -5 \leq x < 0, \\ & -1 \leq y < 0; \\ \frac{\cos x}{\sin y}, & \text{при } 0 \leq x < 2\pi, \\ & 0 \leq y < \frac{\pi}{2}; \\ \ln(x+y), & \text{в інших випадках.} \end{cases}$$

$$19. z = \begin{cases} \sin \frac{x+y}{y^2}, & \text{при } y \neq 0, x > y; \\ \arcsin \frac{x}{y}, & \text{при } y \neq 0, x \leq y; \\ \frac{\pi}{2}, & \text{в інших випадках.} \end{cases}$$

$$20. y = \begin{cases} -100, & \text{при } x \leq 0; \\ \frac{1}{x^2}, & \text{при } 0 < x \leq 1; \\ x, & \text{при } 1 < x \leq 4; \\ 100 + \ln x, & \text{при } x > 4. \end{cases}$$

$$21. k = \begin{cases} \frac{2}{x} - \frac{4}{y}, & \text{при } x < -10, y < -5; \\ \frac{x-y-2}{x+y}, & \text{при } -10 \leq x < 0, \\ & -5 \leq y < 0; \\ \frac{\sin x + \cos x}{\sin y}, & \text{при } 0 \leq x < 2\pi, \\ & 0 \leq y < \frac{\pi}{2}; \\ \ln(x^2 + y^2), & \text{в інших випадках.} \end{cases}$$

$$22. w = \begin{cases} \pi, & \text{при } x < 0; \\ \frac{1}{x^3}, & \text{при } 0 \leq x \leq 3; \\ x^2, & \text{при } 3 < x \leq 10; \\ 6,2 + \log_5 x, & \text{при } x > 10. \end{cases}$$



$$23. y = \begin{cases} \sin x, & \text{при } x \leq 0; \\ \arccos x, & \text{при } 0 < x \leq \pi/2; \\ \log_4 x, & \text{при } \pi/2 < x \leq 64; \\ 1/x^2, & \text{в інших випадках.} \end{cases}$$

$$24. v = \begin{cases} x * y, & \text{при } x < 0, y \leq 0; \\ \operatorname{ctg} \frac{x}{y}, & \text{при } 0 \leq x < 10, 0 < y \leq 9; \\ -100, & \text{в інших випадках.} \end{cases}$$

$$25. z = \begin{cases} \arcsin x + \sin y, & \text{при } x < 0, \\ & y > 0; \\ \cos x^2 - \cos y^2, & \text{при } x > 0, \\ & y < 0; \\ \frac{\sin x + \cos y}{\cos^2 x + \sin^2 y}, & \text{в інших випадках.} \end{cases}$$

## ЛАБОРАТОРНА РОБОТА № 4 ЦИКЛІЧНИЙ ОБЧИСЛЮВАЛЬНИЙ ПРОЦЕС. ТАБУЛЮВАННЯ ФУНКЦІЇ

### 1. Мета роботи

Дослідити роботу операторів циклу алгоритмічної мови C++, засвоїти методику складання, налагодження та розрахунку програм циклічних обчислювальних процесів для табулювання функцій на ПЕОМ.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.

#### Теоретичні відомості

Цикли утворюються для того, щоб виконати деякий оператор чи групу операторів кілька разів. В мові C++ існує три оператора циклу: `for`, `while` і `do while`. Перший з них формально записується у вигляді

`for (вираз_1; вираз_2; вираз_3) тіло_циклу.`

Тіло циклу складається з одного або кількох операторів, об'єднаних фігурними дужками `{ ... }` (після блока крапка з комою не ставиться). У виразах 1, 2, 3 фігурує спеціальна змінна, що називається керуючою. За її значенням встановлюється необхідність повторення циклу або виходу з нього.

Вираз\_1 – початкове значення, вираз\_2 – кінцеве значення керуючої змінної, а вираз\_3 – крок, з яким має змінювати своє значення керуюча змінна з подольшою ітерацією.

### 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.



Ознайомитись з рекомендованою літературою та конспектом лекцій.

### 3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

### 3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

### 3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

## 4. Приклад

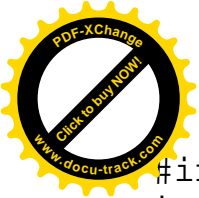
Побудувати таблицю значень для функції  $f(x) = x \sin(x)$  на відрізку  $[0, 3\pi]$  з числом розбиття відрізка  $m=10$ .

### Розв’язання.

*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
f	f	Функція	double	Результат
x	x	Аргумент функції	double	З'ясовується у процесі
a	a	Початкове значення інтервалу	double	Уводиться з клавіатури
b	b	Кінцеве значення інтервалу	double	Уводиться з клавіатури
m	m	Число розбиття	int	Уводиться з клавіатури
step	step	Крок розбиття	double	З'ясовується у процесі
i	i	Керуюча змінна	int	Змінне значення

**Текст програми мовою C++**  
**для компіляторів Borland C**  
`#include <iostream.h>`



```
#include <math.h>
int main()
{
    double f, step, a, b, x;
    int i, m;
    cout<<"Уведіть початкове значення відрізка"<<endl;
    cin>>a;
    cout<<"Уведіть кінцеве значення відрізка"<<endl;
    cin>>b;
    cout<<"Уведіть число розбиття"<<endl;
    cin>>m;

    step =(b - a)/m; // визначаємо, з яким кроком змінюються значення x
    x = a; // задаємо початкове значення x
    for ( i = 0; i <=10; i++ )
    {
        f = x*sin( x ); // присвоюємо значенню у поточне значення x
        cout <<"Значення функції f["<<x<<"]="<<f<<endl;
        x = x + step; // змінюємо поточне значення аргументу на наступне
    }
    return 0;
}
```

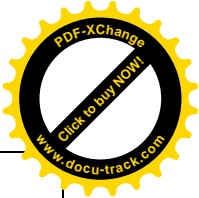
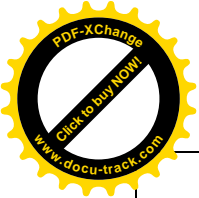
### 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обчислення значень функції відповідно до заданого варіанта.

1. Побудувати таблицю значень для функції  $f(x)$  на відрізку  $[a,b]$  з числом розбивання відрізка  $m$ :

Варіант	Функція	Відрізок	Число розбиття
1	$f(x) = x - \sin(x)$	$[0, \pi/2]$	10
2	$f(x) = \sin(x)$	$[\pi/4, \pi/2]$	15
3	$f(x) = \cos(x)$	$[\pi/3, 2\pi/3]$	20
4	$f(x) = \operatorname{tg}(x)$	$[0, \pi/4]$	10
5	$f(x) = \operatorname{ctg}(x)$	$[\pi/4, \pi/2]$	15
6	$f(x) = \arcsin(x)$	$[0, 1]$	20
7	$f(x) = \arccos(x)$	$[0.5, 1]$	10
8	$f(x) = \operatorname{arctg}(x)$	$[2, 7]$	15
9	$f(x) = \sin(x) - \cos(x)$	$[0, \pi/2]$	20
10	$f(x) = x \cos(x)$	$[0, 3\pi]$	10
11	$f(x) = \sin\left(\frac{1}{x}\right)$	$[\pi/8, 2/\pi]$	10



12	$f(x) = \cos\left(\frac{1}{x}\right)$	$[\pi/4, 4/\pi]$	20
13	$f(x) = \sin(x^2)$	$[\pi/6, 2\pi/3]$	10
14	$f(x) = \cos(x^2)$	$[\pi/3, 3\pi/3]$	15
15	$f(x) = \sin(x) + \operatorname{tg}(x)$	$[0, \pi/4]$	20
16	$f(x) = \cos(x) + \operatorname{ctg}(x)$	$[\pi/4, \pi/2]$	10
17	$f(x) = \operatorname{tg}\left(\frac{x}{2}\right)$	$[0, 2\pi/3]$	15
18	$f(x) = \operatorname{tg}\left(\frac{x}{2}\right) + \cos(x)$	$[\pi/2, \pi]$	20
19	$f(x) = \operatorname{ctg}\left(\frac{x}{3}\right) + \sin(x)$	$[\pi/4, \pi/2]$	10
20	$f(x) = \frac{\sin\left(\frac{x}{4}\right)}{2}$	$[\pi/2, \pi]$	15
21	$f(x) = \cos^2(x) + \ln(x)$	$[1, 10]$	20
22	$f(x) = \operatorname{tg}(x) + \left(\frac{1}{2}\right)^x$	$[0, 1]$	10
23	$f(x) = \ln x + \log_2 x$	$[1, 2]$	12
24	$f(x) = \operatorname{arctg}(x) + 1$	$[10, 20]$	20
25	$f(x) = \cos(x^2) + \sin(x^2)$	$[-\pi/2, \pi/2]$	25

---

**ЛАБОРАТОРНА РОБОТА № 5**  
**ЦИКЛІЧНИЙ ОБЧИСЛЮВАЛЬНИЙ ПРОЦЕС. СКІНЧЕННІ СУМИ**  
**ТА ДОБУТКИ**

---

**1. Мета роботи**

Дослідити роботу операторів циклу алгоритмічної мови C++, засвоїти методику складання, налагодження та розрахунку програм циклічних обчислювальних процесів для обчислення скінченних сум та добуток на ПЕОМ.

**2. Постановка завдання**

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.



## Теоретичні відомості

Для розв'язання задачі обчислення скінченних сум та добутків доречно використання оператора циклу з передумовою `while`, який забезпечує перевірку умови виконання тіла циклу до його безпосереднього виконання.

Оператор `while` формально записується у вигляді  
`while (вираз) тіло_циклу`

Вираз у дужках може набути істинного (`true`) або хибного (`false`) значення. Якщо воно істинне, то виконується тіло циклу і вираз обчислюється заново, а якщо ж ні, – цикл `while` закінчується.

У тілі циклу можуть використовуватися оператори `break` і `continue`. Оператор `break` забезпечує негайний вихід з циклу, оператор `continue` викликає зупинку чергової і початок наступної ітерації.

## 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв'язання задачі мовою C++.

3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

3.4. Оформлення звіту.

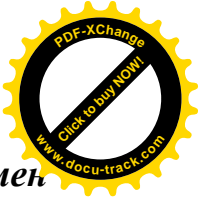
Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

## 4. Приклад

Обчислити значення скінченного добутку  $\cos \frac{\pi}{2^2} \cdot \cos \frac{\pi}{2^3} \cdots \cos \frac{\pi}{2^{n+1}}$ .

**Розв'язання.**



Таблиця символічних імен

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
i	i	Параметр циклу	int	Змінне значення
res	res	Результат розрахунку виразу	double	Результат
π	pi	Математична константа = 3,14	double	Математична константа = 3,14
n	n	Кількість повторів циклу	int	Уводиться з клавіатури

Текст програми мовою C++ для компіляторів Visual C++

```
#include <iostream.h>
#include <math.h>
int main()
{
    int i, n;
    double res; double pi = 3.14;
    cout << "Задайте кількість повторів циклу: ";
    cin >> n;
    if (n >= 1) // якщо кількість повторів задано вірно,
                // виконуються наступні дії
    {
        res = 1; // задаємо початкове значення добутку
        i = 1; // задаємо початкове значення лічильника
        while(i <= n) // цикл виконуватиметься, доки i не дорівнюватиме n
        {
            res *= cos(pi/pow(2,i+1)); // накопичуємо значення добутку
            i++; // значення лічильника збільшується на одиницю
        } // кінець тіла циклу
        cout << endl << "Результат = " << res << endl;
    }
    else // якщо кількість повторів була не вірно задано
    cout<<"Кількість повторів не задана!"<<endl;
    return 0;
}
```

5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

Завдання. Скласти схему алгоритму, таблицю символічних імен та програму для обчислення скінченної суми та добутку відповідно до заданого варіанта:

Знайти значення скінченної суми:

$$1. \frac{\sin x}{1} + \frac{\sin 2x}{2} + \dots + \frac{\sin nx}{n} \quad \left| \quad 2. \frac{1}{(1+1)^2} + \frac{1}{4(2+1)^2} + \dots + \frac{1}{n^2(n+1)^2}.$$



$$3. \frac{1}{1!} + \frac{4}{2!} + \dots + \frac{n^2}{n!}.$$

$$4. 1^4 + 2^4 + \dots + n^4.$$

$$5. 1^2 + 3^2 + \dots + (2n-1)^2.$$

$$6. 1^3 + 2^3 + \dots + n^3.$$

$$7. 1^2 + 2^2 + \dots + n^2.$$

$$8. 2 + 4 + \dots + 2n.$$

$$9. 1 + 3 + \dots + 2n - 1.$$

$$10. 1 + 2 + 3 + \dots + n.$$

$$11. q \sin \alpha + q^2 \sin 2\alpha + \dots + q^n \sin n\alpha.$$

$$12. \frac{1}{(2+1)} + \frac{1}{2(4+1)} + \dots + \frac{1}{n(2n+1)}.$$

$$13. \frac{1}{(1+1)^2} + \frac{3}{4(2+1)^2} + \dots + \frac{2n-1}{n^2(n+1)^2}.$$

$$14. \frac{1}{3} + \frac{1}{8} + \dots + \frac{1}{n^2 - 1}.$$

Обчислити значення скінченного добутку:

$$15. \left(1 + \frac{1}{1(1+2)}\right) \cdot \left(1 + \frac{1}{2(2+2)}\right) \cdot \dots \cdot \left(1 + \frac{1}{n(n+2)}\right).$$

$$16. \frac{5}{8} \cdot \frac{12}{15} \cdot \dots \cdot \frac{n^2 - 4}{n^2 - 1}.$$

$$17. \cos \frac{x}{2} \cdot \cos \frac{x}{2^2} \cdot \dots \cdot \cos \frac{x}{2^n}.$$

$$18. \sin \frac{\pi}{x^2} \cdot \sin \frac{\pi}{x^3} \cdot \dots \cdot \sin \frac{\pi}{x^{n+1}}.$$

$$19. (1+1) \cdot \left(1 + \left(\frac{1}{2}\right)^2\right) \cdot \dots \cdot \left(1 + \left(\frac{1}{2}\right)^{2n}\right).$$

$$20. \frac{2}{\sqrt{2}} \cdot \frac{2}{\sqrt{2+\sqrt{2}}} \cdot \frac{2}{\sqrt{2+\sqrt{2+\sqrt{2}}}} \cdot \dots \cdot \frac{2}{\underbrace{\sqrt{2+\dots+\sqrt{2}}}_n}.$$

$$21. (1+1) \left(1+x^2\right) \cdot \dots \cdot \left(1+x^{2n}\right).$$

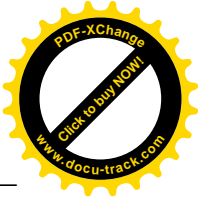
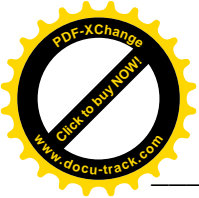
$$22. \left(1 - \frac{2}{2(2+1)}\right) \left(1 - \frac{2}{3(3+1)}\right) \cdot \dots \cdot \left(1 - \frac{2}{n(n+1)}\right).$$

$$23. \frac{7}{9} \cdot \frac{26}{28} \cdot \dots \cdot \frac{n^3 - 1}{n^3 + 1}.$$

$$24. \left(1 - \frac{1}{4}\right) \left(1 - \frac{1}{9}\right) \cdot \dots \cdot \left(1 - \frac{1}{n^2}\right).$$

$$25. a^{-1} \cdot a^{\frac{1}{2}} \cdot \dots \cdot a^{\frac{(-1)^n}{n}}.$$





## ЛАБОРАТОРНА РОБОТА № 6

### ЦИКЛІЧНИЙ ОБЧИСЛЮВАЛЬНИЙ ПРОЦЕС. НЕСКІНЧЕННІ СУМИ

---

#### 1. Мета роботи

Дослідити роботу операторів циклу алгоритмічної мови C++, засвоїти методику складання, налагодження та розрахунку програм циклічних обчислювальних процесів для обчислення нескінченних сум на ПЕОМ.

#### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.

##### Теоретичні відомості

Для розв'язання задачі обчислення нескінченних сум та добутоків доречно використання оператора циклу з постумовою `do while`, який забезпечує виконання тіла циклу перед перевіркою умови виходу з циклу (у задачі – доки не досягнуто значення точності  $\epsilon$ ).

Оператор `do-while` формально записується так:

```
do {тіло_циклу} while (вираз);
```

Основна відзнака між циклами `while` і `do-while` у тому, що тіло в циклі `do-while` виконується як мінімум один раз. Тіло циклу буде виконуватися доти, поки вираз у дужках не набуде значення `false`. Якщо ж воно неправильне ще перед входом у цикл, його тіло виконується рівно один раз.

Дозволяється вкладати одні цикли в інші, тобто в тілі будь-якого циклу можуть з'являтися оператори `for`, `while` і `do-while`.

#### 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв'язання задачі мовою C++.

3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;



- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

#### 4. Приклад

Обчислити наближене значення нескінченної суми  $\frac{1}{1^4} + \frac{1}{3^4} + \dots + \frac{1}{(2n+1)^4} + \dots$

з точністю  $\varepsilon=0.0000119$ .

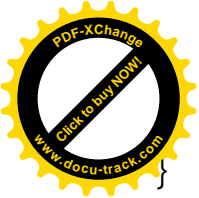
#### Розв'язання.

*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
i	i	Параметр циклу	int	Змінна
memb	memb	Член ряду	double	Змінна
r	r	Сума	double	Результат

#### Текст програми мовою C++ для компіляторів Visual C++

```
#include <iostream.h>
#include <iomanip.h>
#include <cmath>
int main()
{
    int i;
    double r, memb;
    i = 0; // задаємо початкове значення лічильника
    r = 0; // задаємо початкове значення суми
    memb = static_cast<float>(1)/pow((2*i + 1),4); // задаємо
                                                // значення першого члена ряду
    cout << setprecision(10) << setiosflags(ios::fixed |
        ios::showpoint); // установки екрана
    do // початок тіла циклу
    {
        r += memb; // накопичення суми
        memb = static_cast<float>(1)/pow((2*i + 1),4);
                                                // підраховуємо кожний наступний член ряду
        i++; // збільшуємо лічильник на одинцю
    } while (fabs(memb) >= 0.0000119); // тіло циклу буде
                                                // виконуватися, доки модуль члена ряду не зменшиться до ε
    cout << "Результат з точністю до eps = 0.0000119: "
        << r << endl;
```



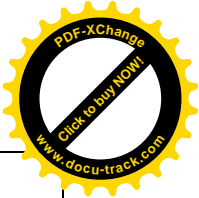
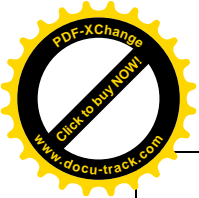
#### 4. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обчислення нескінченних сум та добутків відповідно до заданого варіанта.

Обчислити наближене значення нескінченної суми з точністю  $\varepsilon$ :

Варіант	Функція	Точність $\varepsilon$
1	$1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!} + \dots$	0.001
2	$1 - \frac{1}{1!} + \frac{1}{2!} - \frac{1}{3!} + \dots \pm \frac{1}{n!} \mp \dots$	0.005
3	$1 + \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{2^n} + \dots$	0.0001
4	$1 - \frac{1}{2} + \frac{1}{4} - \frac{1}{8} + \dots \pm \frac{1}{2^n} \mp \dots$	0.05
5	$\frac{1}{1 \cdot 3} + \frac{1}{3 \cdot 5} + \dots + \frac{1}{(2n-1)(2n+1)} + \dots$	0.00005
6	$\frac{1}{3 \cdot 5} + \frac{1}{7 \cdot 9} + \dots + \frac{1}{(4n-1)(4n+1)} + \dots$	0.0001
7	$\frac{1}{1^2} + \frac{1}{3^2} + \dots + \frac{1}{(2n+1)^2} + \dots$	0.001
8	$1 + \frac{1}{2^4} + \frac{1}{3^4} + \dots + \frac{1}{n^4} + \dots$	0.005
9	$1 - \frac{1}{2^4} + \frac{1}{3^4} - \dots \pm \frac{1}{n^4} \mp \dots$	0.0005
10	$\frac{1}{1^4} + \frac{1}{3^4} + \dots + \frac{1}{(2n+1)^4} + \dots$	0.01
11	$\frac{1}{1 \cdot 4} + \frac{1}{4 \cdot 7} + \dots + \frac{1}{(3n-2)(3n+1)} + \dots$	0.05
12	$x - \frac{x^3}{3!} + \frac{x^5}{5!} - \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!} \pm \dots$	0.005
13	$x - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots + (-1)^n \frac{x^{2n}}{(2n)!} \pm \dots$	0.0001
14	$\frac{x-1}{x} + \frac{(x-1)^2}{2x^2} + \dots + \frac{(x-1)^n}{nx^n} + \dots$	0.00001
15	$x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots + (-1)^{n+1} \frac{x^n}{n} \pm \dots$	0.00001



16	$x + \frac{x^3}{3!} + \frac{x^5}{5!} + \dots + \frac{x^{2n+1}}{(2n+1)!} + \dots$	0.05
17	$1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots + \frac{x^{2n}}{(2n)!} + \dots$	0.001
18	$x + \frac{x^3}{3} + \frac{x^5}{5} + \dots + \frac{x^{2n+1}}{2n+1} + \dots$	0.01
19	$1 + \frac{1}{2} \left( 2 \cdot 3x + 3 \cdot 4x^2 + 4 \cdot 5x^3 + \dots \right)$	0.05
20	$1 + \frac{x \ln a}{1!} + \frac{(x \ln a)^2}{2!} + \dots + \frac{(x \ln a)^n}{n!} + \dots$	0.05
21	$\frac{\pi}{2} - \left( x - \frac{x^3}{3} + \frac{x^5}{5} - \dots + (-1)^n \frac{x^{2n+1}}{2n+1} \pm \right)$	0.00005
22	$\frac{1}{2} + \frac{3}{2^2} + \dots + \frac{2n-1}{2^n} + \dots$	0.0005
23	$1 + \frac{1}{4}x - \frac{1 \cdot 3}{4 \cdot 8}x^2 + \frac{1 \cdot 3 \cdot 6}{4 \cdot 8 \cdot 12}x^3 - \frac{1 \cdot 3 \cdot 6 \cdot 10}{4 \cdot 8 \cdot 12 \cdot 16}x^4 \pm \dots$	0.0001
24	$1 - \frac{1}{2^2} + \frac{1}{3^2} - \frac{1}{4^2} + \dots \pm \frac{1}{n^2} \mp \dots$	0.001
25	$1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!} + \dots$	0.005

**ЛАБОРАТОРНА РОБОТА № 7  
ОДНОВИМІРНІ МАСИВИ**

**1. Мета роботи**

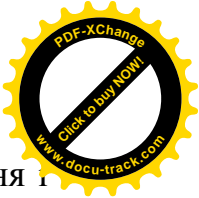
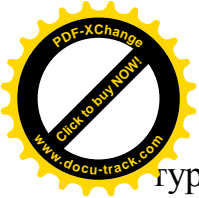
Дослідити роботу операторів циклу мови програмування C++, засвоїти методику складання, налагодження та розрахунку програм циклічних обчислювальних процесів для обробки векторів на ПЕОМ.

**2. Постановка завдання**

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.

**Теоретичні відомості**

Масив являє собою набір однотипних елементів. До всього масиву можна звертатися за ім'ям, а до окремого елемента масиву – за ім'ям з індексом елемента у квадратних дужках. Індекс свідчить про його відносну позицію у струк-



гурі масиву. Число елементів масиву встановлюється під час його визначення і надалі не змінюється. Масиви визначаються так само, як і змінні:

```
int a[100];
char b[20];
float d[50];
```

У першому рядку визначається масив `a` з 100 елементів цілого типу: `a[0], a[1], ..., a[99]` (індексація завжди починається з нуля), у другому елементи масиву `b` мають тип `char`, а в третьому – `float`.

Мова C++ дозволяє ініціалізувати масив при його визначенні. Для цього використовується така форма:

```
тип <ім'я_масиву>[...] ... [...] = {список значень};
```

Приклади.

```
int a[5] = {0, 1, 2, 3, 4};
char ch[3] = {'d', 'e', '9'};
```

Якщо масив не був ініціалізованим зазначеним вище чином, призначити значення його елементам можна і безпосередньо у функціях за допомогою циклу. Цикл надає також можливість обробляти та змінювати дані масиву.

Приклад.

```
for ( i = 0; i < N; i++ )
{
    cout << "Введіть елемент ";
    cin >> c[ i ]; // уведення значень елементів з програми
}
for ( i = 0; i < N; i++ )
    c[ i ]=abs(c[i]) //зміна первинного значення
                    //елемента на його модуль
for ( i = 0; i < N; i++ )
    cout << c[ i ] << " "; // виведення елементів на екран дисплея.
```

### 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;



- результати розрахунку задачі;
- висновки.

#### 4. Приклад

У масиві  $C = (c_1, c_2, \dots, c_{16})$  визначити середнє арифметичне  $sr$  негативних елементів першої половини масиву та добуток  $p$  позитивних елементів другої половини масиву.

#### Розв'язання.

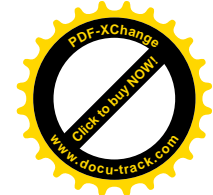
*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
$p$	$p$	Добуток додатних елементів другої половини масиву	int	Результат
$i$	$i$	Індекс елементів масиву	int	Початкові дані
$s$	$s$	Сума від'ємних елементів	int	Результат
$k$	$k$	Кількість від'ємних елементів першої половини масиву	int	Результат
$l$	$l$	Кількість додатних елементів другої половини масиву	int	Результат
$sr$	$sr$	Середнє арифметичне від'ємних елементів першої половини масиву	float	Результат
$m$	$m$	Половина масиву	int	Результат
$N$	$N$	Кількість елементів у масиві	int	16
$c$	$c$	Масив з кількістю елементів $N$	int	Початкові дані

#### Текст програми мовою C++

для компіляторів Visual C++

```
#include <iostream.h>
#include <iomanip.h>
const int N = 16;
int main()
{
    int c[ N ] = { 2, -6, 3, -8, 2, 3, 5, -5, -2, 5, 8,
                  6, -4, -9, 2, 5 }; // визначення масиву C
    int p,i,m,k,l,s;
    float sr;
    cout << "Вхідний масив:" << endl;
    for ( i = 0; i < N; i++ ) // ввід значень масиву
        cout << c[ i ] << " ";
    cout << endl;
    cout << setprecision(3) << setiosflags(ios::fixed |
```



```
ios::showpoint);
k = 0; // початкове значення кількості від'ємних елементів
      // першої половини масиву
s = 0; // початкове значення суми від'ємних елементів масиву
m = N/2; // половина масиву
for (i = 0; i < m; i++) //опрацьовується перша половина масиву
    if ( c[ i ] < 0 ) // пошук від'ємних елементів
    {
        s += c[ i ]; // накопичення суми від'ємних елементів
        k++;        // накопичення кількості від'ємних елементів
    }
p = 1; // початкове значення добутку додатних елементів
      // другої половини масиву
l = 0; // початкове значення кількості додатних елементів масиву
for (i = m; i < N; i++) //опрацьовується друга половина масиву
    if ( c[ i ] > 0 ) // пошук додатних елементів
    {
        p *= c[i]; // накопичення добутку додатних елементів
        l++;      // накопичення кількості додатних елементів
    }
if ( k == 0 ) // перевірка на наявність від'ємних елементів
    cout << "Немає від'ємних елементів у першій по
           ловині масиву!" << endl;
else
{
    sr = static_cast< float >( s )/k; // підрахунок середньо
                                     // арифметичного значення
    // вивід результатів
    cout <<"Сума від'ємних елементів першої половини
           масиву = " << s << endl;
    cout << "Кількість від'ємних елементів першої
           половини масиву = " << k << endl;
    cout << "Середнє арифметичне. від'ємних елементів
           першої половини масиву = " << sr <<endl;
}
if ( l == 0 ) // перевірка на наявність додатних елементів
    cout << "Немає додатних елементів у другій половині
           масиву!" << endl;
else
{
    // вивід результатів
    cout << "Кількість додатних елементів у другій
           половині масиву= " << l << endl;
    cout << "Добуток додатних елементів у другій
           половині масиву= " << p << endl;
}
```



```
return 0;
```

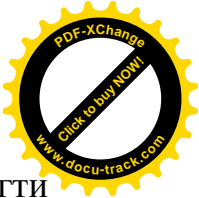
## 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

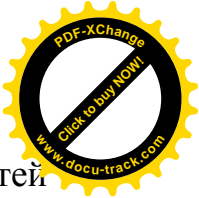
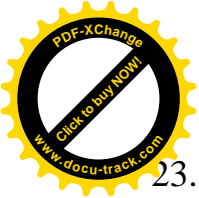
**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обробки вектора відповідно до заданого варіанта.

- Відомі два вектори  $X = (x_1, x_2, \dots, x_{10})$  і  $Y = (y_1, y_2, \dots, y_{10})$ . Знайти скалярний добуток векторів і кут між ними. При обчисленні застосувати формулу  $(x, y) = |x| \cdot |y| \cos \varphi$ .
- Відомі два вектори  $X = (x_1, x_2, \dots, x_{15})$  і  $Y = (y_1, y_2, \dots, y_{15})$ . Знайти кут між ними та модуль векторного добутку векторів  $X$  і  $Y$ . При обчисленні застосувати формулу  $|c| = |x| \cdot |y| \sin \varphi$ , де  $c$  – векторний добуток векторів  $X$  і  $Y$ .
- Відомі два вектори  $X = (x_1, x_2, \dots, x_7)$  і  $Y = (y_1, y_2, \dots, y_7)$ . Знайти суму векторів  $X+Y$  і різницю векторів  $X - Y$  та зберегти їх значення у векторах  $Z$  і  $F$ .
- Відомі два вектори  $X = (x_1, x_2, \dots, x_{20})$  і  $Y = (y_1, y_2, \dots, y_{20})$ . Знайти добуток векторів  $X$  і  $Y$  на число  $\alpha$  і зберегти їх результат у векторах  $G$  і  $H$ .
- Відомі два вектори  $X = (x_1, x_2, \dots, x_{20})$  і  $Y = (y_1, y_2, \dots, y_{20})$ . Знайти лінійну комбінацію векторів  $\alpha X + \beta Y$  і зберегти її у векторі  $U$ .
- Обчислити 20 значень функції  $y = x^2$  на відрізку  $[a, b]$  і зберегти значення аргументу функції в масиві  $X$ , а відповідні значення функції – у масиві  $Y$ .
- Обчислити 25 значень функції  $y = ax^2 + bx + c$  на відрізку  $[e, f]$  та зберегти їх у масиві  $Y$ . Знайти мінімальне і максимальне значення в цьому масиві.
- Обчислити 12 значень функції  $y = ax^2 + bx + c$  на відрізку  $[e, f]$ , зберегти їх у масиві  $Y$  і визначити, чи є функція спадною, зростаючою або немонотонною на цьому відрізку.
- Обчислити 18 значень функції  $y = ax^2 + bx + c$  на відрізку  $[e, f]$ , зберегти їх у масиві  $Y$  і визначити, чи має рівняння  $ax^2 + bx + c = 0$  на відрізку  $[e, f]$  принаймні хоча б один корінь.
- Обчислити 30 значень функції  $y = ax^2 + bx + c$  на відрізку  $[e, f]$ , зберегти їх у масиві  $Y$ . Негативні компоненти масиву  $Y$  скопіювати в масив  $Y_{negative}$ , а позитивні – у масив  $Y_{positive}$ .





11. Обчислити 30 значень функції  $y = ax^2 + bx + c$  на відрізку  $[e, f]$ , зберегти результати у масиві  $Y$ . Обчислити кількість негативних і позитивних компонентів масиву.
12. У масиві  $Time(20)$  зберігаються результати (у секундах) забігу на 100 метрів для 20 спортсменів. Визначити кращий результат і швидкість кожного спортсмена.
13. У масиві  $Time(20)$  зберігаються результати (у секундах) забігу на 100 метрів для 20 спортсменів. Визначити середню швидкість пробігу для кожного спортсмена.
14. У масиві  $X$  зберігаються дані про вартість (у гривнях) кожного кілограма із семи основних продуктів харчування, а масиві  $Y$  – про добову потребу (у грамах) людини в цих продуктах. Визначити вартість добового раціону.
15. У масиві  $Mass$  зберігаються дані про результати 22 зважувань маси тіла. Визначити середнє значення маси тіла та оцінити помилку виміру.
16. У масиві  $Urogai(20)$  зберігаються дані про врожайність пшениці (у центнерах з гектара) сільгоспприємства “Зоря” з 1981 по 2000 р. Визначити найбільш і найменш урожайні роки у господарстві.
17. У масиві  $Kurs1(7)$  зберігаються дані про курс купівлі долара стосовно гривні для семи різних банків, а в масиві  $Kurs2(7)$  – про курс продажу долара стосовно гривні для цих банків. Визначити найвигідніший курс обміну і банки, що працюють за цим курсом.
18. У масиві  $Meat(5)$  зберігаються дані про вартість (у гривнях) кілограма м'яса на п'яти різних ринках м. Дніпропетровська. Визначити середню, мінімальну і максимальну вартість м'яса. Вивести на екран “чорний список” ринків, де вартість м'яса вища середньої.
19. У масиві  $T(10)$  зберігаються дані про температуру повітря на 1 січня 2009 р. у 10 містах світу. Визначити кількість міст, в яких температура була від'ємною, додатною і нульовою.
20. У масиві  $G(10)$  зберігаються відомості про довжину гіпотенуз 10 прямокутних трикутників, а в масиві  $C(10)$  – про довжину відповідних їм катетів. Для кожного з трикутників знайти довжину другого катета і площу та записати ці значення в масиви з іменами  $B$  і  $S$ .
21. У масивах  $Tmin(15)$  і  $Tmax(15)$  зберігаються дані про мінімальну і максимальну температуру, прийнятну для життя 15 різних рослин. Визначити кількість тих, що загинуть при температурі  $t$  чи які перебуватимуть на грані загибелі. Гранню загибелі вважати температуру, що дорівнює межовій.
22. У масиві  $Tair(10)$  зберігаються відомості про середнє арифметичне значення температури води в Чорному морі за липень для 10 різних міст Криму, а в масиві  $Twater(10)$  – про середнє арифметичне значення температури повітря за липень в тих же містах. Визначити місто, в якому різниця між температурою повітря і води мінімальна.



23. У масиві *Square*(6) зберігаються відомості про площу шести областей України, а в масиві *Population*(6) – про кількість населення тих же областей. Визначити найбільш населену і найменш населену області.
24. У масиві *T*(12) містяться дані про дану про середньомісячну температуру для кожного з 12 місяців у Дніпропетровську. Визначити найбільшу від’ємну і найменш додатну температуру.
25. Записати 25 цілих чисел в масив *X*. Переписати від’ємні з них підряд у масив *P*, а додатні – у масив *N*.

---

## ЛАБОРАТОРНА РОБОТА № 8 ДВОВИМІРНІ МАСИВИ (МАТРИЦІ)

---

### 1. Мета роботи

Дослідити роботу операторів циклу мови програмування C++, засвоїти методику складання, налагодження та розрахунку програм для процесів обробки матриць на ПЕОМ.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно заданого варіанта. Налагодити та розв’язати задачу на ПЕОМ.

#### Теоретичні відомості

Двовимірний масив (матриця) подається як одновимірний, елементи якого – також масиви. Наприклад, визначення `char a[10][20]` задає масив, що складається з 10 одновимірних масивів розміром 20 елементів, кожен з яких – символного типу. Елементи двовимірного масиву зберігаються по рядках, тобто якщо ними проходити в напрямку їх розміщення в пам’яті, то спочатку буде змінюватися правий індекс. Наприклад, звернення до дев’ятого елементу п’ятого рядка запишеться `a[5][9]`.

Задано масив:

```
int a[2][3];
```

Елементи масиву *a* так розміщені в пам’яті:

```
a[0][0], a[0][1], a[0][2], a[1][0], a[1][1], a[1][2].
```

Ініціалізувати матрицю, як і вектор, можна під час визначення масиву:

```
int a[ M ][ N ] = { { 11, 2, 6, 9, 7, 14 }, { 14, 25,  
36, 17, 18, 3 } };
```

або безпосередньо в функції:

```
for( i = 0; i < M; i++ )  
{  
    for ( j = 0; j < N; j++ )  
        cout <<"Уведіть елементи матриці";
```



```
cin>> a[ i ][ j ];
```

```
}
```

Доступ до елементів можливий за допомогою вкладених циклів `for`, де перший цикл надає доступ до перебирання лівих індексів (рядків), а другий, укладений – до правих індексів (стовпців).

### 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

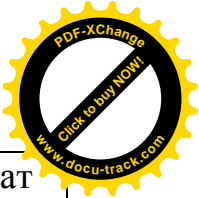
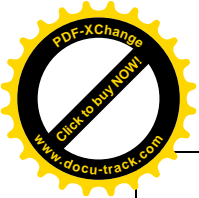
### 4. Приклад

Дано матрицю  $T(2,6)$ . Мінімальний і максимальний елементи матриці поміняти місцями.

**Розв’язання.**

*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
A	a	Матриця з N рядків та M стовпців	int	Початкові дані
i,j	i,j	Індекси елементів матриці A	int	Початкові дані
max	max	Максимальний елемент	int	Результат
min	min	Мінімальний елемент	int	Результат



imax	imax	i - й індекс максимального елемента	int	Результат
jmax	jmax	j - й індекс максимального елемента	int	Результат
imin	imin	i - й індекс мінімального елемента	int	Результат
jmin	jmin	j - й індекс мінімального елемента	int	Результат
vsp	vsp	Буферна змінна	int	Допоміжна змінна
N	N	Кількість стовпців у матриці	int	6
M	M	Кількість рядків у матриці	int	2

**Текст програми мовою C++  
для компіляторів Visual C++2**

```
# include <iostream.h>
const int M = 2;
const int N = 6;
int main()
{
    // ініціалізація масиву a
    int a[M][N]={{11,2,6,9,7,14},{14,25,36,17,18,3}};
    int i, j, max, min, imax, imin, jmax, jmin, vsp;
    cout << "Isxodniy massiv:" << endl;
    for( i = 0; i < M; i++ ) // вивід вихідного масиву
    {
        for ( j = 0; j < N; j++ )
            cout << a[ i ][ j ] << '\t';
        cout << endl;
    }
    max = a[0][0]; // допускаємо, що елемент a[0][0] максимальний
    min = a[0][0]; // допускаємо, що елемент a[0][0] мінімальний
    // відповідно до цього зазначаємо їх індекси
    imax = 0;    jmax = 0;
    imin = 0;    jmin = 0;
    for ( i = 0; i < M; i++ ) // виконуємо доступ до рядків
        for ( j = 0; j < N; j++ ) // виконуємо доступ до стовпців
        {
            if (max < a[i][j]) // пошук максимального елемента
            {
                max = a[ i ][ j ]; // фіксуємо значення елемента
                imax = i; // фіксуємо номер рядка
                jmax = j; // фіксуємо номер стовпця
            }
        }
}
```



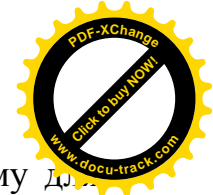
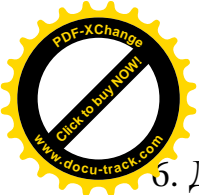
```
if (min > a[i][j]) // пошук мінімального елементу
{
    min = a[ i ][ j ]; // фіксуємо значення елементу
    imin = i; // фіксуємо номер рядка
    jmin = j; // фіксуємо номер стовпця
}
}
cout << endl;
cout << "Максимальний елемент = " << max << endl;
cout << "Мінімальний елемент = " << min << endl;
// міняємо місцями мінімальний і максимальний елементи:
vsp = a[imax][jmax]; // копіюємо максимальний елемент в буфер
a[imax][jmax] = a[imin][jmin]; // на місце максимального
// елементу записуємо мінімальний
a[imin][jmin] = vsp; // на місце мінімального елементу записуємо
// дані з буфера
cout << endl << "Масив після перестановки: " << endl;
for( i = 0; i < M; i++ )
{
    for ( j = 0; j < N; j++ )
        cout << a[ i ][ j ] << '\\t';
    cout << endl;
}
return 0;
}
```

## 5. Варіанти завдань

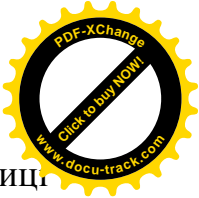
Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для обробки матриці відповідно заданого варіанта.

1. Дано масив розміром  $5 \times 6$  з елементами цілого типу. Написати програму для обчислення кількості елементів масиву, кратних трьом.
2. Дано масив розміром  $7 \times 2$  з елементами дійсного типу. Написати програму для обчислення суми всіх додатних елементів масиву.
3. Дано масив розміром  $3 \times 3$  з елементами символьного типу. Написати програму для обчислення кількості елементів масиву, відмінних від нуля.
4. Дано масив розміром  $4 \times 6$  з елементами цілого типу. Написати програму для обчислення суми елементів масиву.
5. Дано масив розміром  $4 \times 3$  з елементами дійсного типу. Написати програму для обчислення добутку елементів масиву, модуль яких більше одиниці.



6. Дано масив розміром  $3 \times 5$  з елементами цілого типу. Написати програму для обчислення кількості елементів масиву, що при діленні на п'ять дають у залишку одиницю.
7. Дано масив розміром  $5 \times 2$  з елементами дійсного типу. Написати програму, що визначає наявність від'ємних елементів в масиві.
8. Дано масив розміром  $2 \times 4$  з елементами символного типу. Написати програму для обчислення кількості елементів масиву, відмінних від '0'.
9. Дано масив розміром  $2 \times 4$  з елементами дійсного типу. Написати програму, що визначає наявність у масиві хоча б одного елемента, що дорівнює нулю.
10. Дано масив розміром  $4 \times 2$  з елементами цілого типу. Написати програму для обчислення кількості елементів масиву, модуль яких менший 12.
11. Дано масив розміром  $2 \times 5$  з елементами дійсного типу. Написати програму для обчислення кількості елементів масиву, ціла частина яких дорівнює семи.
12. Дано масив розміром  $3 \times 4$  з елементами дійсного типу. Написати програму для обчислення кількості елементів масиву, що при округленні дають шість.
13. Дано масив розміром  $3 \times 2$  з елементами цілого типу. Написати програму для обчислення суми квадратів елементів масиву.
14. Дано масив розміром  $6 \times 2$  з елементами дійсного типу. Написати програму для обчислення номера рядка, в якому розміщено мінімальний елемент масиву.
15. Дано масив розміром  $3 \times 5$  з елементами цілого типу. Написати програму для обчислення номера стовпця, де міститься мінімальний елемент масиву.
16. Дано масив розміром  $4 \times 3$  з елементами дійсного типу. Написати програму для обчислення кількості елементів масиву, дробова частина яких менша 0,1.
17. Дано масив розміром  $3 \times 6$  з елементами дійсного типу. Написати програму для обчислення добутку всіх позитивних елементів масиву.
18. Дано масив розміром  $6 \times 2$  з елементами дійсного типу. Написати програму для обчислення виразу  $max - min$ , де  $max$  – максимальний за модулем елемент масиву, а  $min$  – мінімальний за модулем елемент масиву.
19. Дано масив розміром  $5 \times 6$  з елементами цілого типу. Написати програму для обчислення кількості позитивних елементів масиву, кратних п'яти.
20. Дано масив розміром  $7 \times 2$  з елементами дійсного типу. Написати програму для обчислення суми всіх від'ємних елементів масиву, кратних двом.
21. Дано масив розміром  $3 \times 5$  з елементами цілого типу. Написати програму для обчислення кількості елементів масиву, що при діленні на шість дають у залишку одиницю.



22. Знайти середнє геометричне позитивних елементів кожного рядка матриці  $Y(5 \times 6)$ .
23. Знайти найбільший елемент матриці  $X(4 \times 4)$  і записати одиницю у той рядок і стовпець, в яких він міститься.
24. Знайти найбільший елемент матриці  $A(2 \times 3)$  і номер рядка та стовпця, в якому він міститься.
25. Дано масив розміром  $4 \times 4$  з елементами символьного типу. Написати програму для обчислення кількості елементів масиву, що дорівнюють '0'.

---

## ЛАБОРАТОРНА РОБОТА № 9 ОБРОБКА СИМВОЛЬНОЇ ІНФОРМАЦІЇ

---

### 1. Мета роботи

Дослідити роботу операторів обробки символьної інформації мови програмування C++, засвоїти методику складання та налагодження програм з опрацюванням символьної інформації на ПЕОМ.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налаштувати та розв'язати задачу на ПЕОМ.

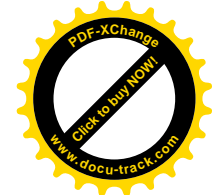
#### Теоретичні відомості

У сімействі персональних комп'ютерів IBM PC використовуються 256 різноманітних символи. Вони мають свої числові коди, значення котрих лежать у діапазоні від 0 до 255, тобто загальна кількість символів дорівнює 256.

Коли Ви натискаєте клавішу на клавіатурі, це приводить до того, що у комп'ютер надсилається сигнал у виді двійкового числа, яке ставиться у відповідність до *кової таблиці*, тобто внутрішнього представлення символів у комп'ютері. Як стандарт прийнята *таблиця ASCII* (*American Standard Code for Information Interchange* – *Американський стандартний код обміну інформацією*). Вона указує на відповідність між зображеннями або умовними позначеннями символів та їх внутрішніми числовими кодами. Ознайомитися з цією таблицею можна в додатку 1.

Мова C++ дає змогу визначати тип рядка двома різними способами і у першому використовується масив символів, а в другому – покажчик на перший символ масиву:

```
char array[6] = "РЯДОК";  
char s[ ] = { 'Р', 'Я', 'Д', 'О', 'К', '\0' }; // при  
визначенні масиву з одночасною ініціалізацією  
кінцеве значення можна не подавати;
```



`char *b;` // другий спосіб визначення рядка – це використання покажчика на символ. Він задає змінну `b`, де може бути адреса певного об'єкта (де міститься рядок).

Під час виконання завдань до лабораторної роботи можна використовувати функції, призначені для роботи з рядками (їх прототипи – у файлі `stdio.h`, `string.h`):

`scanf( )`, `gets( )` – уведення рядків символів;  
`printf( )`, `puts( )` – виведення рядків символів;  
`strcpy(рядок1, рядок2)` – копіювання змісту рядка2 у рядок1;  
`strcat(рядок1, рядок2)` – рядок2 дописується до рядка1, рядок2 не змінюється;  
`strlen(рядок1)` – повертає розмір рядка1, не враховуючи нульовий байт;  
`strcmp(рядок1, рядок2)` – порівнює рядок1 з рядком2 та повертає нуль, якщо вони однакові.

Рядок можна привласнюватимасиву, використовуючи операцію `cin` (взяти з потоку):

```
char word [20];  
cin>>word;
```

Уведений програмістом рядок зберігається у змінній `word`. Зчитуються символи доти, поки не зустрінеться символ нового рядка або кінець файлу. Кількість уведених символів не повинно перевищувати 19, для того, щоб залишилося місце під символ кінця строки `'\0'`.

Якщо в рядок необхідно вводити слова, розділені пропусками, використовується функція `cin.getline( )` з трьома аргументами.

```
char sentence [80];  
cin.getline(sentence, 80, '\n');
```

де `sentence` – масив символів для тексту, `80` – довжина масиву, `'\n'` – символ – обмежник (ним може бути будь-який символ).

Ввід припиняється, якщо зустрінеться символ `'\n'`, кінець файлу або кількість лічених символів виявилась на один менше, ніж вказано в дрогому аргументі(з урахуванням `'\0'`).

Інформацію про найбільш вживані операції для роботи з рядками можна знайти у додатку 2.

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

– скласти таблицю символічних імен;





- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

### 3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

### 3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

## 4. Приклад

**Задача 1.** З клавіатури вводяться символи доти, поки не зустрінеться знак оклику ‘!’. Підрахувати у введеному рядку кількість символів ‘о’.

**Розв’язання.**

*Таблиця символічних імен*

Математичне ім’я	Ім’я у програмі	Смислове значення	Змінна	
			Тип	Значення
i	i	Параметр циклу	int	Змінне значення
k	k	Кількість символів ‘о’	int	Змінне значення
n	n	Лічильник, кількість символів, уведених до символу ‘!’	int	Змінне значення
c[100]	c[100]	Масив символів із 100 елементів	char	З’ясовується у процесі

### Текст програми мовою C++

для компіляторів Visual C++

```
#include <iostream.h>
#include <string.h>
int main()
{
    int i, n, k; n = 0; char c[100];
    cout << "Уведіть рядок:" << endl;
    cin >> c[ n ]; // уводиться перший символ рядка
    while ( c[n]!='!' ) // умова вводу символів, доки не зустрінеться ‘!’
    {
        n++; // лічильник збільшується на одиницю
    }
}
```



```

    cin >> c[ n ]; // уводиться наступний символ
}
k = 0; // початкове значення кількості символів 'o'
for ( i = 0; i < n; i++) // відкриваємо цикл для перебору елементів
{
    if ( c[ i ] == 'o' ) // якщо поточний елемент матриці = 'o',
        k++; // кількість збільшується на одиницю
}
// вивід результатів
if ( k == 0 )
    cout << "Немає символів 'o' " << endl;
else
    cout <<"Кількість символів 'o' = " << k << endl;
return 0;
}

```

**Задача 2.** З клавіатури уводиться ціле число. Виявити, ASCII коду якого символу воно відповідає та представити цей символ як ціле число.

**Розв’язання.**

*Таблиця символічних імен*

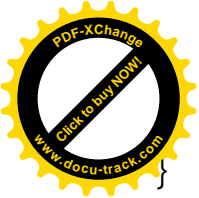
Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
int1	int1	Довільне число	int	Уводиться з клавіатури
int2	int2	Число	int	Результат
ch	ch	Символ	char	Результат

**Текст програми мовою C++ для компіляторів Visual C++**

```

#include <iostream.h>
#include <string.h>
void main ()
{
    char ch;
    int int1, int2;
    cin>>int1; // уводимо ціле число з проміжка 0 .. 255
    ch=char(int1); // конвертуємо число у відповідний символ
    cout<<"ch = " <<ch<<endl; // виводимо на екран символ
    int2=int(ch); // конвертуємо символ назад у число
    cout<<"int2 = " <<int2<<endl; // виводимо число на екран
}

```

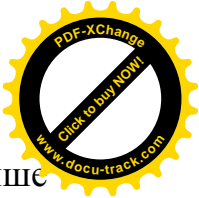
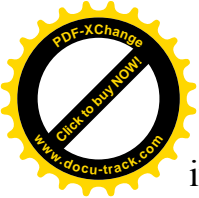


## 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для опрацювання символічної інформації відповідно до заданого варіанта.

1. З клавіатури вводяться символи доти, поки не зустрінеться крапка. Підрахувати кількість уведених символів, букв і цифр, що зустрілися у введеній фразі.
2. З клавіатури вводяться символи доти, поки не зустрінеться крапка або кількість уведених символів перевищить 79. Вивести на екран загальну кількість символів.
3. З клавіатури вводяться символи доти, поки не зустрінеться символ ‘ \* ’. На екран виводити код кожного введенного символу.
4. З клавіатури посимвольно вводиться речення. Підрахувати кількість голосних і приголосних букв у ньому та обчислити їх відсотковий вміст.
5. З клавіатури вводиться послідовність символів доти, поки не зустрінеться символ, що не є цифрою. Підрахувати кількість уведених символів.
6. З клавіатури вводиться послідовність символів. Якщо символ – мала літера алфавіту, то вивести на екран відповідну йому велику літеру. Уведення символів припинити, якщо зустрінеться символ ‘ # ’.
7. З клавіатури вводиться довільне ціле число від 0 до 255. Вивести на екран символ, що відповідає цьому коду. Програма повинна припинити роботу з уведенням числа 256.
8. З клавіатури посимвольно вводиться речення. Визначити, воно розповідне, питальне чи окличне і вивести відповідне повідомлення на екран. Кінцем речення вважати знаки: ‘ ? ’, ‘ ! ’, і ‘ . ’.
9. З клавіатури вводяться символи доти, поки речення не закінчиться. Кінцем речення вважати символи: ‘ ! ’, ‘ ? ’. Підрахувати кількість уведених символів, відмінних від пропуску.
10. З клавіатури вводиться фраза доти, поки не зустрінеться крапка. Підрахувати кількість слів у введеній фразі. Вважати, що слова відокремлюються одне від одного пропусками, комами, двокрапкою, крапкою з комою, тире.
11. З клавіатури посимвольно вводиться речення, в якому визначити кількість слів із закінченням на ‘ я ’.
12. З клавіатури посимвольно вводиться речення, де визначити кількість слів, які не містять букви ‘ у ’.
13. З клавіатури посимвольно вводиться фраза. Визначити, скільки в ній власних назв, тобто слів з великої літери. Фраза вважається введеною, якщо подальший уведений символ – крапка.
14. З клавіатури посимвольно вводиться речення. Якщо воно починається зі слова ‘ Коли ’, вивести випадкове число в діапазоні від 1001 до 1996, а якщо з



іншого слова, то вивести на екран повідомлення: ‘Будь ласка, введіть інше речення’.

15. З клавіатури посимвольно вводиться фраза. Визначити, із символів якого алфавіту складається ця фраза, і вивести на екран одне з повідомлень: ‘Ви використали тільки символи латинського алфавіту’, ‘Ви використали тільки символи кирилиці’, або ‘Ви користувалися символами різних алфавітів’. Фраза введена, якщо наступний уведений символ – крапка.
16. У введеному рядку замінити всі коми на крапки, а крапки – на знаки оклику. Підрахувати кількість таких замін.
17. У рядок через пропуск записано не більше 10 окремих слів. Записати кожне слово в окрему рядкову змінну і вивести на екран кожне слово в окремому рядку.
18. Записати в сім рядкових змінних стільки ж прислів'їв. Підрахувати, у кількох з них зустрічається слово ‘робота’ або однокореневі з ним слова.
19. У кожній з 10 рядкових змінних міститься назва художнього твору і прізвище автора. Вивести на екран наявні серед них твори Т.Г. Шевченка.
20. У рядкову змінну через пропуск записані марки закордонних автомобілів. Замінити їх в цьому рядку *Mersedes* на “Запорожець”, *Opel* на “Жигулі”, *BMW* на “Таврію”, *Reno* на “Москвич”, *Fiat* на “Нива”, *Shkoda* на “Волинь”.
21. Рядок містить деяке слово. Записати в другий рядок (через пропуск) коди символів вихідного рядка, а в третій – вихідний рядок, знаки ‘->’ і отриманий рядок (наприклад, ‘АВВА -> 65 66 66 65’).
22. Записати в 10 рядкових змінних прізвища, імена і по батькові 10 великих письменників. Підрахувати кількість Олександрів у цьому списку і записати їхні прізвища (через пропуск) в окремий рядок.
23. У введеному рядку замінити всі пропуски на коми, а коми – на крапки. В утвореному рядку підрахувати кількість ком.
24. Записати в масив 10 рядкових змінних. б'єднати всі його елементи в один рядок.
25. Записати у вісім рядкових змінних стільки ж рядків з популярними піснями. Підрахувати, в яких з них зустрічається слово ‘кохання’ або однокореневі з ним слова.

---

## ЛАБОРАТОРНА РОБОТА № 10 ФУНКЦІЇ

---

### 1. Мета роботи

Дослідити роботу, призначення, форму запису та особливості вживання функцій мови програмування C++. Засвоїти методику складання програм на ПЕОМ з використанням функцій.



## 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу на ПЕОМ.

### Теоретичні відомості

Програми мовою C++ зазвичай складаються з великої кількості окремих функцій (підпрограм). Вони можуть міститися як в одному, так і в кількох файлах. Усі функції є глобальними. У мові C++ забороняється визначати одну функцію в тілі іншої. Зв'язок між функціями забезпечується через аргументи, повернуті значення й зовнішні змінні.

Функція позначається таким чином:

```
тип ім'я_функції(тип ім'я_параметра_1, тип
                ім'я_параметра_2, ...);
```

Тип функції визначає тип значення, що повертає функція.

При встановленні функції для кожного її параметра можна зазначити лише його тип (наприклад: тип функція (int, float, ...)) та надати ім'я (наприклад: тип функція (int a, float b, ...)).

У мові C++ дозволяється складати функції зі змінною кількістю параметрів. При цьому під час визначення прототипу замість останнього з них ставляться крапки:

```
тип ім'я_функції(тип ім'я_параметра_1,
                тип ім'я_параметра_2,...)
{
    тіло функції
}
```

Передача значення з функції, що викликається, до функції, яка викликає, відбувається за допомогою оператора повернення return:

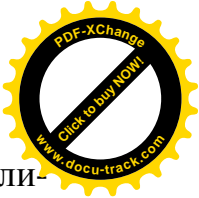
```
return вираз;
```

Таких операторів у підпрограмі може бути кілька, і тоді вони фіксують відповідні крапки виходу. Наприклад:

```
int f(int a, int b)
{
    if (a > b) { printf("max = %d\n", a);
                return a;
            }
    printf("max = %d\n", b); return b;
}
```

Викликати цю функцію можна так:

```
c = f(15, 5);
c = f(d, g);
f(d, g);
```



Після слова return можна нічого не записувати; тоді функції, що викликала, ніякого значення не передається. Керування передається цій функції у випадку входу "після закінчення" (остання закрита фігурна дужка).

### 3. Програма роботи

3.1. Записати запропонований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв'язання задачі мовою C++.

3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

### 4. Приклад

Написати функцію для обчислення  $x^m$ .

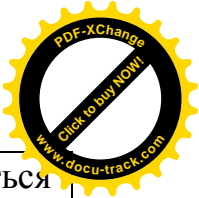
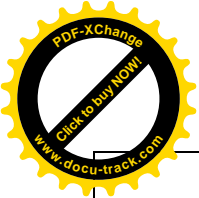
**Розв'язання.**

*Таблиця символічних імен для функції Step*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
r	r	Число у потрібному степені	long int	Результат
a	a	Число, яке потрібно обчислити	long int	Параметр функції
b	b	Степінь числа <i>a</i>	long int	Параметр функції

*Таблиця символічних імен основної програми*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення



x	x	Число, яке потрібно піднести до степеня $s$	long int	З'ясовується у процесі
s	s	Степінь числа $x$	long int	З'ясовується у процесі

**Текст програми мовою C++  
для компіляторів Visual C++**

```
#include <iostream.h>
long int Step(long int a, long int b) // початок функції Step
    // з параметрами a (число, яке потрібно обчислити) та b (ступінь числа)
{
    long int r;
    r = 1;
    if ( b > 0 ) // вираховується число в позитивній степені
    for( int i = 0; i < b; i++ )
    {
        r = r * a; // накопичення добутку b разів
    }
    else r = 0;
    return r; // повернення результату роботи функції
}
int main() // головна функція
{
    long int x, s;
    cout << "Уведіть число: ";
    cin >> x;
    cout << "Уведіть ступінь: ";
    cin >> s;
    cout << x << " в степені " << s << " = "
        << Step( x, s ) << endl; // виклик функції Step
        // безпосередньо у потоці виводу
    return 0;
}
```

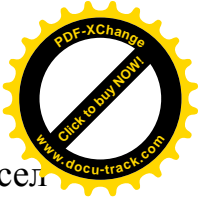
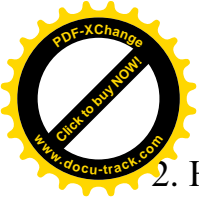
**5. Варіанти завдань**

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для реалізації запропонованих завдань з використанням функцій відповідно до заданого варіанта.

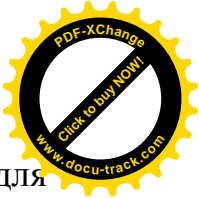
1. Написати функцію для обчислення знака числа за формулою:

$$sign(x) = \begin{cases} 1, & x > 0 \\ 0, & x = 0 \\ -1, & x < 0 \end{cases} \text{ . Знайти значення: } sign(-a).$$

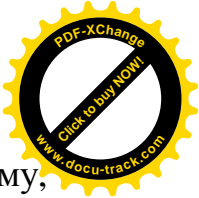


2. Написати функцію для обчислення  $n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$ . Знайти факторіали чисел від 1 до 7.
3. Написати функцію для обчислення кількості сполучень елементів  $C_n^m = \frac{n!}{m!(n-m)!}$ . Для перевірки роботи функції написати основну програму, що визначає  $C_2^4, C_3^6, C_2^6$ .
4. Написати функцію для обчислення суми  $n$  членів арифметичної прогресії за формулою  $S = \frac{(a_1 + a_n) \cdot n}{2}$ . Знайти суму арифметичної прогресії: 1, 2, 3, ..., 100.
5. Написати функцію для обчислення суми  $n$  членів геометричної прогресії за формулою  $S = \frac{a_1 \cdot (1 - q^n)}{1 - q}$ . Знайти суму геометричної прогресії: 1, 2, 4, 8, ..., 100.
6. Написати функцію для піднесення числа  $x$  до степеня  $m$  ( $m$  – натуральне число). Знайти  $x^2, x^4, x^5$ .
7. Написати функцію для обчислення арксинуса. Знайти  $\arcsin(0,9), \arcsin(0,1), \arcsin(-0,9), \arcsin(0,99)$ , за допомогою формули  $Y = \arctg\left(\frac{x}{\sqrt{1-x^2}}\right)$ .
8. Дано масив розміром  $7 \times 2$  з елементами дійсного типу. Написати програму для обчислення суми додатних елементів масиву, яку оформити у вигляді функції.
9. Написати функцію для обчислення арккосинуса. Знайти  $\arccos(0,5), \arccos(0,6), \arccos(-0,5), \arccos(0,99)$ , за допомогою формули  $Y = \arctg\left(\frac{\sqrt{1-x^2}}{x}\right)$ .
10. Дано масив розміром  $5 \times 2$  з елементами дійсного типу. Написати програму, що визначає наявність в масиві від'ємних елементів. Визначення цієї наявності оформити у вигляді функції.
11. Дано масив розміром  $2 \times 4$  з елементами дійсного типу. Написати програму, що визначає наявність у масиві хоча б одного елемента, що дорівнює нулю. Визначення цієї наявності оформити у вигляді функції.
12. Дано масив розміром  $4 \times 6$  з елементами цілого типу. Написати програму для обчислення добутку ненульових елементів масиву. Обчислення цього добутку оформити у вигляді функції.
13. Дано масив розміром  $6 \times 2$  з елементами дійсного типу. Написати програму для обчислення номера рядка, в якому міститься мінімальний елемент масиву. Обчислення номера цього рядка оформити у вигляді функції.





14. Дано масив розміром  $3 \times 5$  з елементами цілого типу. Написати програму для обчислення номера стовпця, в якому міститься мінімальний елемент масиву. Обчислення номера цього стовпця оформити у вигляді функції.
15. Дано масив розміром  $6 \times 2$  з елементами дійсного типу. Написати програму для обчислення  $M = \max - \min$ , де  $\max$  – максимальний за модулем елемент масиву, а  $\min$  – мінімальний за модулем елемент масиву. Обчислення значення  $M$  оформити у вигляді функції.
16. Написати функцію для обчислення тангенса. Знайти  $ctg(0,9)$ ,  $ctg(0,1)$ ,  $ctg(-0,9)$ , скориставшись формулою  $Y = \frac{\cos(x)}{\sin(x)}$ .
17. Дано масив розміром  $3 \times 3$  з елементами символьного типу. Написати програму для обчислення кількості елементів масиву, що не є цифрами. Обчислення кількості цих елементів оформити у вигляді функції.
18. Дано масив розміром  $4 \times 4$  з елементами цілого типу. Написати програму для обчислення добутку додатних елементів масиву. Обчислення цього добутку оформити у вигляді функції.
19. Написати функцію для обчислення тангенса. Знайти  $tg(0,9)$ ,  $tg(0,1)$ ,  $tg(-0,9)$ , скориставшись формулою  $Y = \frac{\sin(x)}{\cos(x)}$ .
20. Написати функцію для піднесення числа  $x$  до степеня 3, тобто обчислити  $x^3$ .
21. Дано масив розміром  $5 \times 5$  з елементами цілого типу. Написати програму, що визначає наявність у масиві додатних елементів. Визначення цієї наявності оформити у вигляді функції.



22. Дано масив розміром  $3 \times 4$  з елементами дійсного типу. Написати програму, що визначає кількість негативних елементів. Визначення цієї кількості оформити у вигляді функції.
23. Написати функцію для обчислення площі трикутника за формулою Герона:  $S = \sqrt{p(p-a)(p-b)(p-c)}$ , де  $a, b, c$  – сторони трикутника, а  $p$  – напівпериметр. Обчислити площу трикутників за такими вихідними даними:

Сторона $a$ , см	Сторона $b$ , см	Сторона $c$ , см
3.1	4.5	1.2
4.0	6.7	2.9

24. Написати функцію для обчислення третьої сторони трикутника за відомими двома сторонами і кутом між ними. Обчислити сторони трикутників за табличними даними. Для обчислень скористатися формулою  $a^2 = b^2 + c^2 - 2 \cdot b \cdot c \cdot \cos \alpha$ .

Сторона $a$ , см	Сторона $b$ , см	Кут $\alpha$ , рад
3	4	$\pi/2$
2	3	$\pi/4$

25. Написати функцію для обчислення об'єму кульового сектора  $V = \frac{2}{3}\pi R^2 H$ , де  $R$  – радіус сфери, а  $H$  – висота сектора. Обчислити об'єм кульових секторів за такими вихідними даними:

Радіус сфери $R$ , см	Висота сектора $H$ , см
6	4
7.5	4
12	3.2

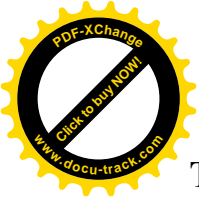
## ЛАБОРАТОРНА РОБОТА № 11 ФУНКЦІЇ З ПАРАМЕТРАМИ

### 1. Мета роботи

Дослідити роботу операторів функцій з параметрами мови програмування C++ , знати призначення, форму запису та особливості застосування функцій з параметрами. Засвоїти методику складання та налагодження програм із використанням функцій з параметрами на ПЕОМ.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанту. Налогодити та розв'язати задачу на ПЕОМ.



## Теоретичні відомості

У мові C++ аргументи функції можуть передаватися за значенням, тобто функція, що викликається, отримує свою тимчасову копію кожного аргументу, а не його адресу. Це означає, що функція не може змінити значення змінної у програмі, яка викликала цю функцію. Однак це легко зробити, якщо передавати у функцію не змінні, а їх адреси. Наприклад:

```
void swap(int *a, int *b)
{
    int tmp = *a;
    *a = *b;
    *b = tmp;
}
```

Виклик з програми функції `swap(&b, &c)` (тут підпрограмі передаються адреси змінних `b` і `c`) призведе до того, що значення змінних `b` і `c` поміняються місцями.

Коли як аргумент функції використовується ім'я масиву, то передається лише адреса початку масиву, а самі елементи не копіюються. Функція може змінювати елементи масиву, зсуваючись (індексуванням) від його початку.

Функції можна передати масив у вигляді параметра. Тут можливі три варіанти:

1. Параметр задається як масив (наприклад: `int m[100];`).
2. Параметр задається як масив без зазначення його розмірності (наприклад: `int m[];`).
3. Параметр задається як покажчик (наприклад: `int *m;`). Цей варіант використовується найчастіше.

Незалежно від вибраного варіанта викликуваної функції передається покажчик на початок масиву. Самі ж елементи масиву не копіюються.

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв'язання задачі мовою C++.

3.3. Розв'язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв'язати складену програму.

3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);



- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

#### 4. Приклад

Дано матрицю  $A(2,3)$ . Знайти максимальний і мінімальний елементи та створити функції для пошуку кожного з них. Уведення та виведення матриці виконати за допомогою функцій.

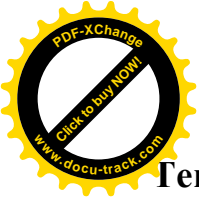
**Розв’язання.**

*Таблиця символічних імен основної програми*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
mat	mat	Матриця $[N][M]$	int	Уводиться
mx	mx	Максимальний елемент матриці	int	З'ясовується у процесі
mn	mn	Мінімальний елемент матриці	int	З'ясовується у процесі
N	N	Кількість стовпців	int	2
M	M	Кількість рядків	int	3

*Таблиця символічних імен для функції PoiskMaxMin*

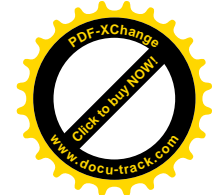
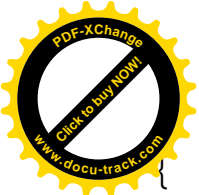
Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
x	x	Матриця із $[n][m]$ елементів	int	Параметр функції
n	n	Кількість стовпців у матриці	int	Параметр функції
m	m	Кількість рядків у матриці	int	Параметр функції
i	i	Параметр циклу	int	Змінне значення
j	j	Параметр циклу	int	Змінне значення
*max	*max	Максимальний елемент матриці	int	З'ясовується у функції
*min	*min	Мінімальний елемент матриці	int	З'ясовується у функції



## Текст програми мовою C++ для компіляторів Visual C++

```
#include <iostream.h>
const int N = 2, M = 3;
// функція вводу масиву
void Input(int x[N][M], int n, int m, char *name)
{
    for ( int i = 0; i < n; i++)
    {
        for ( int j = 0; j < m; j++ )
        {
            cout <<"Уведіть"<<name<<"["<<i<<"]["<<j<<"]";
            cin >> x[ i ][ j ];
        }
    }
}
// функція виводу масиву
void Output( int x[N][M], int n, int m, char *name )
{
    for ( int i = 0; i < n ; i++ )
    {
        for ( int j = 0; j < m; j++ )
        {
            cout << x[ i ][ j ] << '\t';
        }
        cout << endl;
    }
}
// функція пошуку максимального та мінімального елементів
void PoiskMaxMin( int x[N][M], int n, int m, int *max,
int *min )
{
    *max = x[0][0];
    *min = x[0][0];
    for ( int i = 0; i < n; i++ )
    {
        for ( int j = 0; j < m; j++ )
        {
            if ( *max < x[ i ][ j ] ) *max = x[ i ][ j
];
            if ( *min > x[ i ][ j ] ) *min = x[ i ][ j
];
        }
    }
}

int main() // головна функція
```



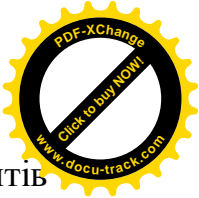
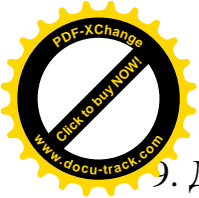
```
int mat[ N ][ M ];
int mx,mn;
Input(mat,N,M,"mat"); // виклик функції вводу масиву
cout << endl << endl;
Output(mat,N,M,"mat");// виклик функції виводу масиву
cout << endl << endl;
PoiskMaxMin( mat,N, M, &mx, &mn ); // виклик функції пошуку
// максимального та мінімального елементів
cout << "Максимальний елемент = " << mx << endl;
cout << "Мінімальний елемент = " << mn << endl;
return 0;
}
```

## 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму для розв'язання відповідної до заданого варіанта задачі з використанням функцій з параметрами,.

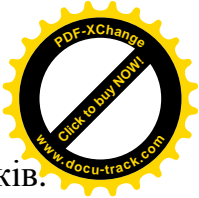
1. Дано матриці  $A(4,4)$  і  $Y(3,3)$ , для кожної з яких обчислити суму елементів над, під головною діагоналлю та на головній діагоналі. Уведення, виведення матриці та обчислення сум оформити у вигляді функцій.
2. Дано матриці  $A(4,4)$  і  $Y(5,5)$ , для кожної з яких знайти найбільший і найменший елементи головної діагоналі. Уведення, виведення матриці, пошук мінімального і максимального елементів оформити у вигляді функцій.
3. Дано матриці  $A(3,3)$ ,  $Y(5,5)$ ,  $C(4,4)$ . Для кожної з них визначити максимальні елементи у рядках і перемістити їх на головну діагональ. Уведення, виведення матриці та її перетворення оформити у вигляді функцій.
4. Дано матриці  $A(4,3)$  і  $Y(3,5)$ . Знайти середні геометричні позитивних елементів кожного їх рядка. Уведення, виведення матриці та пошук середніх геометричних оформити у вигляді функцій.
5. Дано матриці  $A(6,3)$  і  $Y(4,5)$ . Знайти найбільший елемент кожної з них і записати одиницю у той рядок і стовпець, в яких він розмішений. Уведення, виведення і перетворення матриці оформити у вигляді функцій.
6. Дано матриці  $A(3,3)$  і  $Y(5,5)$ , транспонувати їх. Уведення, виведення і транспонування матриці оформити у вигляді функцій.
7. Дано матриці  $A(6,2)$  і  $Y(4,5)$ . Знайти найбільший елемент кожної з них і номер рядка і стовпця, в яких він розміщений. Уведення, виведення матриці і пошук максимального елемента оформити у вигляді функцій.
8. Дано матриці  $A(4,4)$  і  $Y(3,4)$ . Знайти найменший елемент кожної з них і записати у рядок, де міститься цей елемент, '10'. Уведення, виведення матриці та її перетворення матриці оформити у вигляді функцій.



9. Дано матриці  $A(4,5)$  і  $Y(5,5)$ . Для кожної з них обчислити суму елементів рядків і визначити найменше значення цих сум і номер відповідного рядка. Уведення, виведення матриці і пошук номера рядка оформити у вигляді функцій.
10. Дано матриці  $A(3,3)$  і  $Y(4,5)$ , де визначити кількість додатних та від'ємних елементів. Уведення, виведення матриці і пошук кількості додатних та від'ємних елементів оформити у вигляді функцій.
11. Дано матриці  $A(5,2)$  і  $Y(3,2)$ . Для кожної з них знайти найбільші елементи кожного рядка і записати в одновимірний масив. Уведення, виведення матриці та формування одновимірного масиву оформити у вигляді функцій.
12. Дано матриці  $A(4,3)$ ,  $B(5,4)$ ,  $C(5,4)$  і два дійсних числа  $x$  і  $y$ . Знайти  $x*A$ ,  $y*B$ ,  $x*C+y*B$ . Уведення, виведення матриці, додавання, множення на число оформити у вигляді функцій.
13. Дано матриці  $A(4,4)$ ,  $Y(5,5)$ ,  $C(6,6)$ . Знайти транспоновані матриці  $A^m$ ,  $B^m$ ,  $C^m$ . Уведення, виведення, транспонування матриці оформити у вигляді функцій.
14. Дано матриці  $A(4,4)$ ,  $Y(3,3)$ ,  $C(5,5)$ . Для кожної з них обчислити суму елементів над, під головною діагоналлю та на головній діагоналі. Уведення, виведення матриці та обчислення сум оформити у вигляді функцій.
15. Дано матриці  $A(3,3)$ ,  $Y(5,5)$ ,  $C(4,4)$ . Знайти третю норму для кожної з них. Уведення, виведення матриці, обчислення норми оформити у вигляді функцій, скориставшись формулою:

$$\|A\|_{III} = \sqrt{\sum_{i=1}^n \sum_{k=1}^n a_{ik}^2}.$$

16. Дано матриці  $A(3,5)$ ,  $Y(5,2)$ ,  $C(4,6)$ , у кожній з них поміняти місцями перший та останій рядки. Уведення, виведення матриці, обмін рядків оформити у вигляді функцій.
17. Дано матриці  $A(4,5)$ ,  $Y(5,3)$ ,  $C(4,6)$ , у кожній з яких поміняти місцями перший і останній стовпці. Уведення, виведення матриці, обмін стовпців оформити у вигляді функцій.
18. Дано матриці  $A(3,5)$ ,  $Y(5,2)$ ,  $C(4,6)$ , у кожній з яких визначити місце розміщення максимального елемента. Уведення, виведення матриці, пошук місця оформити у вигляді функцій.
19. Дано матриці  $A(4,5)$ ,  $Y(5,3)$ ,  $C(4,6)$ . Утворити три нові матриці діленням кожного елемента вихідної матриці на середнє арифметичне її елементів. Уведення, виведення та зміну матриць оформити у вигляді функцій.
20. Дано матриці  $A(3,3)$ ,  $Y(5,5)$ ,  $C(4,4)$ , у кожній з яких поміняти місцями елементи на головній і побічній діагоналях. Уведення, виведення, перетворення матриці оформити у вигляді функцій.
21. Дано матриці  $A(3,3)$ ,  $Y(5,5)$ ,  $C(4,4)$ . Для кожної з них обчислити суму елементів головної діагоналі. Уведення, виведення матриці, обчислення суми елементів зазначеної діагоналі оформити у вигляді функцій.



22. Дано матриці  $A(3,5)$ ,  $Y(5,2)$ ,  $C(4,6)$ . Для кожної з них обчислити суми рядків. Уведення, виведення матриці, обчислення сум рядків, виведення вектора сум оформити у вигляді функцій.
23. Дано матриці  $A(4,5)$ ,  $Y(5,3)$ ,  $C(2,6)$ . Для кожної з них обчислити суми стовпців. Уведення, виведення матриці, обчислення сум стовпців, виведення вектора сум оформити у вигляді функцій.
24. Дано матриці  $A(3,3)$ ,  $Y(5,5)$ ,  $C(4,4)$ . Для кожної з них знайти максимальний і мінімальний елементи головної діагоналі. Уведення, виведення матриці, пошук мінімального та максимального елементів оформити у вигляді функцій.
25. Дано матриці  $A(3,5)$ ,  $Y(5,2)$ ,  $C(4,6)$ . Для кожної з них обчислити кількість додатних і від'ємних елементів. Уведення, виведення матриці, обчислення кількості названих елементів оформити у вигляді функцій.

---

## ЛАБОРАТОРНА РОБОТА № 12 РОБОТА З ТЕКСТОВИМИ ФАЙЛАМИ

---

### 1. Мета роботи

Вивчити та дослідити засоби алгоритмічної мови програмування C++, які використовують функції обробки текстових файлів, а також методику складання, налагодження та розрахунку програм.

### 2. Постановка задачі

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанту. Налаштувати та розв'язати задачу на ПЕОМ.

### Теоретичні відомості

У мові C++ визначено три класи файлового введення/виведення:

`ifstream` – вхідні потоки для читання з файла;

`ofstream` – вихідні потоки для запису в файл;

`fstream` – потоки для читання та запису файлів.

Під час роботи з файлами перелічених класів допустимо використовувати їх методи, а також операції “розмістити в потоці” (<<) та „ узяти з потоку” (>>).

Конструктори відповідних класів складають об'єкти потоків, що зв'язані з файлами. Наприклад, оператори

```
ofstream outfile("Test.dat");  
if (!outfile)
```





```
{ cout <<"Неможливо скласти файл";  
  return;  
}
```

становлять вихідний потік `outfile`, зв'язаний з файлом `"Test.dat"`, одночасно складаючи сам файл. Якщо операція неможлива, значення `outfile` дорівнює 0, а оператор `if` припиняє роботу.

Аналогічно можливе складання вхідного потоку, зв'язаного з файлом:

```
ifstream infile("Test.dat");  
if (!infile)  
{ cout <<"Неможливо відкрити файл";  
  return; }
```

Дані у файлі можна розмістити за допомогою потоків:

```
outfile << i << " " <<k << " " << s <<endl;
```

або ж із застосуванням методів `put` та `write`:

```
outfile.put('z') // записує у файл символ 'z';  
outfile.write(s,5) // записує у файл п'ять елементів символьного  
                    масиву s.
```

У класі `ifstream` також можливе використання як операцій „узяти з потоку” (`>>`), наприклад,

```
infile >> i1 >> i2 >> endl;
```

так і аналогічних методів `get` та `getline`, що мають свої модифікації:

```
infile.get() // уводить один символ із зазначеного потоку та  
             повертає його як значення виклику функції.
```

Повертає *EOF* у разі закінчення файла;

```
infile.getline(s,80) // читання з файла даних по рядках  
                    (s рядків по 80 символів).
```

Закрити файл можна за допомогою методу `close`:

```
outfile.close();  
infile.close().
```

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

– скласти таблицю символічних імен;



- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

### 3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

### 3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму(або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

## 4. Приклад

У текстовому файлі розмістити дані про радіуси 12 кіл. За допомогою формул обрахувати довжину та площі кіл. У новий файл записати такі дані: радіус, довжина та площа кола.

### Розв’язання.

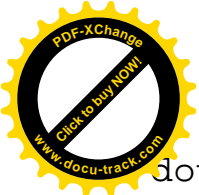
*Таблиця символічних імен*

Математичне ім'я	Ім'я у програмі	Смислове значення	Змінна	
			Тип	Значення
RadIn	RadIn	Файл для читання даних	ifstream	Файлові дані
NewFile	NewFile	Файл для введення результату	ofstream	Результат
i	i	Параметр циклу	int	Змінне значення
r	r	Радіус	double	Занесене у файл
l	l	Довжина	double	Результат
s	s	Площа	double	Результат
n	n	Порядковий номер кола	double	Результат

### Текст програми мовою C++

для компіляторів Visual C++

```
# include <iostream.h>
# include <ostream.h>
# include <fstream.h>
# include <iomanip.h>
void main()
{
```



```
double r,l,s,n;
fstream Radin("radius.txt",ios::out); // складаємо
// файл для запису

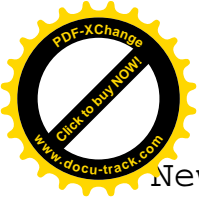
"radius.txt"
if (!Radin) // перевірка чи вдалося скласти файл
{ cout <<"Error file Radin"<<endl;
exit(1);
}

for (int i= 0; i<4; i++)
{
cout<<"Vvedite radius"<<i+1<<" kruga";
cin>>r; // Уведення даних про радіуси
Radin<<i+1<<" "<<r<<endl; // переписуємо дані у файл "radius.txt"
}
Radin.close; // закриваємо файл "radius.txt"
ifstream Radout("radius.txt",ios::in); // відкриваємо
// файл для читання з

"radius.txt"
if (!Radout) // перевірка чи вдалося відкрити файл
{ cout <<"Error file Radout"<<endl;
exit(1);
}
fstream NewFile ("newfile.txt",ios::out); // складаємо
// файл для запису "newfile.txt "

if (!NewFile) // перевірка чи вдалося скласти файл
{ cout <<"Error file NewFile"<<endl;
exit(1);
}
cout<<"Nomer\tRadius\tDlina\tPloshad"<<endl;
for(i=0;i<4;i++)
{

Radout>>n>>r; // зчитуємо дані з файлу "radius.txt"
l=2*3.14*r; // підраховуємо довжину кола
s=3.14*r*r; // підраховуємо площу кола
// вивід даних на екран:
cout<<setiosflags(ios::fixed|ios::showpoint)<<
setprecision(2)<<n<<'\t'<<r<<'\t'<<l<<'\t'<<s<<endl;
// запис даних у файл "newfile.txt ":
NewFile<<setiosflags(ios::fixed|ios::showpoint)<<
setprecision(2)<<n<<'\t'<<r<<'\t'<<l<<'\t'<<s<<endl;
}
Radout.close; // закриваємо файл "radius.txt"
```



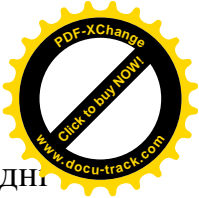
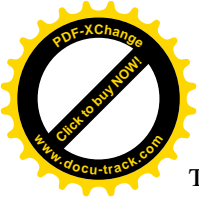
```
NewFile.close; // закриваємо файл "newfile.txt "  
}
```

## 5. Варіанти завдань

Кожен студент вибирає варіант завдання за номером у списку групи.

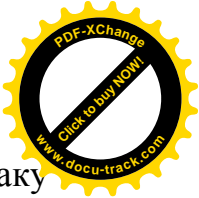
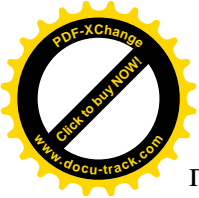
**Завдання.** Скласти схему алгоритму, таблицю символічних імен і програму для обробки текстового файла відповідно до заданого варіанта.

1. У текстовому файлі – дані про довжини сторін 20 прямокутних паралелепіпедів. Обчислити об'єми цих паралелепіпедів. Результати розрахунку записати у файл як таблицю, що містить стовпці: довжину першої, другої та третьої сторін паралелепіпеда, об'єм паралелепіпеда. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
2. У текстовому файлі – дані про довжини сторін 15 прямокутників. Обчислити периметри і площі цих прямокутників. Результати розрахунку записати у файл як таблицю, що містить стовпці: довжину першої та другої сторін прямокутника, периметр та площу прямокутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
3. У текстовому файлі – дані про радіуси 12 кіл. Обчислити довжину кіл і площу кругів, обмежених цими колами. Результати розрахунку записати у файл як таблицю, що містить стовпці: радіус і довжину кола, площу круга. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:  $l = 2 \cdot \pi \cdot R$ ,  $S = \pi \cdot R^2$ .
4. У текстовому файлі – дані про радіуси основ і висоти 12 циліндрів. Обчислити об'єм і площу бічної поверхні цих циліндрів. Результати розрахунку записати у файл як таблицю, що містить стовпці: радіус основи, висоту, об'єм та площу бічної поверхні циліндра. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:  $V = \pi \cdot R^2 \cdot h$ ,  $S_{\text{бічн}} = 2 \cdot \pi \cdot R \cdot h$ .
5. У текстовому файлі – дані про радіуси основ і висоти 15 конусів. Обчислити об'єм і площу бічної поверхні цих конусів. Результати розрахунку записати у файл як таблицю, що містить стовпці: радіус основи, висоту, об'єм та площу бічної поверхні конуса. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:  $V = \frac{1}{3} \pi \cdot R^2 \cdot h$ ,  $S_{\text{бічн}} = \pi \cdot R \cdot l$ .
6. У текстовому файлі – дані про катети 18 прямокутних трикутників. Обчислити гіпотенузи і площі цих трикутників. Результати розрахунку записати у файл як таблицю, що містить стовпці: перший та другий катети



трикутника, а також гіпотенузу та площу трикутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

7. У текстовому файлі – дані про довжини трьох сторін 14 трикутників. Обчислити площі і периметри цих трикутників. Результати розрахунку записати у файл як таблицю, що містить стовпці: довжину, другої та третьої сторін трикутника, площу та периметр трикутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
8. У текстовому файлі – дані про довжини основ і висоти 19 трапецій. Обчислити площі цих трапецій. Результати розрахунку записати у файл як таблицю, що містить стовпці: довжину нижньої та верхньої основи трапеції, висоту та площу трапеції. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
9. У текстовому файлі – дані про довжини сторін 15 рівносторонніх трикутників. Обчислити площі і периметри цих трикутників. Результати розрахунку записати у файл як таблицю, що містить стовпці: довжину сторони, периметр та площу трикутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
10. У текстовому файлі – дані про кількість сторін, довжини сторін і радіуси вписаних кіл для 14 правильних багатокутників. Обчислити площі і периметри цих багатокутників. Результати розрахунку записати у файл як таблицю, що містить стовпці: кількість сторін, довжину сторони, периметр та площу багатокутника, а також радіус вписаного у багатокутник кола. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:  $S = \frac{n}{2} a \cdot r$ .
11. У текстовому файлі – дані про довжини сторін і кути між ними для 18 паралелограмів. Обчислити площі і периметри цих паралелограмів. Результати розрахунку записати у файл як таблицю, що містить стовпці: довжину першої та другої сторін паралелограма, кут між сторонами, периметр та площу паралелограма. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
12. У текстовому файлі – дані про довжини сторін і кутів між ними для 18 паралелограмів. Вибрати паралелограми, що є прямокутниками, і записати у файл таблицю, що містить таку інформацію: довжину першої та другої сторін прямокутника, периметр та площу прямокутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.
13. У текстовому файлі – дані про кількість сторін, довжини сторін і радіуси вписаних кіл для 14 правильних багатокутників. Вибрати багатокутники з



парною кількістю сторін і записати у файл таблицю, що містить таку інформацію: кількість сторін багатокутника, довжину його сторони, радіус вписаного в нього кола, периметр та площу багатокутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:  $S = \frac{n}{2} a \cdot r$ .

14. У текстовому файлі – дані про довжини сторін 15 рівносторонніх трикутників. Вибрати трикутники зі стороною, більшою одиниці і меншою трьох, і записати у файл таблицю, що містить таку інформацію: довжину сторони трикутника, периметр та площу трикутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

15. У текстовому файлі – дані про довжини основ і висоти 19 трапецій. Вибрати трапеції з висотою, меншою верхньої основи, і записати у файл таблицю, що містить таку інформацію: довжину нижньої та верхньої основ трапеції, висоту та площу трапеції. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

16. У текстовому файлі – дані про довжини трьох сторін 14 трикутників. Вибрати рівносторонні трикутники і записати у файл таблицю, що містить таку інформацію: довжину сторони трикутника, площу та периметр трикутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

17. У текстовому файлі – дані про катети 18 прямокутних трикутників. Вибрати рівнобічні трикутники і записати у файл таблицю, що містить таку інформацію: катет, гіпотенузу, та площу трикутника. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

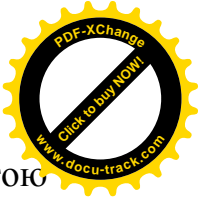
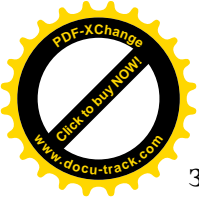
18. У текстовому файлі – дані про радіуси основ і висоти 15 конусів. Вибрати конуси, радіус основи яких більше висоти, і записати у файл таблицю, що містить таку інформацію: радіус основи, висоту, об'єм конуса та площу його бічної поверхні. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:

$$V = \frac{1}{3} \pi \cdot R^2 \cdot h, S_{\text{бічн}} = p \cdot R \cdot l.$$

19. У текстовому файлі – дані про радіуси основ і висоти 12 циліндрів. Вибрати циліндри, радіус основи яких дорівнює висоті, та записати у файл таблицю, що містить таку інформацію: радіус основи, висоту, об'єм циліндра та площу його бічної поверхні. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:

$$V = \pi \cdot R^2 \cdot h, S_{\text{бічн}} = 2 \cdot \pi \cdot R \cdot h.$$

20. У текстовому файлі – дані про радіуси 12 кіл. Вибрати кола, довжина яких більше семи, але менше 10, і записати у файл таблицю, що містить таку інформацію: радіус і довжину кола та площу круга. Записати відповідні



заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора. Указівка:  $l = 2 \cdot \pi \cdot R$ ,  $S = \pi \cdot R^2$ .

21. У текстовому файлі – дані про довжини сторін 15 прямокутників. Вибрати прямокутники, що є квадратами, і записати у файл таблицю, що містить таку інформацію: довжину сторони, периметр та площу квадрата. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

22. У текстовому файлі – дані про довжини сторін 20 прямокутних паралелепіпедів. Вибрати паралелепіпеди, що є кубами, і записати у файл таблицю, що містить таку інформацію: довжину сторони та об'єм куба. Записати відповідні заголовки для стовпців. Файл із вихідними даними сформувати за допомогою текстового редактора.

23. У текстовий файл записані масиви  $A(2 \times 2)$ ,  $B(2 \times 2)$ ,  $C(3 \times 3)$  і  $D(3 \times 3)$ . Обчислити  $\frac{\det(A) + \text{Tr}(C)}{\text{Tr}(B)\text{Tr}(D) + \det(B)}$ ,

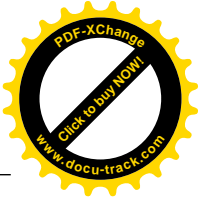
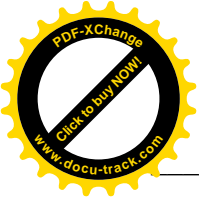
де  $\det$  – визначник ( $\det(A) = a_{11}a_{22} - a_{21}a_{12}$ );

$\text{Tr}$  – слід матриці ( $\text{Tr}(A) = \sum_{i=1}^N a_{ij}$ ).

Обчислення визначника і сліду матриці оформити за допомогою функцій.

24. У текстовий файл записані матриці  $A(5 \times 3)$ ,  $B(2 \times 6)$ ,  $C(4 \times 4)$  і  $D(3 \times 5)$ . Дописати в цей файл кількість додатних і від'ємних елементів масивів. Для виведення масивів та обчислення кількості зазначених елементів застосувати функції.

25. У текстовий файл записані матриці  $A(4 \times 4)$ ,  $B(3 \times 3)$  і  $C(5 \times 5)$ . Додати до діагональних елементів кожної з них значення мінімального за модулем ненульового елемента масиву. Отримані в такий спосіб матриці вивести в інший текстовий файл. Для введення, виведення і додавання до діагональних елементів числа застосувати функції, а також використати функцію при визначенні зазначеного мінімального елемента.



# ЛАБОРАТОРНА РОБОТА № 13

## СТРУКТУРИ

---

### 1. Мета роботи

Вивчити та дослідити засоби мови програмування C++, які використовують для роботи зі структурами, а також методику складання та налагодження програм.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налагодити та розв'язати задачу та на ПЕОМ.

#### Теоретичні відомості

Структура – це об'єднання одного чи більше об'єктів (змінних, масивів, покажчиків, інших структур і т.п.), які можуть не належати до одного типу.

Визначення структури формально має вигляд

```
struct тип {тип елемента_1 ім'я елемента_1; // список
                                                    елементів;
           .....
           тип елемента_n ім'я елемента_n};
```

Наприклад:

```
struct date {int day;
             int month;
             int year; };
```

Слідом за фігурною дужкою, що завершує список елементів, можна розмістити змінні зазначеного типу, наприклад:

```
struct date {...} a, b, c;
```

Дозволяється вкладати структури одна в одну, наприклад:

```
struct man {char name[20], fam[20];
            struct date bd;
            int age; };
```

Визначимо змінні, значення яких належать уведеному типу:

```
struct man man_[100]; // визначено масив man_, що складається
                       з 100 структур типу man.
```

Щоб звернутися до окремого елемента структури, необхідно назвати її ім'я, поставити крапку і написати ім'я потрібного елемента, наприклад:

```
man_[j].bd.day = 22;
man_[i].age = 19;
man_[j].bd.year = 1982;
```

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.





Ознайомитись з рекомендованою літературою та конспектом лекцій.

### 3.2. Програмування задачі:

- скласти таблицю символічних імен;
- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

### 3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

### 3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

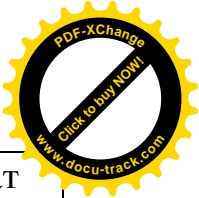
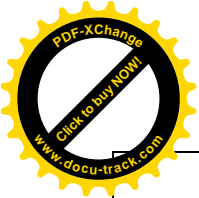
## 4. Приклад

Написати програму для опрацювання інформації про співробітників фірми (прізвище, рік народження, стать, термін роботи у фірмі, заробітна плата). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список осіб, молодших за 25 років.

### Розв’язання.

*Таблиця символічних імен*

Математичне ім’я	Ім’я у програмі	Смислове значення	Змінна	
			Тип	Значення
i	i	Параметр циклу	int	Змінне значення
rik	rik	Поточний рік	int	Уводиться з клавіатури
sotr	sotr	Змінна структурного типу <i>Inform</i>	Inform	Результат
fio	Fio	Змінна структурного типу <i>Inform</i> . Прізвище	char	Результат
year	year	Змінна структурного типу <i>Inform</i> . Рік народження	int	Результат
pol	pol	Змінна структурного типу <i>Inform</i> . Стать.	char	Результат
sokrab	sokrab	Змінна структурного типу <i>Inform</i> . Термін роботи.	int	Результат



zarpl	zarpl	Змінна структурного типу <i>Inform</i> . Заробітна плата	int	Результат
-------	-------	--	-----	-----------

### Текст програми мовою C++

для компіляторів Visual C++

```
#include <iostream.h>
```

```
struct Inform // визначення структури
```

```
{  
    char fio[40];  
    int year;  
    char pol;  
    int srokrab;  
    int zarpl;  
};
```

```
const int N = 2;
```

```
int main()
```

```
{  
    Inform sotr[N];  
    int i, rik;  
    // Уведення інформації про співробітників:  
    cout << " Інформація про співробітників:" << endl;  
    for ( i = 0; i < N; i++ )  
    {  
        cout << "Уведіть прізвище: ";  
        cin >> sotr[i].fio;  
        cout<<"Уведіть рік народження"<<sotr[i].fio<<": ";  
        cin >> sotr[i].year;  
        cout << "Уведіть стать" << sotr[i].fio << ": ";  
        cin >> sotr[i].pol;  
        cout << "Уведіть стаж" << sotr[i].fio << ": ";  
        cin >> sotr[i].srokrab;  
        cout << "Уведіть зарплату" << sotr[i].fio << ": ";  
        cin >> sotr[i].zarpl;  
    }  
    cout<< "Уведіть поточний рік"<<endl;  
    cin>> rik;  
    for ( i = 0; i < N; i++ )  
    {  
        if ((rik-sotr[i].year)<25)  
        {  
            cout << sotr[i].fio<<endl;  
            cout << sotr[i].year<<endl;  
            cout << sotr[i].pol<<endl;  
            cout << sotr[i].srokrab<<endl;  
            cout << sotr[i].zarpl<<endl;  
        }  
    }  
}
```



```
}  
return 0;  
}
```

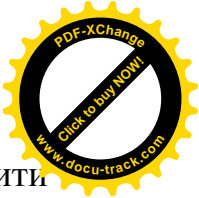
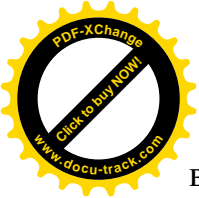
---

## 5. Варіанти завдань

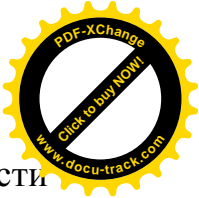
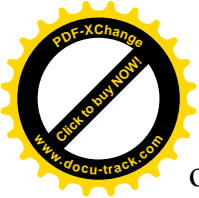
Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти таблицю символічних імен та програму для опрацювання інформації з допомогою структури відповідно до заданого варіанта.

1. Написати програму для опрацювання інформації про товари, що зберігаються на складі (назва, вартість, країна-виробник, кінцевий термін реалізації, кількість). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список товарів із терміном реалізації, що минає (менше місяця).
2. Написати програму для опрацювання інформації про паспортні дані громадян (прізвище, ім'я, по батькові, дата народження, серія та номер паспорта, дата видачі, ким виданий). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список осіб, у паспорт яких необхідно вклеїти нове фото.
3. Написати програму для опрацювання інформації про країни світу (назва країни, площа, кількість населення, державна мова, грошова одиниця). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список країн, густина населення в яких перевищує зазначену користувачем.
4. Написати програму для опрацювання інформації про продукти харчування (назва продукту, калорійність продукту, вміст у продукті вітамінів (А, В, С, Е, Р, РР), вартість 1 кг продукту). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список продуктів, що містять зазначені користувачем вітаміни.
5. Написати програму для опрацювання інформації про керування поїздів (номер поїзда, станція відправлення та призначення, час відправлення та прибуття, тип поїзда (пасажирський, швидкий). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список поїздів, що прибувають до потрібної станції.
6. Написати програму для опрацювання інформації про кішок, поданих на виставку (кличка, порода, вік). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список тварин заданої породи та віку.
7. Написати програму для опрацювання інформації про книги (автор, назва, видавництво, рік видання). Програма має надавати користувачеві можливість



- вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран перелік книг потрібного автора, виданих до 2009 р.
8. Написати програму для опрацювання інформації про пасажирів (паспортні дані пасажирів, номер рейсу, вага багажу). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список пасажирів, багаж яких перевищує 10 кг.
  9. Написати програму для опрацювання інформації про студентів (паспортні дані студента, група, стать, екзаменаційні оцінки з вищої математики, програмування та фізики). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список студентів чоловічої статі, в яких оцінка з програмування нижча від середньої оцінки з вищої математики.
  10. Написати програму для опрацювання інформації про навчальні справи студентів (прізвище, ім'я та по батькові студента, рік народження, стать, факультет, спеціальність, середній бал за семестр). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список студентів жіночої статі, для яких середній бал за семестр більше заданого.
  11. Написати програму для опрацювання інформації про автомобілі, що здаються на прокат (марка автомобіля, рік випуску, кілометраж). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список автомобілів з заданими роком випуску та довжиною кілометражу.
  12. Написати програму для опрацювання інформації про матричні принтери (назва принтера, фірма-виробник, швидкість друку, розмір буфера, габарити: висота, ширина, довжина, вага). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список принтерів з швидкістю друку вище середньої.
  13. Написати програму для опрацювання інформації про текстові процесори (назва, вартість, максимальна кількість вікон, пошук помилок, багатошпальтовий набір, фоновий друк). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список текстових процесорів, що задовольняють заданим вимогам користувача.
  14. Написати програму для опрацювання інформації про твори мистецтва, виставлені на аукціон (номер лота, назва твору, дані про автора, матеріал, початкова вартість). Програма має надавати користувачеві можливість вводити дані з клавіатури і заносити їх у файл; читати дані з файла і виводити на екран список лотів, вартість яких не більше заданої.
  15. Написати програму для ведення обліку торгових операцій фірми. Про кожну таку операцію мають зберігатися дані: реєстраційний номер, суть угоди,

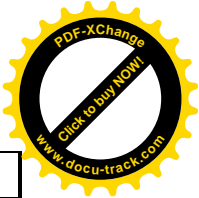


обсяг, форма розрахунку (готівкою, безготівковий, бартер), дата. Ввести інформацію про декілька угод (вибрати самостійно) і вивести на екран усі безготівкові торгові операції.

16. На автовокзалі зберігається інформація про автобусні рейси: номер рейсу, пункти відправлення і призначення, час прибуття і відправлення, тип автобуса (ПАЗ, ЛАЗ, ЛиАЗ, Икарус). Для кожного типу автобуса відома його пасажиромісткість (у спеціальному масиві). Увести інформацію про декілька рейсів (вибрати самостійно) і вивести для зазначеного пункту призначення максимальну кількість пасажирів, які прибувають туди всіма рейсами протягом дня.
17. У курортному готельному господарстві зберігається інформація про вільні номери в готелях: клас (двомісний, одномісний люкс, двокімнатний люкс і т.д.), кількість номерів, вартість номера за добу, назва готелю ("Зірочка", "Дружба", "Морський", "Прибій", "Турист"). Вивести за вимогою клієнта список готелів, в яких є потрібна кількість номерів необхідного класу.
18. Бібліотека має каталог усіх книг, що видаються на абонементі. У цьому списку про кожну книгу зберігається така інформація: автор, назва, обсяг (у сторінках), кількість примірників, жанр (навчальні посібники, класична художня література, сучасний детектив, сучасна фантастика). Надати можливість перегляду літератури певного жанру, а також вивести список навчальних посібників обсягом понад 200 сторінок.
19. У прайс-листі фірми записана інформація про комп'ютери, запропоновані до продажу: тип процесора (i80386DX, i80486SX, i80486DX, i80486DX2, i80486DX4, Pentium), тактова частота, об'єм оперативної пам'яті, об'єм кеш-пам'яті, ціна (\$). Задовольняючи запити покупця, видати список комп'ютерів, які він у змозі придбати.
20. У ЕОМ аеропорту зберігається інформація про рейси: номер рейсу, пункт призначення, тип літака (ЯК-40, ТУ-134, ТУ-154, АН-24, БОІНГ-747), час відльоту і прильоту, тривалість польоту. Вивести на екран загальну тривалість польотів для відповідного типу літаків.
21. В одному з чорноморських портів база даних зберігає інформацію про рейси пасажирських суден: порт призначення, дата і час відплиття, назва судна, порт приписки (Одеса, Новоросійськ, Ялта, Анапа, Севастополь, Херсон, Миколаїв). Написати програму для виведення на екран повної інформації про рейси до потрібного порту.
22. Учитель географії створив базу даних, з переліком деяких найбільших природних прісних озер світу з такими відомостями: назва озера, континент, країна, площа, найбільша глибина. Вивести на екран озера, що зацікавили Вас діапазоном глибин.

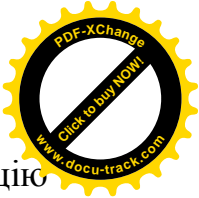
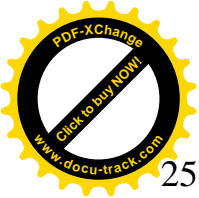
Приклад бази даних:

Озеро	Континент	Країна	Площа, кв.км	Глибина, м
Ладозьке	Європа	Росія	18100	225
Онезьке	Європа	Росія	9720	127



Венерн	Європа	Швеція	5550	100
Веттерн	Європа	Швеція	1900	119
Біле	Європа	Росія	1290	6
Віг	Європа	Росія	1250	130
Меларен	Європа	Швеція	1140	64
Пяйянне	Європа	Фінляндія	1065	93
Інарйярві	Європа	Фінляндія	1050	60
Верхне	Північна Америка	Канада, США	82400	393
Гурон	Північна Америка	Канада, США	59600	208
Мічиган	Північна Америка.	США	58000	281
Велике Ведмеже	Північна Америка	Канада	30200	137
Велике Невільничне	Північна Америка	Канада	28600	156
Ері	Північна Америка	США	52700	64
Вінніпег	Північна Америка	Канада	24300	28
Онтаріо	Північна Америка	Канада, США	19500	236
Буенос-Айрес	Південна Америка	Чилі, Аргентина	2400	145
Вікторія	Африка	Танзанія, Кенія	68000	80
Танганьіка	Африка	Танзанія, Заір	34000	1470
Ньяса	Африка	Мозамбік, Танзанія	30800	706
Мверу	Африка	Замбія, Заір	5200	15

23. Написати програму для реєстрації собак, що беруть участь у виставці. Реєстраційна картка включає дані: порода (вівчарка, бульдог, бультер'єр, спанієль, пудель), кличка, стать, вік. Вивести на екран повну інформацію про тих собак, що цікавлять користувача.
24. В автосалоні підготовлено список пропонованих до продажу автомобілів. Про кожен з них міститься така інформація: марка ("Жигулі", "Волга", "Москвич", "Запорожець", Mercedes, Opel, BMW, Ford, Toyota), модель, ціна (\$), колір. Вивести на екран марки машин, що відповідають фінансовим можливостям покупця.



25. Перед проведенням вітрильної регати оргкомітет здійснює реєстрацію екіпажів, кожен з яких повідомляє про назву яхти, її клас (Фін, Зоряний, Торнадо, Солинг), кількість членів екіпажу, порт приписки. Після завершення регати в базу даних для кожної яхти заноситься результат (місце серед суден даного класу). Вивести на екран повну інформацію про яхти для кожного класу за результатами змагань.

---

## ЛАБОРАТОРНА РОБОТА № 14 ДИНАМІЧНІ МАСИВИ

---

### 1. Мета роботи

Вивчити та дослідити засоби мови програмування C++ із застосуванням динамічної пам'яті, а також методики складання та налагодження програм для обробки матриць на ПЕОМ з використанням динамічної пам'яті.

### 2. Постановка завдання

Скласти схему алгоритму, таблицю символічних імен та програму мовою C++ відповідно до заданого варіанта. Налаштувати та розв'язати задачу на ПЕОМ.

#### Теоретичні відомості

Вільна пам'ять у програмах мовою C++ виділяється за допомогою операції `new`, що застосовується до специфікації типу (його абстрактного імені). При цьому буде виділено пам'ять, достатню для розміщення об'єкта такого типу, буде повернено покажчик на виділену пам'ять, наприклад:

```
int *pi = new int; // виділена пам'ять для об'єкта типу int. Тип поверненого значення "покажчик на int".
```

Масив виділяється у вільній пам'яті за допомогою наступного за специфікацією типу розмірності, який розміщений в квадратних дужках. Розмірність може бути виразом довільної складності. Операція `new` повертає покажчик на перший елемент масиву. Наприклад :

```
int i = 100;  
char *ps = new char[i*2]; // ps указує на масив з 200 елементами типу char.
```

Надалі доступ до елементів динамічного масиву здійснюється як до звичайного: `ps[n]`, де `n` – номер елемента.

Пам'ять, виділена за допомогою операції `new`, буде зайнята доти, поки програміст її повністю не звільнить. Для такого звільнення цієї пам'яті використовується операція `delete`, яка застосовується до покажчика, що адресує динамічний об'єкт. Наприклад :



```
int *pi = new int; // пам'ять відведена;
delete pi; // пам'ять звільнена.
```

Для звільнення пам'яті з-під масиву необхідно вставляти пару порожніх квадратних дужок між delete і покажчиком :

```
int *pia = new int[100]; // відведення пам'яті для
                        // одновимірного масиву
delete [] pia; // звільнення пам'яті з одновимірного масиву
```

Відведення пам'яті для двовимірного масиву відбувається наступним чином:

```
int **mas = new int* [m] // виділена пам'ять для покажчиків на
                        // m стовпців, результат - покажчик на
покажчик
for (int i = 0; i<m; i++)
    pmass[i]=new int [n]; // виділена пам'ять для елементів масиву
                        // типу int з кількістю рядків n
```

Доступ до елементів масиву звичайний: pmass[k][l], де k – номер елемента в рядку, l – номер елемента в стовпці.

```
delete pmass; // звільнення пам'яті з двовимірного масиву
```

Для відведення пам'яті також використовуються функції malloc() і calloc().

Функція malloc() приймає один параметр – розмір блока пам'яті, що виділяється, у байтах і повертає покажчик на виділений блок пам'яті. Тип покажчика – void\*, тому перед його використанням слід чітко позначати тип. Функція calloc() приймає два параметри – число елементів і розмір елемента та ініціалізує виділену пам'ять нулями. Функція повертає – void\*.

Приклади :

```
int *pi =(int*)malloc(sizeof(int)); // пам'ять для одного int;
int *pia2 =...
(int*)malloc(size*sizeof(int)); //для масиву int розміром
                                // size int *pia2 =...;
(int*)calloc(size, sizeof(int)); // те ж саме з ініціалізацією
                                // нулями.
```

Для звільнення пам'яті, відведеної malloc() або calloc(), використовують функцію free(), в якій один параметр: покажчик на пам'ять, яку потрібно звільнити. Він може бути будь-якого типу. Наприклад :

```
free(pia2).
```

### 3. Програма роботи

3.1. Записати запропанований викладачем варіант завдання з п. 5.

Ознайомитись з рекомендованою літературою та конспектом лекцій.

3.2. Програмування задачі:

– скласти таблицю символічних імен;





- скласти схему алгоритму або псевдокод (за вказівкою викладача);
- скласти програму розв’язання задачі мовою C++.

### 3.3. Розв’язання задачі на ПЕОМ.

На ПЕОМ набрати, відлагодити та розв’язати складену програму.

### 3.4. Оформлення звіту.

Звіт повинен містити:

- номер лабораторної роботи та її назву;
- умову задачі та дані свого варіанта;
- таблицю символічних імен;
- схему алгоритму (або псевдокод);
- лістинг (або рукопис) програми;
- результати розрахунку задачі;
- висновки.

## 4. Приклад

Дано матрицю. Визначити середнє арифметичне її елементів. Розмір матриці вводити з клавіатури, а елементи розмістити в пам’яті динамічно.

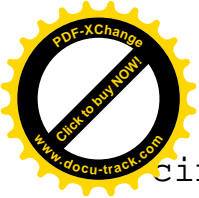
**Розв’язання.**

*Таблиця символічних імен*

Математичне ім’я	Ім’я у програмі	Смислове значення	Змінна	
			Тип	Значення
N	n	Кількість рядків у масиві	int	Уводиться з клавіатури
M	m	Кількість стовпців у масиві	int	Уводиться з клавіатури
S	S	Сума елементів масиву, середнє арифметичне елементів	float	З’ясовується у процесі
**pmas	**pmas	Показчик на показчик	int	З’ясовується у процесі
pmas	pmas	Масив	int	Уводиться з клавіатури
J	j	Параметр циклу	int	Змінна
I	i	Параметр циклу	int	Змінна

**Текст програми мовою C++ для компіляторів Borland C**

```
# include <iostream.h>
int main ()
{
int n,m;
cout<<"Vvedite kolichestwo strok";
```



```
cin>>n;
cout<<"Vvedite kolichestvo stolbzo\n";
cin>>m;
int **pmas=new int* [m]; // виділяємо пам'ять для m стовпців
float S=0;
for (int i = 0; i<m; i++)
    pmas[i]=new int [n]; // виділяємо пам'ять для елементів
                        // типу int в n строках
for ( i = 0; i<n; i++)
for ( int j =0;j<m; j++) // уведення масиву
{ cout<<"Wvedite pmas["<<i<<" , "<<j<<" ]=";
  cin>>pmas[i][j];
  S+=pmas[i][j]; // накопичення суми елементів
}

for ( i = 0; i<n; i++)
for ( int j =0;j<m; j++) // вивід масиву на екран
{ cout<<"  pmas["<<i<<" , "<<j<<" ]="<<pmas[i][j];
}
S=S/(n*m); // підрахунок середньо-арифметичного значення
cout<<"S="<<S;
delete pmas; // звільнення пам'яті для масиву
return 0;
}
```

## 5. Варіанти завдань

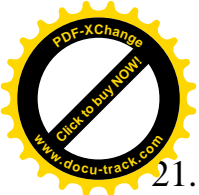
Кожен студент вибирає варіант завдання за номером у списку групи.

**Завдання.** Скласти схему алгоритму, таблицю символічних імен та програму обробки динамічного масиву відповідно до заданого варіанта.

1. Визначити добуток позитивних елементів кожного стовпця матриці A, її розмір вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
2. Визначити добуток елементів парних стовпців матриці, її розмір вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
3. Визначити суму елементів матриці, які за модулем перевищують одиницю, у кожному парному стовпці і кількість таких елементів. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
4. Визначити суму позитивних елементів розміщених над головною діагоналлю. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
5. Визначити кількість рядків, які містять нульові елементи матриці, та їх номери. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.



6. Визначити значення максимального елемента кожного рядка матриці та номер стовпця, в якому він розміщений. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
7. Максимальний і мінімальний елементи матриці поміняти місцями. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
8. Визначити мінімальний і максимальний елементи матриці та їх добуток. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
9. Визначити суму негативних елементів матриці, розташованих під головною діагоналлю. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
10. Визначити максимальний елемент матриці на головній діагоналі та суму елементів цієї діагоналі. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
11. Визначити для кожного непарного рядка кількість нульових елементів. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
12. Знайти середнє геометричне позитивних елементів кожного рядка матриці, її розмір вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
13. Знайти найбільший елемент матриці та записати одиницю в той рядок і стовпець, на перехресті яких він міститься. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
14. З матриці  $X$  побудувати матрицю  $Y$ , помінявши місцями рядки і стовпці. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
15. Знайти найбільший елемент матриці і номер рядка і стовпчика, у яких він міститься. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
16. Знайти найменший елемент матриці та записати в рядок, де міститься цей елемент, '10'. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
17. Обчислити суму елементів кожного рядка матриці, визначити найменше значення цих сум і номер відповідного рядка. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
18. Визначити кількість додатних і від'ємних елементів матриці. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
19. Знайти найбільші елементи кожного рядка матриці  $X$  і записати їх у масив  $Y$ . Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
20. Обчислити суму елементів матриці, розташованих над головною діагоналлю. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.



21. Перемножити матриці  $A(N \times M)$  і  $B(N \times M)$ . Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
22. Знайти найбільший елемент головної діагоналі матриці і вивести на екран увесь рядок, у якому він міститься. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
23. Визначити максимальний елемент на головній діагоналі та добуток позитивних елементів зазначеної діагоналі. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
24. Визначити добуток елементів парних стовпців матриці, її розмір вводити з клавіатури, а елементи розмістити в пам'яті динамічно.
25. Знайти суму елементів матриці, що мають задану різницю індексів  $i-j=k$ . Число  $k$  може бути і негативним. Розмір матриці вводити з клавіатури, а елементи розмістити в пам'яті динамічно.



## ДОДАТОК 1

### Коди ASCII (0-127)

(American Standard Code for Information Interchange)

DEC	CHAR	Name	DEC	CHAR	DEC	CHAR	DEC	CHAR
0	Ctrl-@	NUL	32	SPC	64	@	96	'
1	Ctrl-A	BOH	33	!	65	A	97	a
2	Ctrl-B	STX	34	"	66	B	98	b
3	Ctrl-C	ETX	35	#	67	C	99	c
4	Ctrl-D	EOT	36	\$	68	D	100	d
5	Ctrl-E	ENQ	37	%	69	E	101	e
6	Ctrl-F	ACK	38	&	70	F	102	f
7	Ctrl-G	BEL	39	'	71	G	103	g
8	Ctrl-H	BS	40	(	72	H	104	h
9	Ctrl-I	HT	41	)	73	I	105	i
10	Ctrl-J	LF	42	*	74	J	106	j
11	Ctrl-K	VT	43	+	75	K	107	k
12	Ctrl-L	PF	44	,	76	L	108	l
13	Ctrl-M	CR	45	-	77	M	109	m
14	Ctrl-N	SO	46	.	78	N	110	n
15	Ctrl-O	SI	47	/	79	O	111	o
16	Ctrl-P	DLE	48	0	80	P	112	p
17	Ctrl-Q	DC1	49	1	81	Q	113	q
18	Ctrl-R	DC2	50	2	82	R	114	r
19	Ctrl-S	DC3	51	3	83	S	115	s
20	Ctrl-T	DC4	52	4	84	T	116	t
21	Ctrl-U	NAK	53	5	85	U	117	u
22	Ctrl-V	SYN	54	6	86	V	118	v
23	Ctrl-W	ETB	55	7	87	W	119	w
24	Ctrl-X	CAN	56	8	88	X	120	x
25	Ctrl-Y	EM	57	9	89	Y	121	y
26	Ctrl-Z	SUB	58	:	90	Z	122	z
27	Ctrl-[	ESC	59	;	91	[	123	(
28	Ctrl-\	FS	60	<	92	\	124	
29	Ctrl-]	GS	61	=	93	]	125	)
30	Ctrl-^	RS	62	>	94	^	126	~
31	Ctrl-_	US	63	?	95	_	127	DEL

## ДОДАТОК 2



## Функції для роботи з рядками

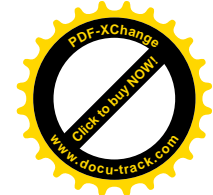
**string.h** — заголовний файл стандартної бібліотеки мови Сі, що містить функції для роботи з нуль-термінованими рядками і різноманітними функціями роботи з пам'ятю.

<b>Им'я</b>	<b>Примітка</b>
NULL	розширюється в <a href="#">null pointer</a> ; тобто, значення, що гарантовано не є валідною адресою об'єкта в пам'яті.
size_t	беззнакове ціле, що має той же тип, що і результат оператора sizeof.

### Функції

Им'я	Примітка
void * <a href="#">memcpy</a> (void *dest, const void *src, size_t n);	копіює n байт з області пам'яті src в dest, які не повинні перетинатися, інакше результат не визначений (можливо як правильне копіювання, так і ні)
void * <a href="#">memmove</a> (void *dest, const void *src, size_t n);	копіює n байт з області пам'яті src в dest, які на відміну від memcpy можуть перетинатися
void * <a href="#">memchr</a> (const void *s, char c, size_t n);	повертає вказівник на перше вхождення c в перших n байтах s, або NULL, якщо не знайдено
int <a href="#">memcmp</a> (const void *s1, const void *s2, size_t n);	порівнює перші n символів в областях пам'яті
void * <a href="#">memset</a> (void *, int z, size_t);	заповнює область пам'яті одним байтом z
char * <a href="#">strcat</a> (char *dest, const char *src);	дописує строку src в кінець dest
char * <a href="#">strncat</a> (char *dest, const char *, size_t);	дописує не більше n початкових символів строки src (або всю src, якщо її довжина менша) в кінець dest
char * <a href="#">strchr</a> (const char *, int);	шукає символ в рядку, починаючи з голови і повертає його адресу, або NULL якщо не знайдено
char * <a href="#">strrchr</a> (const char *, int);	шукає символ в рядку, починаючи з хвоста і повертає його адресу, або NULL якщо не знайдено
int <a href="#">strcmp</a> (const char *, const char *);	лексикографічне порівняння рядків
int <a href="#">strncmp</a> (const char *, const char *, size_t);	лексикографічне порівняння перших n байтів рядків
char * <a href="#">strcpy</a> (char *toHere, const char *fromHere);	копіює строку из одного места в другое
char * <a href="#">strncpy</a> (char *toHere, const char *fromHere, size_t n);	копіює до n байт рядка з одного місця в інше
size_t <a href="#">strlen</a> (const char *);	повертає довжину рядка
size_t <a href="#">strspn</a> (const char *s, const char *accept);	визначає максимальну довжину початкового підрядка, що складається виключно з байтів, перерахованих в accept
size_t <a href="#">strcspn</a> (const char *s, const char *reject);	визначає максимальну довжину початкового підрядка, що складається виключно з байтів, не перерахованих в reject
char * <a href="#">strpbrk</a> (const char *s, const char *accept);	знаходить перше вхождення любого символа, перерахованого в accept
char * <a href="#">strstr</a> (const char *haystack, const char *needle);	знаходить перше вхождення рядка needle в haystack

## СПИСОК ЛІТЕРАТУРИ



1. Гилберт Стивен, Маккарти Билл. Самоучитель Borland C 6 в примерах. Учебник: Пер. с англ. – К.: ООО «ТИД «ДС», 2002. – 496 с.
2. Дейтел Харви, Дейтел Пол. Как программировать на С++: Пер. с англ. – 3-е изд. М.: ЗАО «Издательство БИНОМ», 2001. – 1152 с.
3. Кёниг Эндрю, Му Барбара Э. Эффективное программирование на С++. Серия С++ *In-Depth*, т. 2. : Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – 384 с.
4. Либерти Джесс. Освой самостоятельно С++ за 21 день – Учеб. пособие: Пер. с англ. – 3-е изд. – М.: Издательский дом «Вильямс», 2001. – 816 с.
5. Прата Стивен. Язык программирования С++. Лекции и упражнения. Учебник: Пер. с англ. – СПб.: ООО «ДиаСофтЮП», 2005– 1104 с.
6. Саттер Герб. Решение сложных задач на С++. Серия С++ *In-Depth*: Пер. с англ. – М.: Издательский дом «Вильямс», 2002. – Т. 4. – 400 с.
7. Страуструп Б. Язык программирования С++, спец. изд.: Пер. с англ. – М.: СПб.: «Издательство БИНОМ» – «Невский Диалект», 2002. – 1099 с.



Упорядники:

**Алексеев Михайло Олександрович**  
**Коротенко Леонід Михайлович**  
**Шевцова Ольга Сергіївна**

**МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ТА ІНДИВІДУАЛЬНІ  
ЗАВДАННЯ  
ДО ЛАБОРАТОРНИХ РОБІТ З ДИСЦИПЛІНИ  
„АЛГОРИТМІЗАЦІЯ ТА ПРОГРАМУВАННЯ”  
ЧАСТИНА 2**

Редактор \_\_\_\_\_

Підписано до друку \_\_\_\_\_ Формат 30x42/4  
Папір офсет. Ризографія. Ум.. друк. арк. \_\_.  
Обл.-вид. арк.. \_\_. Тираж 50 прим. Зам. № \_\_\_\_.

ДВНЗ «Національний гірничий університет»  
490005, м. Дніпропетровськ, просп. К.Маркса, 19.