

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Кафедра систем штучного інтелекту



## Звіт

На тему: «Лінійні алгоритми. Розгалужені алгоритми. Умовні та логічні оператори. Змінні. Константи. Типи даних. Розмір типів даних. Ввід вивід. Базові операції та вбудовані функції. Коментарі.»

*з дисципліни:* «Основи програмування»

до:

ВНС Лабораторної Роботи № 1  
Алготестер Лабораторної Роботи № 1  
Практичних Робіт до блоку № 2

**Виконав(ла):**  
Студент(ка) групи ШІ-12  
Соснило Богдан Тарасович

## **Тема роботи:**

Операції створення змінних та констант, введення-виведення даних; використання тернарних операторів, лінійні та розгалужені алгоритми; умовні та логічні оператори, типи даних, вивчення та застосування функцій.

## **Мета роботи:**

Вивчити та застосувати основні операції створення змінних і констант, введення-виведення даних, тернарні оператори, лінійні та розгалужені алгоритми, умовні та логічні оператори, різні типи даних і функції. Навчитись використовувати ці знання для написання коректного програмного коду."

## **Теоретичні відомості:**

1. Визначення та оголошення змінних та констант:

<https://acode.com.ua/urok-31-initsializatsiya-prysvoyuvannya-i-ogoloshennya-zminnyh/>

2. Типи даних:

<http://cpp.dp.ua/typy-danyh/>

3. Ввід та вивід:

<http://cpp.dp.ua/potokove-vvedennya-vyvedennya/>

4. Коментарі:

<https://acode.com.ua/urok-12-komentari/>

5. Оператори розгалуження:

<https://acode.com.ua/urok-67-operator-umovnogo-rozgaluzhennya-if-else/>

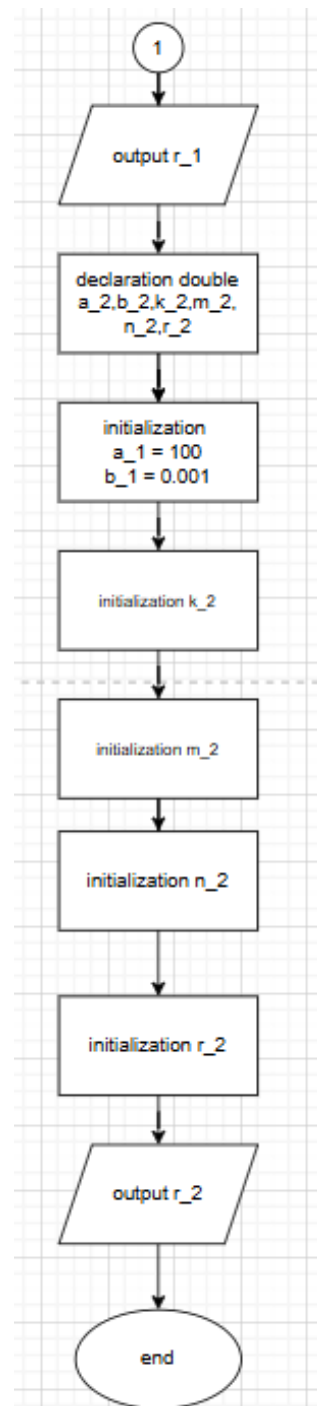
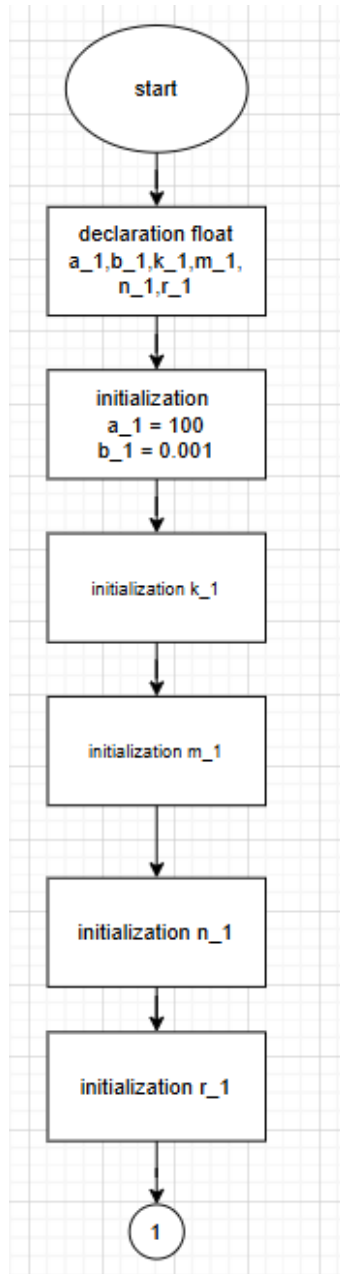
<https://acode.com.ua/urok-68-operator-switch/>

## **Виконання роботи**

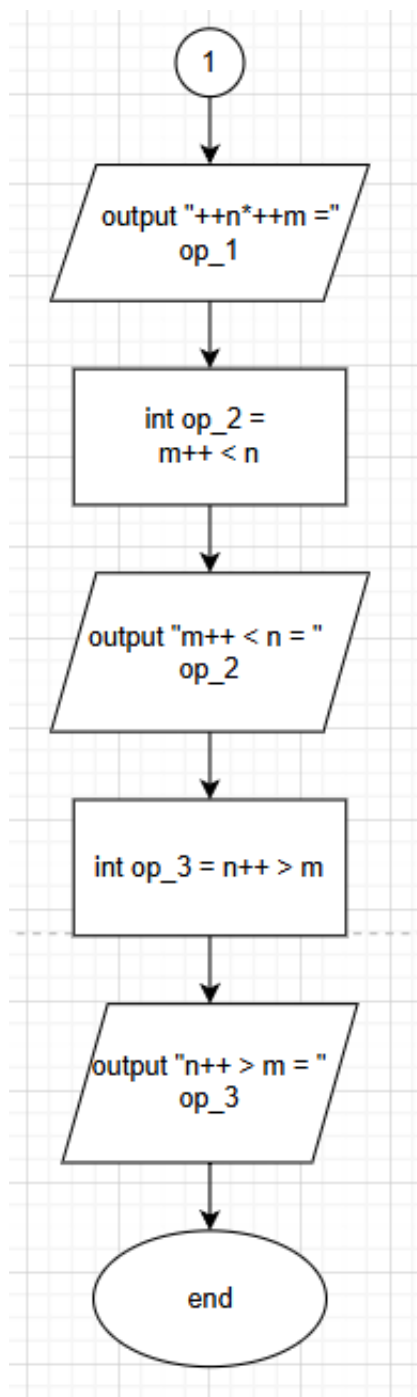
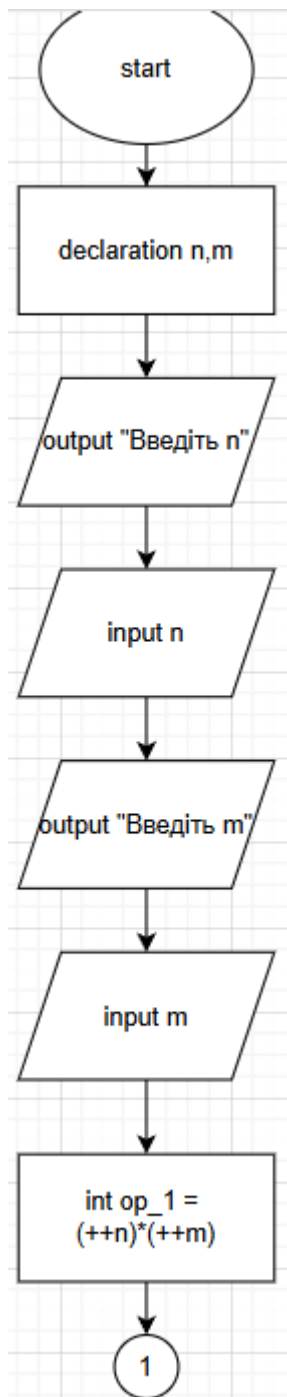
- **Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)**

Робити блоксхеми зайняло в мене приблизно 2.5 години

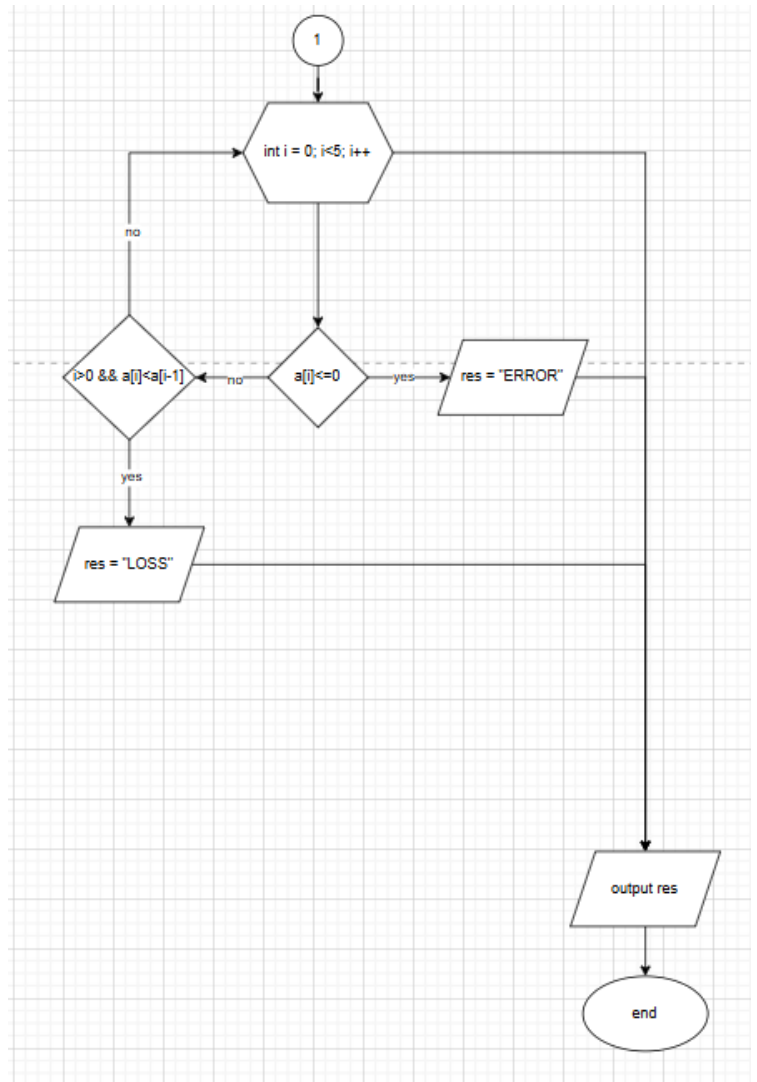
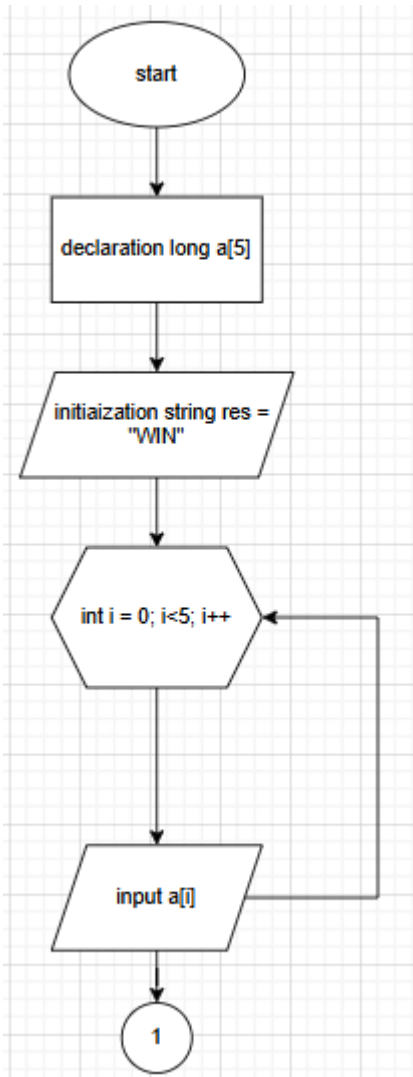
## Блоксхема до VNS Lab 1 task 1 (1)



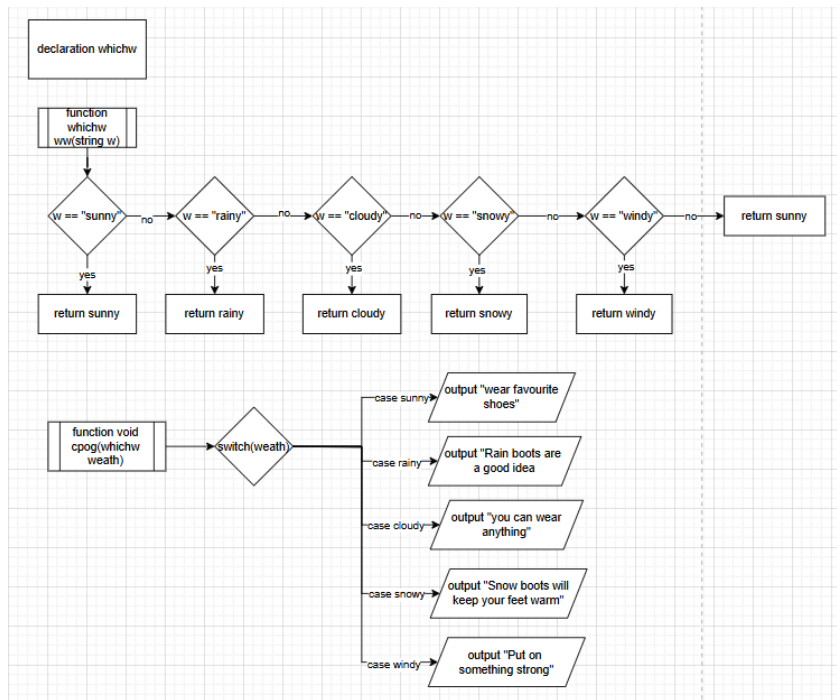
## Блоксхеми до VNS Lab 1 task 2 (2)



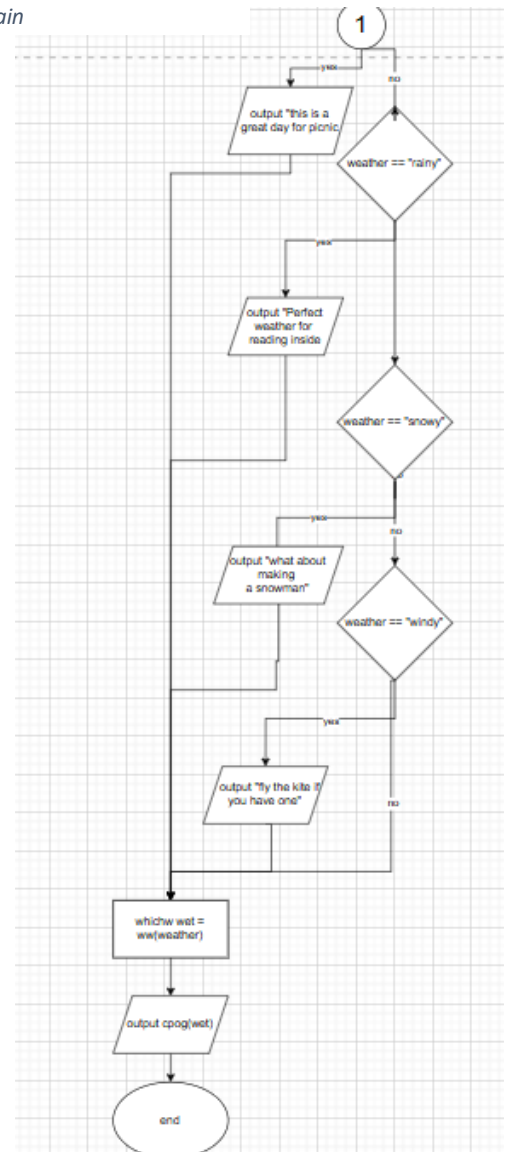
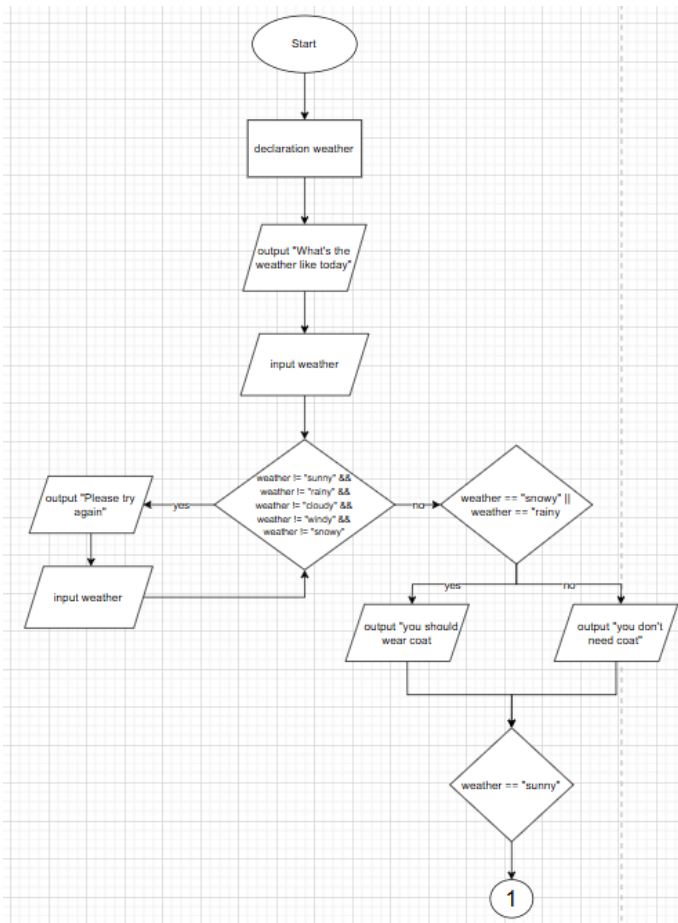
### Блоксхема Algotester Labs Lab1v3 (3)



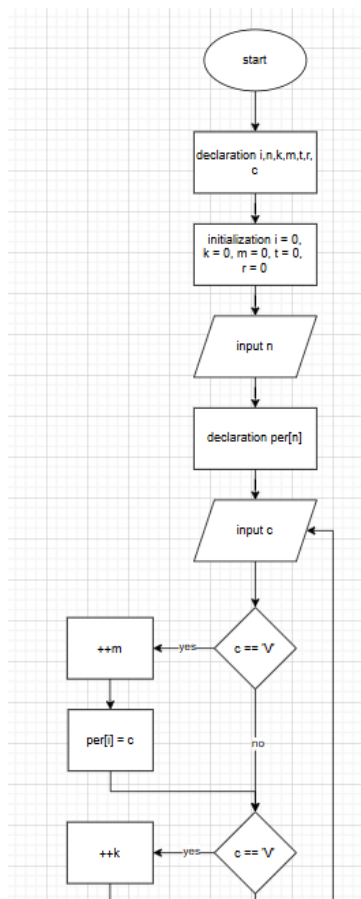
## Блоксхеми до Class Practice Work (4)



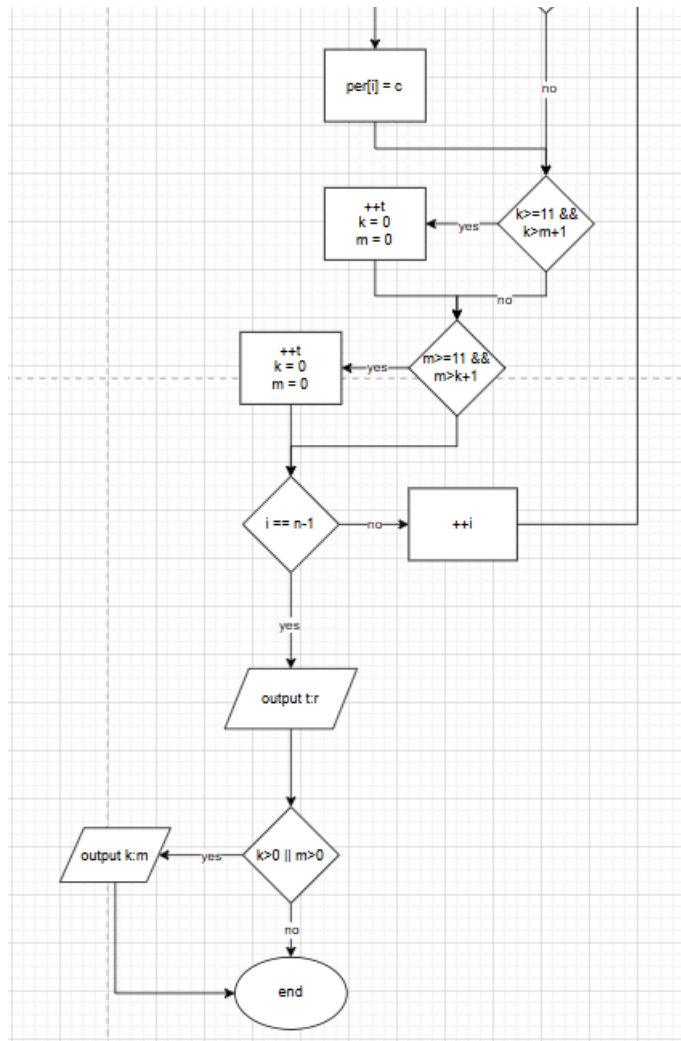
Функції, які потім викликаються в main



## Блоксхема до Self Practice Work (5)



1 частина блоксхеми



2 частина блоксхеми

### Task 3 - Lab# programming: VNS Lab 1 Task 1

Зайняло це завдання приблизно 20 хвилин

```

v #include <iostream>
#include <math.h>
using namespace std;

v int main() {
    float a_1,b_1, k_1, m_1, n_1,r_1;
    a_1 = 100;
    b_1 = 0.001;
    k_1 = pow((a_1-b_1),4);
    m_1 = pow(a_1,4)-4*pow(a_1,3)*b_1;
    n_1 = 6*pow(a_1,2)*pow(b_1,2)-4*a_1*pow(b_1,3)+pow(b_1,4);
    r_1 = (k_1-m_1)/n_1;
    cout << r_1 << endl;

    double a_2,b_2, k_2, m_2, n_2,r_2;
    a_2 = 100;
    b_2 = 0.001;
    k_2 = pow((a_2-b_2),4);
    m_2 = pow(a_2,4)-4*pow(a_2,3)*b_2;
    n_2 = 6*pow(a_2,2)*pow(b_2,2)-4*a_2*pow(b_2,3)+pow(b_2,4);
    r_2 = (k_2-m_2)/n_2;
    cout << r_2;

    return 0;
}

```

Results

0

1

#### Task 4 - Lab# programming: VNS Lab 1 Task 2

Написання цього коду зайняло в мене приблизно 20 хвилин.



```

#include <iostream>

using namespace std;

int main()
{
    int n, m;

    cout << "Введіть n" << endl;
    cin >> n;
    cout << "Введіть m" << endl;
    cin >> m;

    int op_1 = (++n)*(++m);
    cout << "++n*++m = " << op_1 << endl;

    int op_2 = m++ < n;
    cout << "m++ < n = " << op_2 << endl;

    int op_3 = n++ > m;
    cout << "n++ > m = " << op_3 << endl;

    return 0;
}

```

```

Введіть n
5
Введіть m
6
++n*++m = 42
m++ < n = 0
n++ > m = 0

```

## Task 5 - Lab# programming: Algotester Lab 1 Task 1

Написання цього коду зайняло в мене приблизно 30 хвилин

```

#include <iostream>
using namespace std;

int main() {
    long a[5];
    string res = "WIN";

    for (int i = 0; i < 5; i++) {
        cin >> a[i];
    }
    for (int i = 0; i < 5; i++) {
        if (a[i] <= 0) {
            res = "ERROR";
            break;
        }

        if (i > 0 && a[i] > a[i - 1]) {
            res = "LOSS";
            break;
        }
    }

    cout << res;
    return 0;
}

```

```

5 4 3 2 1
WIN

```

Lab 1v3	Lab 1v3	1 sec., 256 MiB	✓	1	34
---------	---------	-----------------	---	---	----

## Task 6 - Practice# programming: Class Practice Task

Написання цього коду зайняло в мене приблизно 1 годину

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  enum whichw {
6      sunny,
7      rainy,
8      cloudy,
9      snowy,
10     windy,
11 };
12
13 whichw ww(string w){
14     if(w == "sunny"){
15         return sunny;
16     }
17     if(w == "rainy"){
18         return rainy;
19     }
20     if(w == "cloudy"){
21         return cloudy;
22     }
23     if(w == "snowy"){
24         return snowy;
25     }
26     if(w == "windy"){
27         return windy;
28     }
29     return sunny;
30 }
31
32 void cpog(whichw weath){
33
34     switch(weath)
35     {
36         case sunny:
37             cout << "wear favourite shoes";
38             break;
39         case rainy:
40             cout << "Rain boots are a good ideal!";
41             break;
42         case cloudy:
43             cout << "you can wear anything";
44             break;
45         case snowy:

```

```

46             cout << "Snow boots will keep your feet warm!";
47             break;
48         case windy:
49             cout << "Put on something strong!";
50             break;
51     }
52 }
53
54
55
56 int main() {
57
58     string weather;
59
60
61     cout << "What's the weather like today?" << endl;
62     getline(cin,weather);
63     while((weather != "sunny") && (weather != "rainy") && (weather != "cloudy") && (weather != "windy") && (weather == "snowy")){
64         cout << "please try again and make it right: ";
65         getline(cin,weather);
66     }
67
68
69
70
71
72
73     if(weather == "snowy" || weather == "rainy"){
74         cout << "you should wear coat" << endl;
75     }
76     else {
77         cout << "you don't need coat today" << endl;
78     }
79     if(weather == "sunny"){
80         cout << "this is a great day for a picnic" << endl;
81     }
82     else if(weather == "rainy"){
83         cout << "Perfect weather for reading inside" << endl;
84     }
85     else if(weather == "snowy"){
86         cout << "what about making a snowman?" << endl;
87     }
88     else if(weather == "windy" ){

```

```

89         cout << "fly the kite if you have one" << endl;
90     }
91
92     whichw wet = ww(weather);
93     cpog(wet);
94
95     return 0;
96 }

```

```

what's the weather like today?
sunny
you don't need coat today
this is a great day for a picnic
wear favourite shoes

```

## Task 7 - Practice# programming: Self Practice Task

(зайняло в мене приблизно 30 хвилин)

### Коля, Вася і Теніс

Обмеження: 2 сек., 256 МБ

Коли Коля та Вася прийшли робити ремонт на «Екстралогіку» — першим, що вони побачили в офісі, був стіл для настільного тенісу. Поки всі інші працювали, Коля та Вася вирішили пограти. Через декілька годин прийшов директор і накричав на заробітчан через те, що вони нічим не займаються. Тож Вася і Коля мусили йти працювати.

По дорозі вони сперечалися, хто ж виграв і з яким рахунком. Оскільки вони записували результати кожної подачі, то це можна поррахувати. Але оскільки гра тривала дуже довго — поррахувати це вручну дуже тяжко.

Всього відбулося  $n$  подач. Про кожну з них ми знаємо, хто переміг: За виграну подачу граєць отримує одне очко. Партія вважається виграною, коли один з гравців набере не менше одинадцяти очок з перевагою щонайменше у два очки. Наприклад, за рахунків 11:9, 4:11, 15:13 партія закінчується, а за рахунків 11:10 та 99:98 — ні. Як тільки Коля і Вася закінчили одну партію — вони починають іншу.

Знаючи, хто переміг кожної подачі — виведіть загальний рахунок по партіях в грі Коля-Вася. А якщо вони не дограли останню партію, то і її рахунок теж.

#### Вхідні дані

У першому рядку задано ціле число  $n$  — загальна кількість подач.

У другому рядку задано  $n$  символів  $c_i$ .  $c_i = K$ , якщо  $i$ -ту подачу виграв Коля, та  $c_i = V$ , якщо  $i$ -ту подачу виграв Вася.

#### Вихідні дані

У першому рядку виведіть загальний рахунок гри по партіях у форматі  $k : v$ , де  $k$  — кількість партій, у яких переміг Коля, а  $v$  — кількість партій, у яких переміг Вася.

Якщо вони не дограли останню партію, то в другому рядку в такому ж форматі виведіть рахунок останньої партії.

```

1  #include <iostream>
2  using namespace std;
3
4
5  int main(){
6
7      int n,i,k,m,t,r;
8      char c;
9      i=0;
10     k = 0;
11     m = 0;
12     t = 0;
13     r = 0;
14     cin >> n;
15     char per[n];
16     while(true){
17
18         cin >> c;
19         if(c == 'V'){
20             ++m;
21             per[i] = c;
22         }
23         if(c == 'K'){
24             ++k;
25             per[i] = c;
26         }
27         if(k>= 11 && k>m+1){
28             ++t;
29             k = 0;
30             m = 0;
31         }
32         if(m>= 11 && m>k+1){
33             ++r;
34             k = 0;
35             m = 0;
36         }
37         if (i == n-1){

```

```

37         if (i == n-1){
38             break;
39         }
40         ++i;
41     }
42
43
44     cout << t << ":" << r << endl;
45     if(k>0 || m>0){
46         cout << k << ":" << m;
47     }
48
49     return 0;
50 }
51

```

15

WKKKWKVKVKWK

0:0

7:8

2 дні тому

C++ 23

Зараховано

0.008

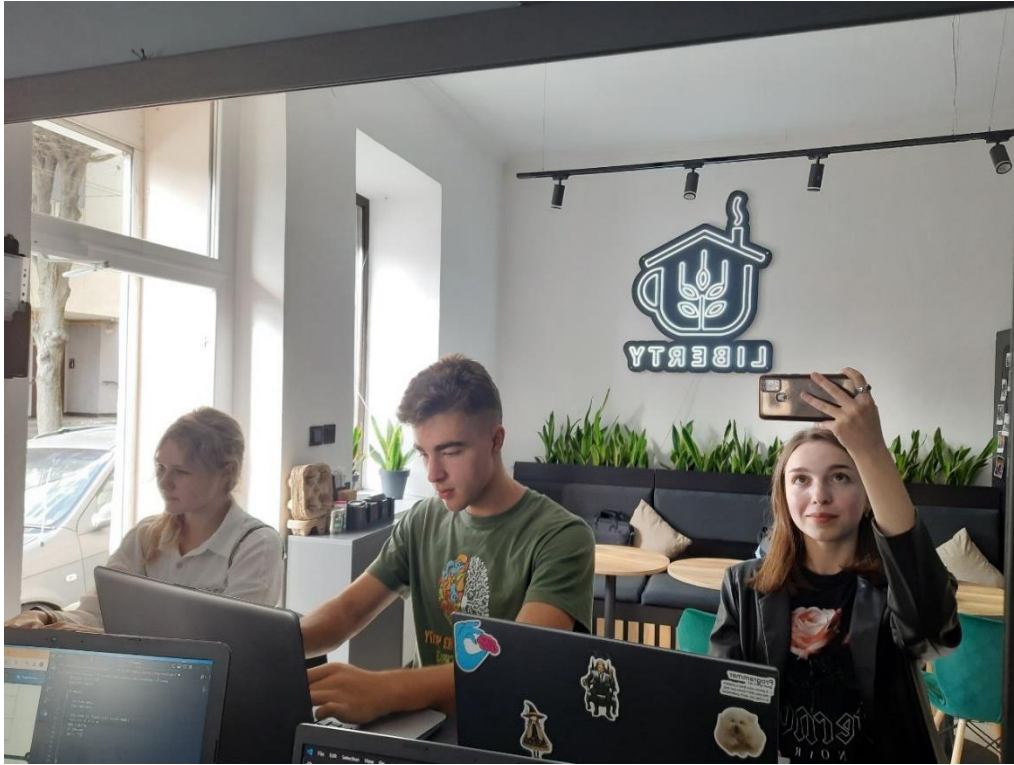
1.250

[Перегляд](#)

### Зустрічі з командою

З командою зустрічались доволі часто, обговорювали завдання по 2 і наступних епіках.





Висновок: Під час виконання лабораторної роботи ми ознайомилися з основними елементами програмування, що є фундаментом для створення будь-яких програм. Ми розглянули лінійні та розгалужені алгоритми, які дозволяють керувати виконанням програми залежно від певних умов. Навчилися працювати зі змінними та константами, використовувати різні типи даних і визначати їхній розмір, що важливо для ефективного використання пам'яті. Також ми дослідили можливості умовних та логічних операторів, які надають можливість програмі приймати рішення