

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



## Звіт

### **про виконання лабораторних та практичних робіт блоку № 2**

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 групи ІІІ-11  
Вербицький Юрій

Львів 2024

**Тема роботи:** Лінійні алгоритми. Розгалужені алгоритми. Умовні та логічні оператори. Змінні. Константи. Типи даних. Розмір типів даних. Ввід вивід. Базові операції та вбудовані функції. Коментарі.

**Мета роботи:** Покращити навички програмування через застосування лінійних і розгалужених алгоритмів. Навчитись використовувати умовні та логічні оператори, змінні, константи та коментарі. Опанувати різні типи даних і розуміти їх розміри. Зрозуміти принципи роботи основних операцій та вбудованих функцій.

### **Теоретичні відомості:**

#### **1) Перелік тем:**

- Тема №1 - Системи числення.
- Тема №2 - Компіляція.
- Тема №3 - Змінні, Константи, Типи даних та їх Розміри.
- Тема №4 - Бібліотеки в C++.
- Тема №5 - Ввід та Вивід даних.
- Тема №6 - Базові Операції та Вбудовані Функції.
- Тема №7 - Коментарі у Коді.
- Тема №8 - Лінійні алгоритми.
- Тема №9 - Розгалужені алгоритми та Умовні Оператори.
- Тема №10 - Логічні Оператори.

#### **2) Індивідуальний план опрацювання теорії:**

*Тема №1 Системи числення.*

Джерела:

<https://dotnettutorials.net/lesson/introduction-to-number-system/>

▪ *Тема №2 Компіляція.*

Джерела: <https://www.scaler.com/topics/how-to-compile-cpp/>

*Тема №3 Змінні, Константи, Типи даних та їх Розміри.*

Джерела: методичні відомості до лабораторної роботи №1

[https://www.w3schools.com/cpp/cpp\\_data\\_types.asp](https://www.w3schools.com/cpp/cpp_data_types.asp)

[https://www.w3schools.com/cpp/cpp\\_variables.asp](https://www.w3schools.com/cpp/cpp_variables.asp)

*Тема №4 Бібліотеки в C++.*

Джерела: [https://www.youtube.com/watch?app=desktop&v=a5kUr-u2UNo&ab\\_channel=MichaelForest](https://www.youtube.com/watch?app=desktop&v=a5kUr-u2UNo&ab_channel=MichaelForest)

<https://cplusplus.com/reference/>

*Тема №5 Ввід та Вивід даних.*

Джерела: методичні відомості до лабораторної роботи №1

вивід у c++: [https://www.w3schools.com/cpp/cpp\\_output.asp](https://www.w3schools.com/cpp/cpp_output.asp)

Ввід у c++: [https://www.w3schools.com/cpp/cpp\\_user\\_input.asp](https://www.w3schools.com/cpp/cpp_user_input.asp)

Ввід/вивід у c: <https://www.programiz.com/c-programming/c-input-output>

*Тема №6 Базові Операції та Вбудовані Функції.*

Джерела: <https://acode.com.ua/urok-107-vbudovani-funktsiyi/>

[https://www.w3schools.com/cpp/cpp\\_math.asp](https://www.w3schools.com/cpp/cpp_math.asp)

*Тема №7 Коментарі у Коді.*

Джерела: [https://www.w3schools.com/cpp/cpp\\_comments.asp](https://www.w3schools.com/cpp/cpp_comments.asp)

*Тема №8 Лінійні алгоритми.*

Джерела:

Лінійний пошук: <https://www.geeksforgeeks.org/linear-search/>

Сортування вибіркою: <https://youtu.be/uQxG9gBROog?si=G29n3iZ3MTKzV7nb>

Сортування бульбашкою: <https://youtu.be/maB87eyn7h8?si=cC2fx4RPn98COC9a>

*Тема №9 Розгалужені алгоритми та Умовні Оператори.*

Джерела:

If/else: <https://youtu.be/zogwWqGyM2c?si=kGj7p1oiRmzQ5rod>

Оператор switch: [https://www.w3schools.com/cpp/cpp\\_switch.asp](https://www.w3schools.com/cpp/cpp_switch.asp)

*Тема №10 Логічні Оператори.*

Джерела: [https://www.w3schools.com/cpp/cpp\\_operators\\_logical.asp](https://www.w3schools.com/cpp/cpp_operators_logical.asp)

## **Виконання роботи:**

### **1) Перелік завдань:**

- Epic 2 Task 1 - Theory Education Activities
- Epic 2 Task 2 - Requirements management (understand tasks) and design activities (draw flow diagrams and estimate tasks 3-7)
- Epic 2 Task 3 - Lab# programming: VNS Lab 1 Task 1
- Epic 2 Task 4 - Lab# programming: VNS Lab 1 Task 2
- Epic 2 Task 5 - Lab# programming: Algotester Lab 1 Task 1
- Epic 2 Task 6 - Practice# programming: Class Practice Task
- Epic 2 Task 7 - Practice# programming: Self Practice Task
- Epic 2 Task 8 - Result Documentation Report and Outcomes Placement Activities (Docs and Programs on GitHub)
- Epic 2 Task 9 - Results Evaluation and Release

## 2) Умови завдань:

### Task 3:

$$\frac{(a+b)^4 - (a^4 + 4a^3b)}{6a^2b^2 + 4ab^3 + b^4},$$

при  $a=100$ ,  $b=0.001$

Обчислити значення виразу при різних дійсних типах даних (float й double).

Обчислення варто виконувати з використанням проміжних змінних. Порівняти й пояснити отримані результати.

### Task 4:

1)  $++n * ++m$

2)  $m++ < n$

3)  $n++ > m$

Обчислити значення виразів. Пояснити отримані результати.

**Task 5:** Ви створюєте простий порадник щодо погоди. Користувач вводить поточні погодні умови, а програма видає рекомендації щодо активності на основі погоди.

Можливі варіанти погоди:

- sunny;
- rainy;
- cloudy;
- snowy;
- windy;

**Task 6:** У вашого персонажа є  $N$  хітпойнтів та  $M$  мани.

Персонаж 3 рази використає закляття, кожне з яких може використати хітпойнти та ману одночасно.

Якщо якесь закляття забирає і хітпойнти і ману - ваш персонаж програє, отже для виграшу треба використовувати при одному заклинанні **АБО** хітпойнти, **АБО** ману.

Якщо в кінці персонаж буде мати додатню кількість хітпойнтів та мани ( $H, M > 0$ ) - він виграє, в іншому випадку програє.

Ваше завдання у випадку виграшу персонажа вивести **YES**, вивести **NO** у іншому випадку.

## Input

2 цілих числа  $H$  та  $M$  - хітпойнти та мана персонажа

3 рядки по 2 цілих числа,  $hi$  та  $mi$  - кількість хітпойнтів та мани, які ваш персонаж потратить за хід на  $i$  заклинання

## Output

YES - якщо ваш персонаж виграє

NO - у всіх інших випадках

### Task 7:

Персонажу по одному дають сторони 5 кубів  $a_1..a_5$ , з яких він будує піраміду.

Коли він отримує куб з ребром  $a_i$  - він його ставить на існуючий, перший ставить на підлогу (вона безмежна).

Якщо в якийсь момент об'єм куба у руці (який будуть ставити) буде більший ніж у куба на вершині піраміди - персонаж програє і гра закінчується. Розмір усіх наступних кубів після програшу **не враховується**.

Тобто якщо  $a_{i-1} < a_i - 1 < a_i$  - це програш.

Ваше завдання - сказати як закінчиться гра.

## Input

5 цілих чисел  $a_1..a_5$  - сторони кубів

## Output

Існуючі варіанти:

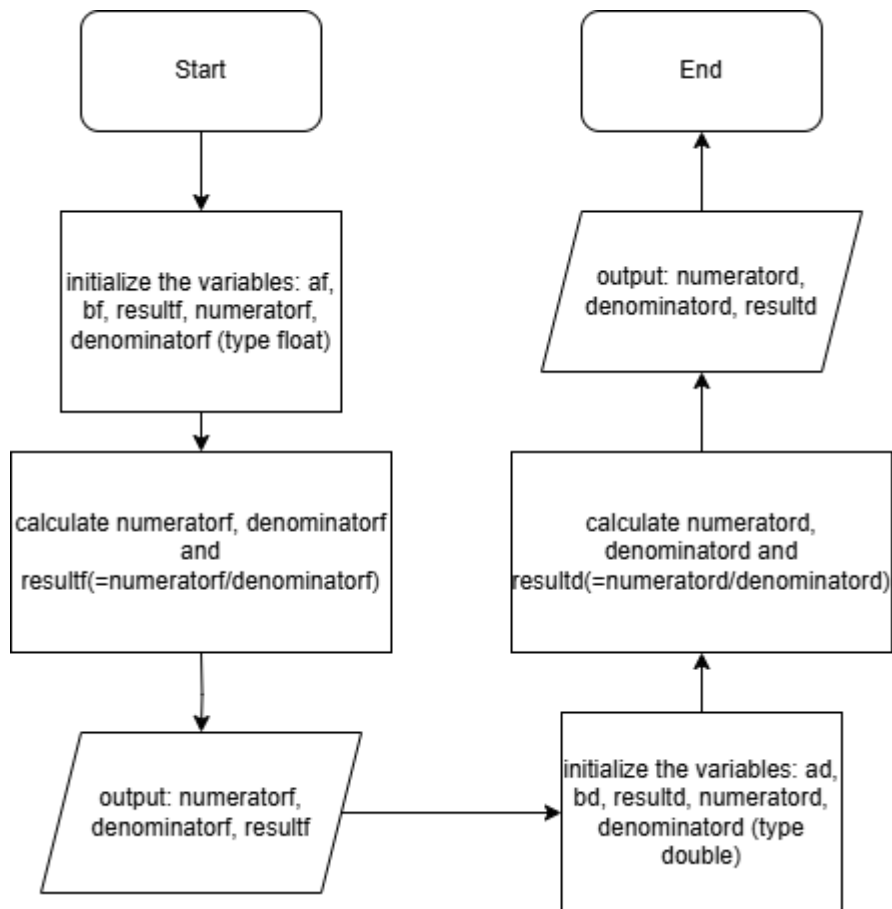
**LOSS** - якщо персонаж не зможе поставити куб.

**WIN** - якщо персонаж зможе поставити усі куби

**ERROR** - якщо сторона куба буде мати неможливу довжину, тобто  $a_i \leq 0$

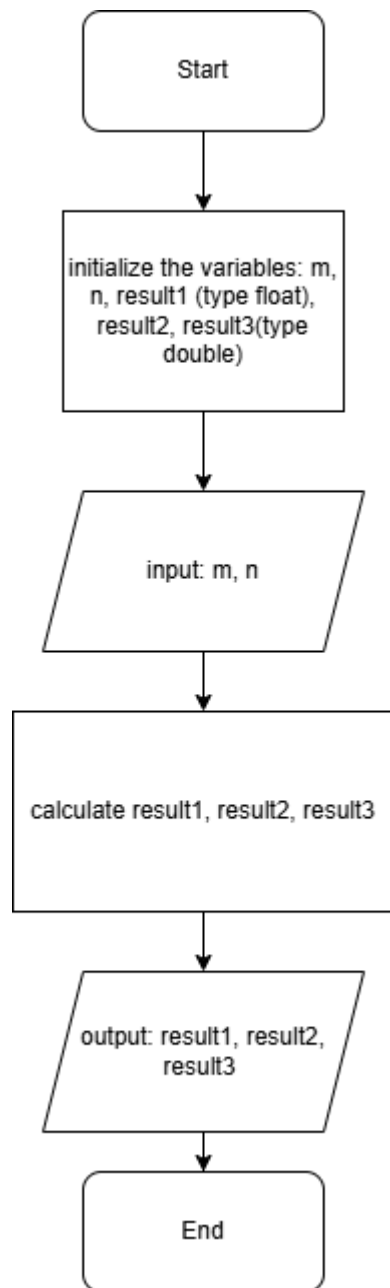
### 3) Дизайн та планова оцінка часу виконання завдань:

Task 3 - Lab# programming: VNS Lab 1 Task 1 Варіант 9



Орієнтований час виконання: 25 хв

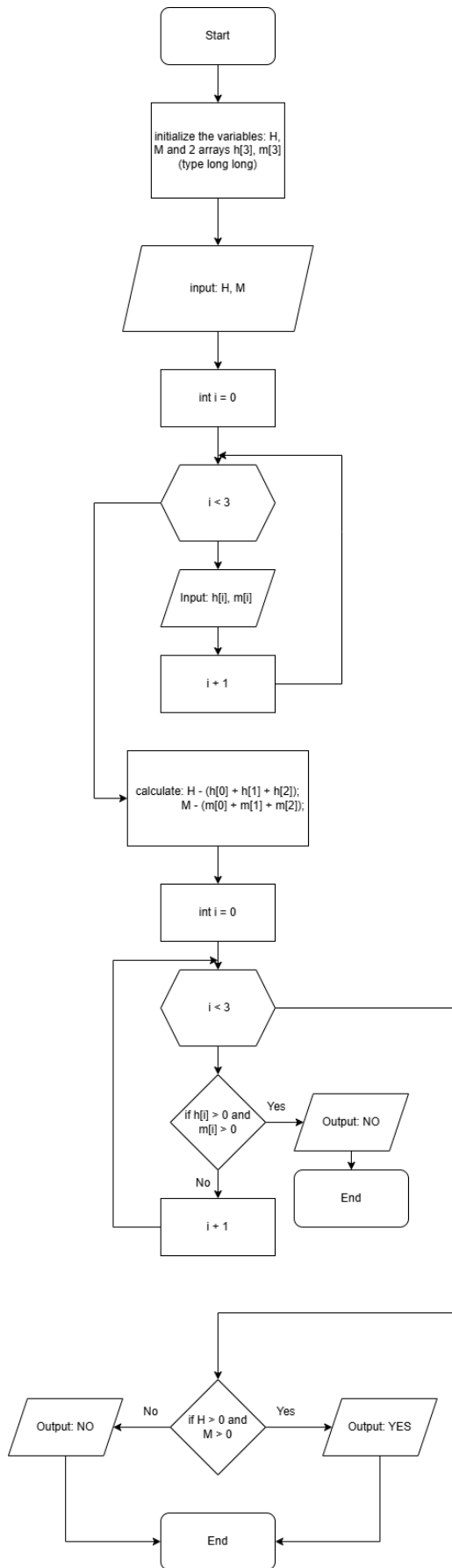
Task 4 - Lab# programming: VNS Lab 1 Task 2 Варіант 9



Орієнтовний час виконання: 10хв

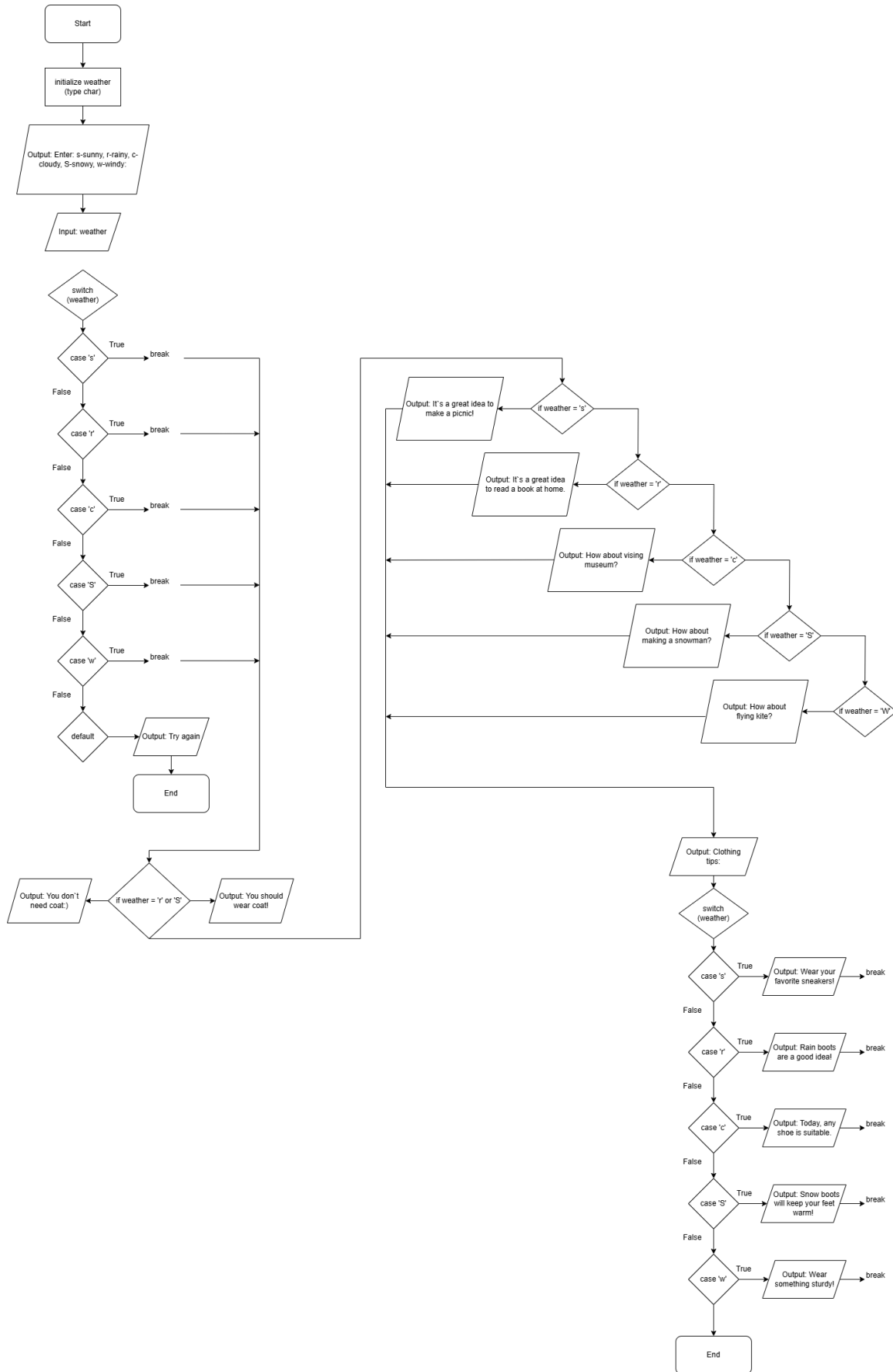


## Task 5 - Lab# programming: Algotester Lab 1 Task 1



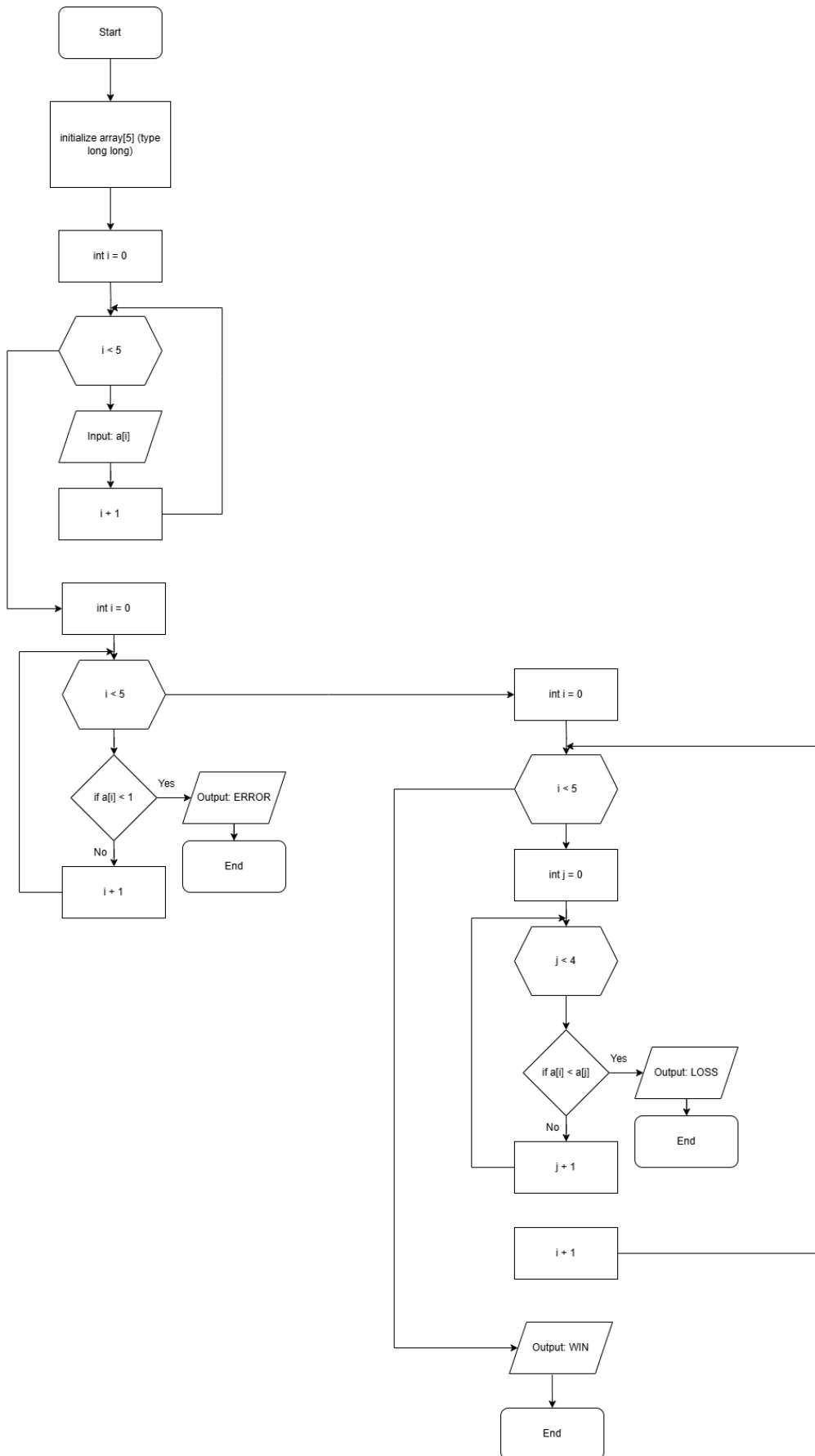
Орієнтований час виконання: 1 год

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



Орієнтований час виконання: 1,5 год

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



Орієнтований час виконання 10хв

#### 4) Код програм з посиланням на зовнішні ресурси:

Task 3 - Lab# programming: VNS Lab 1 Task 1 Варіант 9

```
vns_lab_1_task_1_variant_1_yurii_verbytskyi.cpp > main()
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6
7  int main(){
8      float af = 100;
9      float bf = 0.001;
10     float resultf;
11     float numeratorf;
12     float denominatorf;
13
14     numeratorf = pow(af+bf, 4) - (pow(af, 4) + 4*pow(af,3) * bf);
15     denominatorf = 6*pow(af, 2)*pow(bf, 2) + 4*af*pow(bf, 3) + pow(bf, 4);
16     resultf = numeratorf / denominatorf;
17     cout << "Float result: " << endl;
18     cout << numeratorf << endl;
19     cout << denominatorf << endl;
20     cout << resultf << endl;
21
22     double ad = 100;
23     double bd = 0.001;
24     double resultd;
25     double numeratord;
26     double denominatord;
27
28     numeratord = pow(ad+bd, 4) - (pow(ad, 4) + 4*pow(ad,3) * bd);
29     denominatord = 6*pow(ad, 2)*pow(bd, 2) + 4*ad*pow(bd, 3) + pow(bd, 4);
30     resultd = numeratord / denominatord;
31     cout << "Double result: " << endl;
32     cout << numeratord << endl;
33     cout << denominatord << endl;
34     cout << resultd << endl;
35
36     return 0;
37 }
```

Task 4 - Lab# programming: VNS Lab 1 Task 2 Вapianт 9

```
vns_lab_1_task_2_variant_1_yurii_verbytskyi.cpp > main()
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6
7  int main(){
8      float m;
9      float n;
10     float result1;
11     bool result2;
12     bool result3;
13     cout << "Enter m: ";
14     cin >> m;
15     cout << "Enter n: ";
16     cin >> n;
17
18     result1 = (++n) * (++m);
19     result2 = (m++) < n;
20     result3 = (n++) > m;
21
22     cout << result1 << endl;
23     cout << result2 << endl;
24     cout << result3 << endl;
25
26     return 0;
27 }
```

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

```
algotester_lab_1_variant_1_yurii_verbytskyi.cpp > main()
1  #include <iostream>
2  using namespace std;
3  int main(){
4  long long H, M;
5  long long h[3];
6  long long m[3];
7  cin >> H >> M;
8  for(int i = 0; i < 3; i++){
9      cin >> h[i] >> m[i];
10 }
11     H = H - (h[0] + h[1] + h[2]);
12     M = M - (m[0] + m[1] + m[2]);
13     for(int i=0; i<3; i++){
14         if (h[i] > 0 && m[i] > 0) {
15             cout << "NO" << endl;
16             return 0;
17         }
18     }
19
20     if( H > 0 && M > 0){
21         cout << "YES";
22     }else cout << "NO";
23     return 0;
24 }
```

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

```
practice_work_task_1_yurii_verbytskyi.cpp > main()
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6      char weather;
7
8      cout << "Enter: s-sunny, r-rainy, c-cloudy, S-snowy, w-windy:" << endl;
9      cin >> weather;
10
11
12      switch (weather) {
13      case 's':
14      case 'r':
15      case 'c':
16      case 'S':
17      case 'w':
18          break;
19      default: // same as else
20          cout << "Try again" << endl;
21          return 0;
22      }
23
24      if(weather == 'r' || weather == 'S'){
25          cout << "You should wear coat!" << endl;
26      }else cout << "You don't need coat:)" << endl;
27
28
29      if(weather=='s'){
30          cout << "It's a great idea to make a picnic!" << endl;
31      }
32      else if(weather=='r'){
33          cout << "It's a great idea to read a book at home." << endl;
34      }
35      else if(weather=='c'){
36          cout << "How about vising museum?" << endl;
37      }
38      else if(weather=='S'){
39          cout << "How about making a snowman?" << endl;
40      }
41      else if(weather=='w'){
42          cout << "How about flying kite?" << endl;
43      }
44
45
46      cout << "Clothing tips: ";
47      switch (weather) {
48      case 's':
49          cout << "Wear your favorite sneakers! \n";
50          break;
51      case 'r':
52          cout << "Rain boots are a good idea! \n";
53          break;
54      case 'c':
55          cout << "Today, any shoe is suitable. \n";
56          break;
57      case 'S':
58          cout << "Snow boots will keep your feet warm! \n";
59          break;
60      case 'w':
61          cout << "Wear something sturdy! \n";
62          break;
63      }
64
65
66
67
68      return 0;
69  }
```

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

### Algotester Lab 1 Task 3

```
self_practice_work_algotester_task_1_yurii_verbytskyi.cpp > main()
1  #include <iostream>
2
3  using namespace std;
4
5  int main(){
6
7      long long a[5];
8
9      for(int i = 0; i < 5; i++){
10         cin >> a[i];
11     }
12
13     for(int i = 0; i < 5; i++){
14         if(a[i] < 1){
15             cout << "ERROR";
16             return 0;
17         }
18     }
19
20     for(int i = 0; i < 4; i++){
21         for(int j = i + 1; j < 5; j++){
22             if(a[i] < a[j]){
23                 cout << "LOSS";
24                 return 0;
25             }
26         }
27     }
28
29
30
31     cout << "WIN";
32
33
34     return 0;
35 }
36
```



## 5) Результати виконання завдань та фактично затрачений час

Task 3 - Lab# programming: VNS Lab 1 Task 1 Варіант 9

```
Float result:  
-2.13752  
0.0600004  
-35.6251  
Double result:  
0.0600004  
0.0600004  
1
```

Результат вийшов таким, тому що на певному етапі обрахунків, кількість цифр(в типі float) виявилась недостатньою, що докорінно змінило кінцевий результат.

Фактично затрачений час: 15 хв

Task 4 - Lab# programming: VNS Lab 1 Task 2 Варіант 9

```
Enter m: 5  
Enter n: 7  
48  
1  
1  
PS C:\Users\Yura\Desktop\c++>
```

```
Enter m: 4  
Enter n: 4  
25  
0  
0  
PS C:\Users\Yura\Desktop\c++>
```

Фактичний час затрачений на виконання: 10хв

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

50 60	50 60	50 60
0 10	10 0	30 0
10 0	0 10	20 0
30 0	10 10	1 0
YES	NO	NO

Фактичний час затрачений на виконання: 1 год

Алготестер:

Contact Us

College

Sponsors

Donate

english

Юрій Вербицький

Lab 1v1. Lab 1v1 | NULP\_LABS\_Programming\_Basics\_2024

NULP\_LABS\_Programming\_Basics\_2024 > Lab 1v1

← Back

✓ Lab 1v1

Lab 1v2

✓ Lab 1v3

Lab 2v1

Lab 2v2

Lab 2v3

Lab 3v1

Lab 3v2

Lab 3v3

Lab 4v1

Lab 4v2

Lab 4v3

Lab 5v1

Lab 5v2

Lab 5v3

Lab 6v1

Lab 6v2

Lab 6v3

Lab 7v1

Lab 7v2

Lab 7v3

Scoreboard

Html

Pdf

Submit a solution

### Lab 1v1

Limits: 1 sec., 256 MiB

У вашого персонажа є  $H$  хітпойнтів та  $M$  мани.

Персонаж 3 рази використовує закляття, кожне з яких може використати хітпойнти та ману одночасно.

Якщо якийсь закляття забирає і хітпойнти і ману - ваш персонаж програє, отже для виграшу треба використовувати при одному заклинанні **АБО** хітпойнти, **АБО** ману.

Якщо в кінці персонаж буде мати додатню кількість хітпойнтів та мани ( $H, M > 0$ ) - він виграє, в іншому випадку програє.

Ваше завдання у випадку виграшу персонажа вивести **YES**, вивести **NO** у іншому випадку.

### Input

2 цілих числа  $H$  та  $M$  - хітпойнти та мана персонажа

3 рядки по 2 цілих числа,  $h_i$  та  $m_i$  - кількість хітпойнтів та мани, які ваш персонаж потратить за хід на  $i$  заклинання

### Output

**YES** - якщо ваш персонаж виграє

**NO** - у всіх інших випадках

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

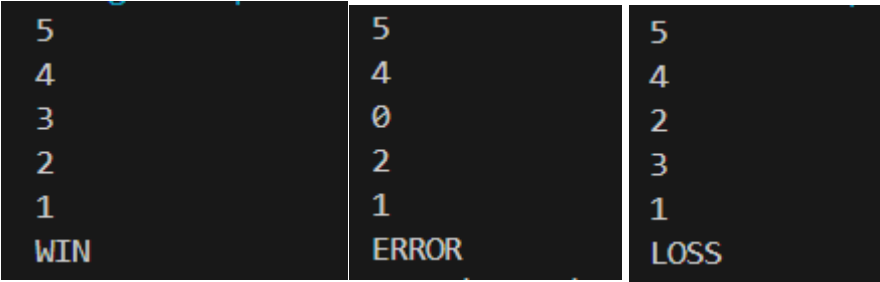
```
Enter: s-sunny, r-rainy, c-cloudy, S-snowy, w-windy:  
w  
You don't need coat:)  
How about flying kite?  
Clothing tips: Wear something sturdy!
```

```
Enter: s-sunny, r-rainy, c-cloudy, S-snowy, w-windy:  
r  
You should wear coat!  
It's a great idea to read a book at home.  
Clothing tips: Rain boots are a good idea!
```

```
Enter: s-sunny, r-rainy, c-cloudy, S-snowy, w-windy:  
f  
Try again
```

Фактичний час виконання: 1 год

Task 7 - Practice# programming: Self Practice Task



Алготестер:

Contact Us

College

Sponsors

Donate

YoutTube

Facebook

Telegram

Flag icon

english

Info icon

Profile icon

Юрій Вербицький

Lab 1v3. Lab 1v3 | NULP\_LABS\_Programming\_Basics\_2024

NULP\_LABS\_Programming\_Basics\_2024 > Lab 1v3

Back

Lab 1v1

Lab 1v2

Lab 1v3

Lab 2v1

Lab 2v2

Lab 2v3

Lab 3v1

Lab 3v2

Lab 3v3

Lab 4v1

Lab 4v2

Lab 4v3

Lab 5v1

Lab 5v2

Lab 5v3

Lab 6v1

Lab 6v2

Lab 6v3

Lab 78v1

Lab 78v2

Lab 78v3

Scoreboard

Html

Pdf

Submit a solution

Lab 1v3

Limits: 1 sec., 256 MiB

Персонажу по одному дають сторони 5 кубів  $a_{1..5}$ , з яких він буде піраміду.

Коли він отримує куб з ребром  $a_i$  - він його ставить на існуючий, перший ставить на підлогу (вона безмежна).

Якщо в якийсь момент об'єм куба у руці (який будуть ставити) буде більший ніж у куба на вершині піраміди - персонаж програє і гра закінчується.

Розмір усіх наступних кубів після програшу **не враховується**.

Тобто якщо  $a_{i-1} < a_i$  - це програш.

Ваше завдання - сказати як закінчиться гра.

Input

5 цілих чисел  $a_1 \dots a_5$  - сторони кубів

Output

Існуючі варіанти:

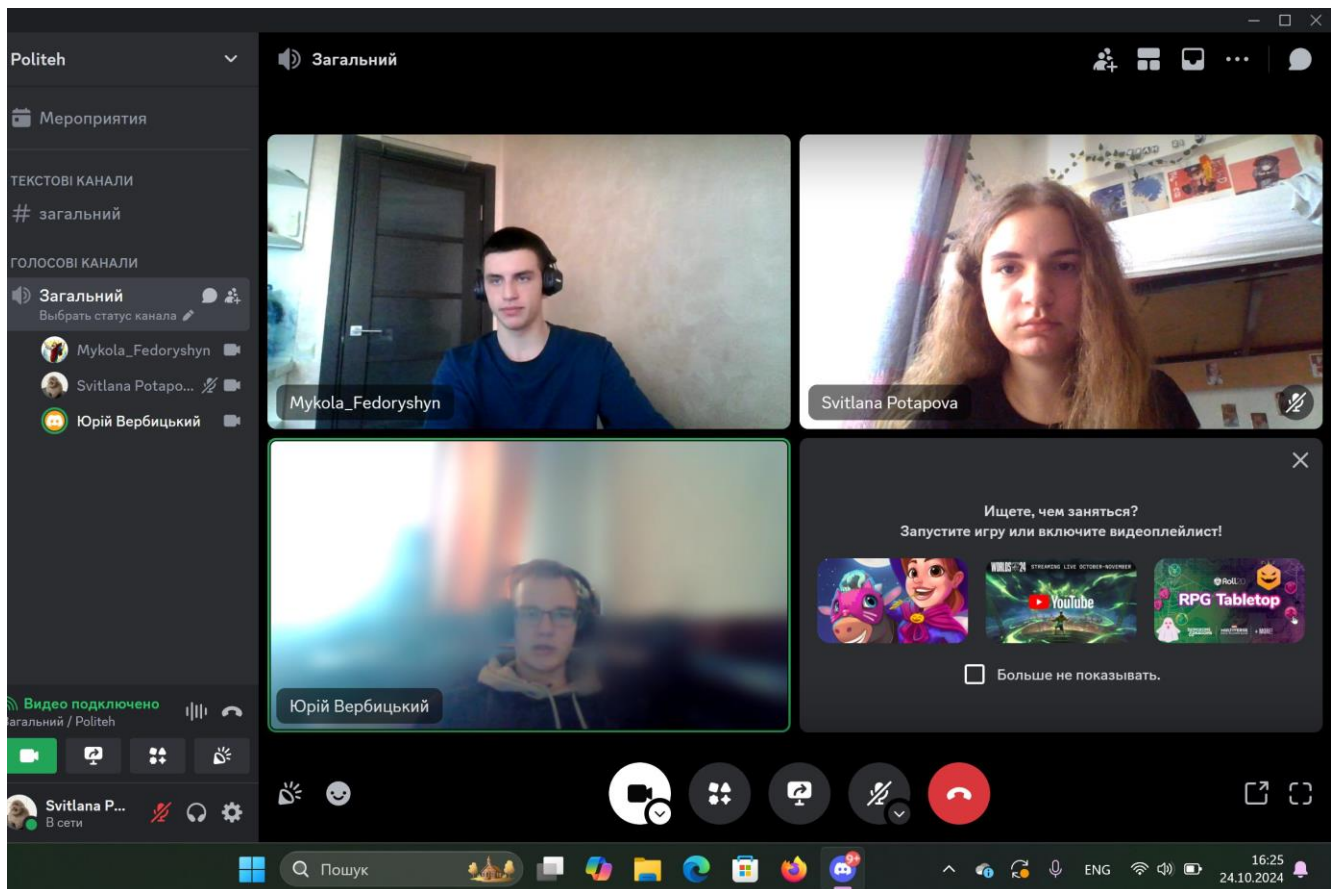
**LOSS** - якщо персонаж не зможе поставити куб.

**WIN** - якщо персонаж зможе поставити усі куби

Фактичний час виконання: 3 год

## б) Робота з комадою

Відео-зустріч:



Посилання на pull request:

**Висновок:** У межах цього епіку я ознайомився з лінійними і розгалуженими алгоритмами, навчився застосовувати умовні та логічні оператори, працювати зі змінними, константами і коментарями, а також розібрався з принципами роботи базових операцій і вбудованих функцій.