

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



Звіт

**про виконання розрахунково-графічних робіт блоку № 7
з дисципліни: «Основи програмування»**

до:

**ВНС Розрахунково-графічних робіт № 1-4
Практичних Робіт до блоку № 7**

Виконав:

Студент групи ШІ-12
Стефанович Євгеній

Львів 2024

Мета роботи:

Одержати практичні навички в розробці і дослідженні алгоритмів розв'язання задач.

Теоретичні відомості:**Теми, необхідні для виконання роботи:**

Всі теми, пройдені під час семестру.

Джерела використані для ознайомлення з вищезазначеними темами:

Ознайомився під час навчання.

Виконання роботи:

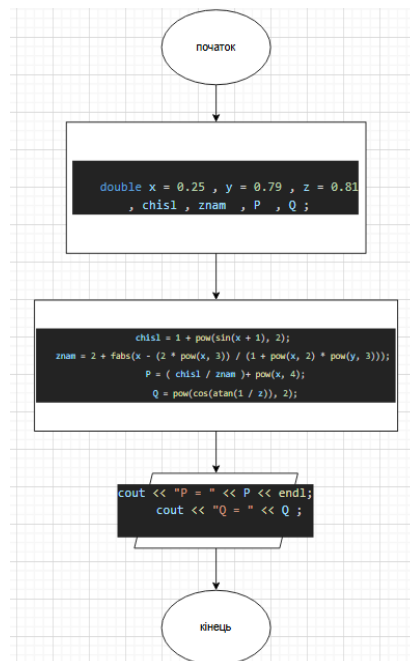
Завдання №1 – VNS Practice Work Task 1 v16

Задача:

Розробити лінійний алгоритм для розв'язання задачі.

Варіант 16.
$$P = \frac{1 + \sin^2(x + 1)}{2 + \left| x - 2x^3 / (1 + x^2 y^3) \right|} + x^4; \quad Q = \cos^2\left(\arctg \frac{1}{z}\right),$$

де $x=0,25$; $y=0,79$; $z=0,81$.



```
> Users > Eugene > Desktop > epic_7 > G:\vns_practice_work_1_task_eugenie_stefanovich.cpp > main()
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  int main() {
7
8      double x = 0.25 , y = 0.79 , z = 0.81 , chisl , znam , P , Q ;
9
10     chisl = 1 + pow(sin(x + 1), 2);
11     znam = 2 + fabs(x - (2 * pow(x, 3)) / (1 + pow(x, 2) * pow(y, 3)));
12     P = ( chisl / znam )+ pow(x, 4);
13     Q = pow(cos(atan(1 / z)), 2);
14
15     cout << "P = " << P << endl;
16     cout << "Q = " << Q ;
17
18     return 0;
19 }
20
```

```
PS C:\Users\Eug>
PS C:\Users\Eug>
P = 0.860141
Q = 0.396172
PS C:\Users\Eug>
```

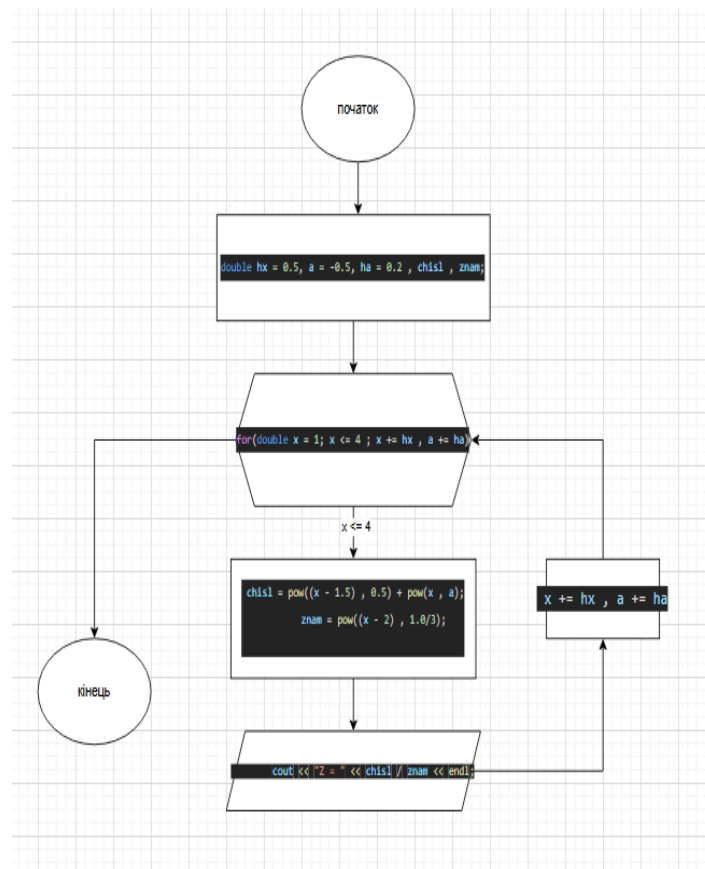
затратність ~15хв

Завдання №2 – VNS Practice Work Task 2 v16

Задача:

Варіант 15. $z = \frac{\sqrt{x-1,5} + x^a}{(x-2)^{1/3}}; \quad x \in [1,4]; \quad h_x = 0,5; \quad a > -0,5;$

$h_a = 0,2$, де x і a змінюються одночасно.



затратність ~10хв

```
C: > Users > Eugene > Desktop > epic_7 > vns_practice_work_2_task_eugenie_stefanovich.cpp > main()
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  int main()
7  {
8      double hx = 0.5, a = -0.5, ha = 0.2, chisl, znam;
9
10     for(double x = 1; x <= 4; x += hx, a += ha)
11     {
12         chisl = pow((x - 1.5), 0.5) + pow(x, a);
13         znam = pow((x - 2), 1.0/3);
14         cout << "Z = " << chisl / znam << endl;
15     }
16     return 0;
17 }
```

затратність ~5хв

```

PS C:\Users\Eugene\Downloads\
Z = nan
Z = nan
Z = inf
Z = 2.64074
Z = 2.61513
Z = 2.86975
Z = 3.34954
PS C:\Users\Eugene\Downloads\

```

Завдання №3 – VNS Practice Work Task 3 v10

Варіант 10. Обчислення сили струму в електричній ланцюгу. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Обчислення сили струму в електричному ланцюзі.

Введіть початкові дані:

Напруга (вольт) > **36**

Опір (Ом) > **1500**

Сила струму: 0.024 Ампер.

Обчислення різниці творів $(a*b)$ і $(c*d)$ чотирьох довільних дійсних чисел a , b , c і d .

Введіть з клавіатури значення: $a=2,1265123$; $b=3,1116743$; $c=1,5224222$; $d=4,346372$.

затратність ~10хв

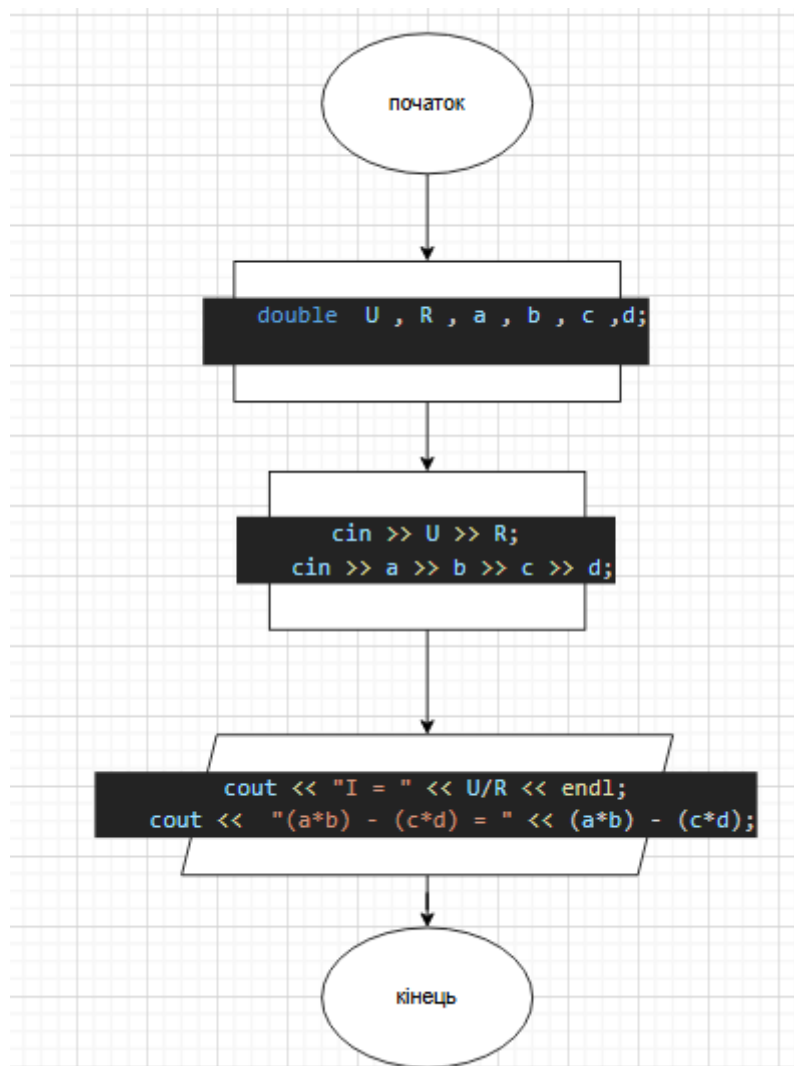
```

C: > Users > Eugene > Desktop > epic_7 > vns_practice_work_3_task_eugenie_stefanovich.cpp > main()
1  #include <iostream>
2  #include <cmath>
3
4  using namespace std;
5
6  int main()
7  {
8      double U, R, a, b, c, d;
9
10     cin >> U >> R;
11     cin >> a >> b >> c >> d;
12
13     cout << "I = " << U/R << endl;
14     cout << "(a*b) - (c*d) = " << (a*b) - (c*d);
15     return 0;
16 }

```

```
PS C:\Users\Eugene\Downloads\output> & .\36 1500
2.1265123 3.1116743 1.5224222 4.346372
I = 0.024
(a*b) - (c*d) = 4.50285e-07
PS C:\Users\Eugene\Downloads\output> █
```

затратність ~10хв



Завдання №4 – VNS Practice Work Task 2 variant 11

Задача:

Варіант 11. Написати програму, яка виводить таблицю значень функції $y = -2,4x + 5x - 3$ в діапазоні від -2 до 2, з кроком 0,5.

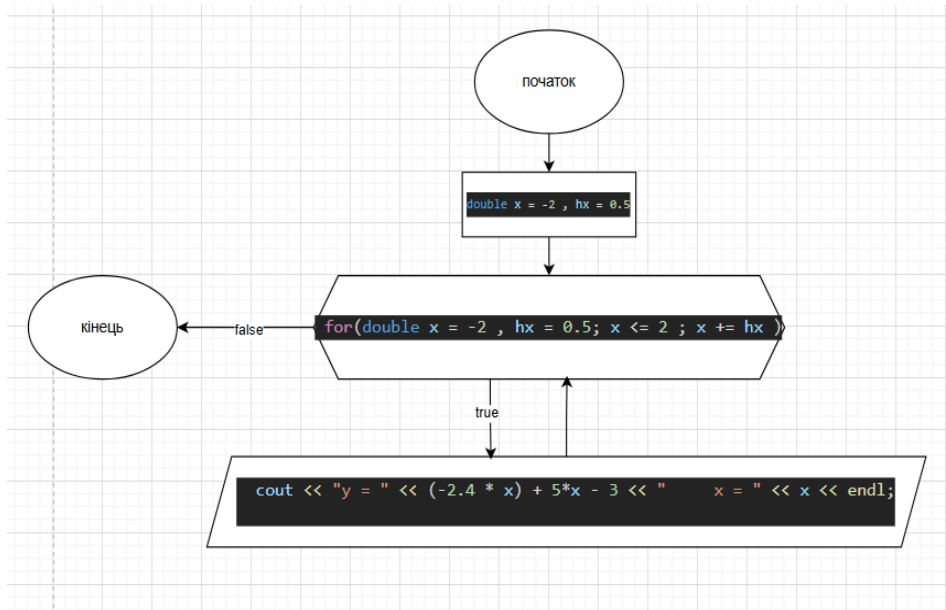
```
C:\Users\Eugene\Desktop> cd epic_7 & g++ vns_practice_work_4_task_eugenie_stefanovich.cpp & main()
1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      for(double x = -2 , hx = 0.5; x <= 2 ; x += hx )
8          cout << "y = " << (-2.4 * x) + 5*x - 3 << "      x = " << x << endl;
9
10     return 0;
11 }
```

```

PS C:\Users\Eugene\Downloa
y = -8.2      x = -2
y = -6.9      x = -1.5
y = -5.6      x = -1
y = -4.3      x = -0.5
y = -3        x = 0
y = -1.7      x = 0.5
y = -0.4      x = 1
y = 0.9       x = 1.5
y = 2.2       x = 2
PS C:\Users\Eugene\Downloa

```

затратність ~10хв



затратність ~10хв

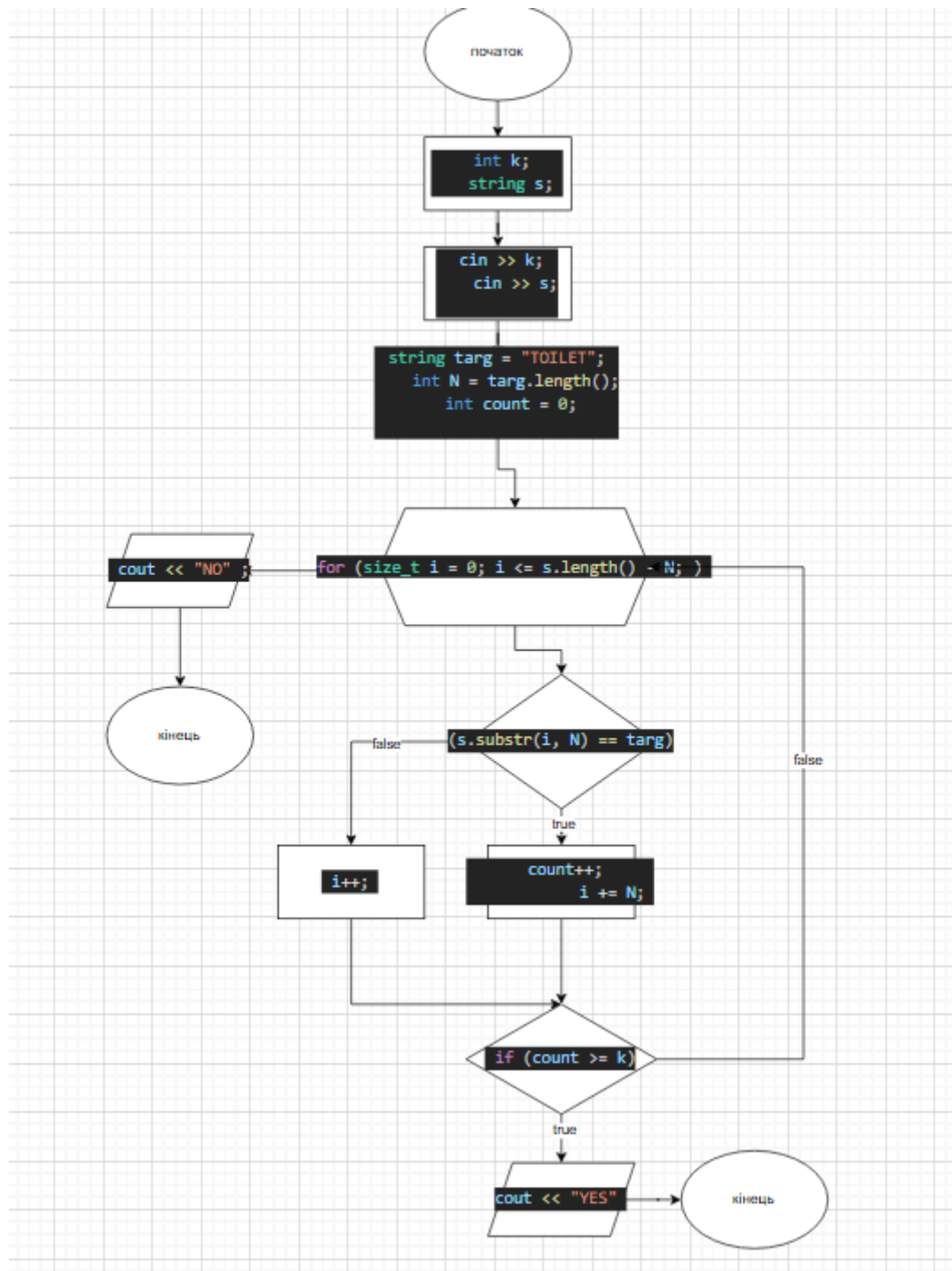
Завдання №5 – Algotester Допоможе чи заб'є?

Задача:

До Тойлет-мена, відомого вам білоруського супергероя, дуже часто звертаються по допомогу різні люди. Проте, очевидно, усім мужній гігант допомогти не зможе — не вистачить часу. Та й не дуже хоче.

Саме тому, коли до нього приходить певне SMS-повідомлення з проханням про допомогу, він погодиться допомогти тоді й лише тоді, коли в цьому повідомленні знайдеться хоча б к входжень рядка **TOILET**, які не перетинаються.

За заданим повідомленням s, яке складається з великих латинських символів, виведіть **YES**, якщо Тойлет-мен погодиться допомогти людині, яка написала це повідомлення. У протилежному разі виведіть **NO**.



```

> Users > Eugene > Desktop > epic_7 > G: algotester_practice_work_5_task_eugenie_stefanovich.cpp > main()
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main() {
7      int k;
8      string s;
9
10
11      cin >> k;
12      cin >> s;
13
14      string targ = "TOILET";
15      int N = targ.length();
16      int count = 0;
17
18
19      for (size_t i = 0; i <= s.length() - N; ) {
20          if (s.substr(i, N) == targ) {
21              count++;
22              i += N;
23          } else {
24              i++;
25          }
26
27
28          if (count >= k) {
29              cout << "YES" << endl;
30              return 0;
31          }
32      }
33
34      cout << "NO" << endl;
35      return 0;
36  }
37

```

затратність ~10хв

```

PS C:\Users\Eugene\Downloads\output> & .\1
2
HELPTOILETMENPLEASETOILET
YES
PS C:\Users\Eugene\Downloads\output>

```

затратність ~15хв

Завдання №6 – Algotester Lab 1 v3

Задача:

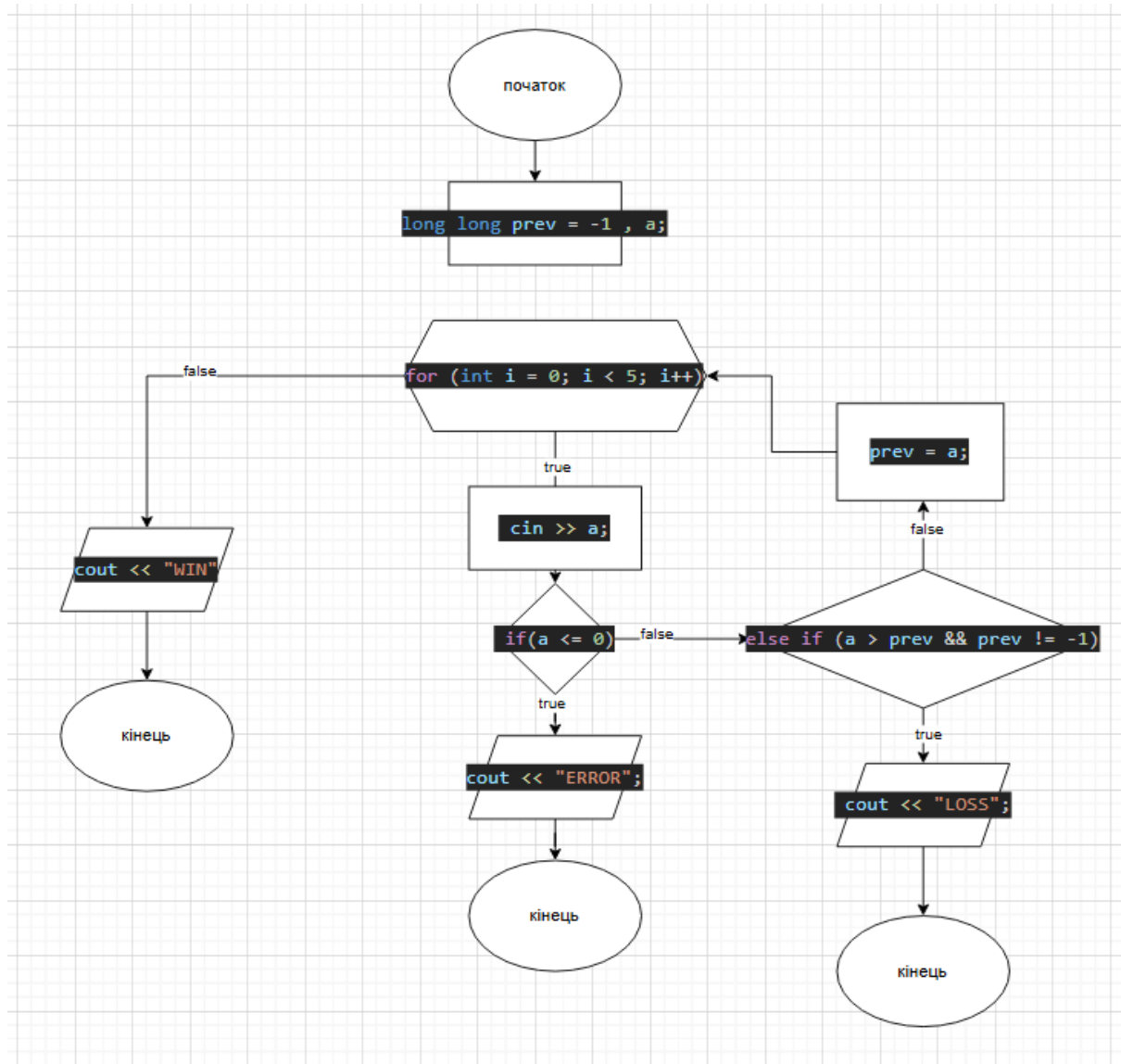
Персонажу по одному дають сторони 5 кубів $a_1..5$, з яких він будує піраміду.

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 $a_{i-1} < a_i$ - це програш.

Ваше завдання - сказати як закінчиться гра.



затратність ~20хв

```
C:\Users\Eugene\Desktop> cd epic_7 & g++ algotester_practice_work_6_eugenie_stefanovich.cpp & main()
1 #include <iostream>
2 #include <vector>
3
4 using namespace std;
5
6 int main()
7 {
8     long long prev = -1, a;
9     for (int i = 0; i < 5; i++){
10         cin >> a;
11         if(a <= 0){
12             cout << "ERROR";
13             exit(0);
14         } else if (a > prev && prev != -1){
15             cout << "LOSS";
16             exit(0);
17         }
18         prev = a;
19     }
20     cout << "WIN";
21
22
23     return 0;
24 }
```

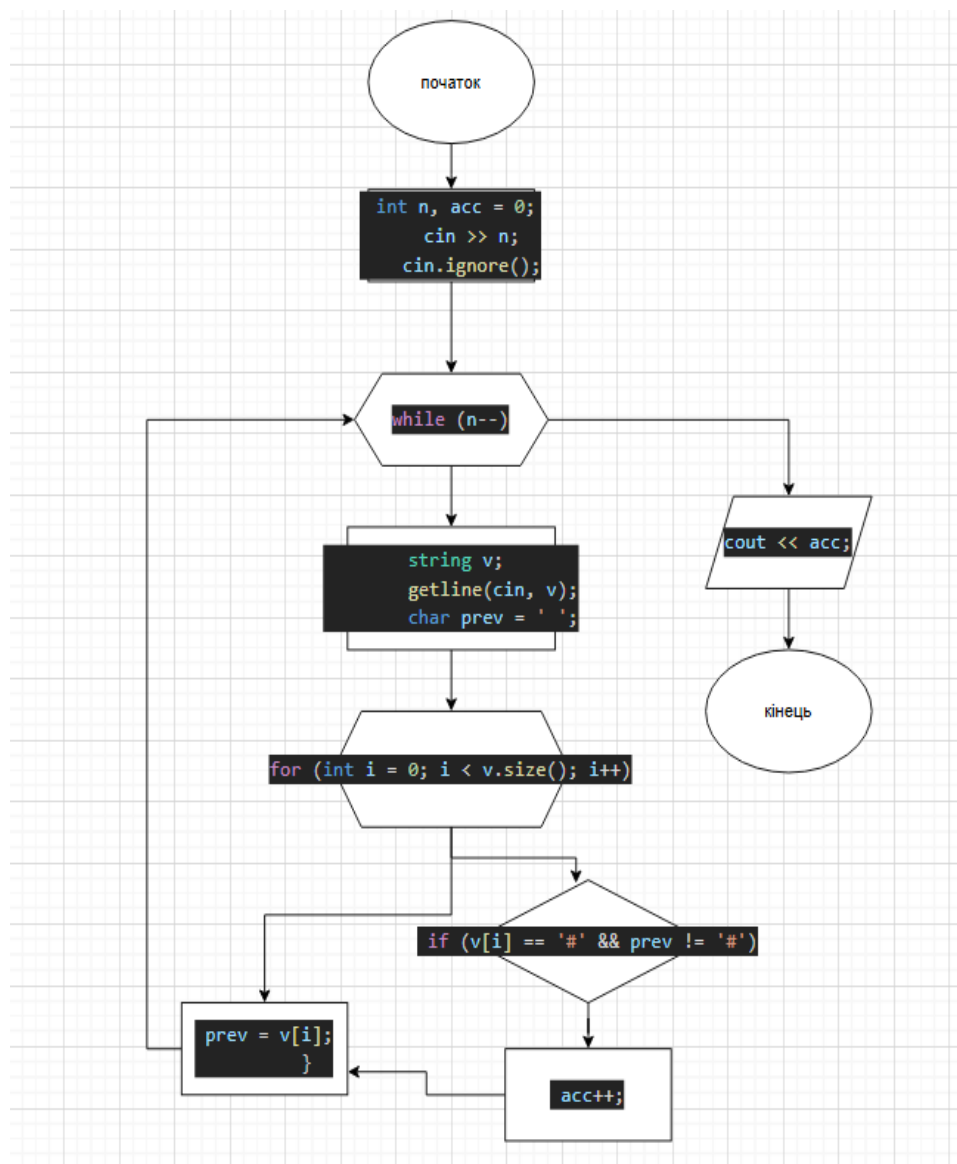
ПРОБЛЕМЫ ВЫХОДНЫЕ ДАННЫЕ КОНСОЛЬ ОТЛАДКИ ТЕРМИНАЛ

PS C:\Users\Eugene\Desktop\epic_7\output> & .\'algotester_practice_work_5_task_eugenie_stefanovich.exe'
10 10 10 10 1 1 1 1
YES
PS C:\Users\Eugene\Desktop\epic_7\output> cd 'c:\Users\Eugene\Desktop\epic_7\output'
PS C:\Users\Eugene\Desktop\epic_7\output> & .\'algotester_practice_work_5_task_eugenie_stefanovich.exe'
10 10 10 10 5 1 1 1
NO
PS C:\Users\Eugene\Desktop\epic_7\output> cd 'c:\Users\Eugene\Desktop\epic_7\output'
PS C:\Users\Eugene\Desktop\epic_7\output> & .\'algotester_practice_work_5_task_eugenie_stefanovich.exe'
10 10 10 10 11 1 1 1
ERROR
PS C:\Users\Eugene\Desktop\epic_7\output> cd 'c:\Users\Eugene\Desktop\epic_7\output'
PS C:\Users\Eugene\Desktop\epic_7\output> & .\'algotester_practice_work_5_task_eugenie_stefanovich.exe'
1 1 1 1 1 1 1 11
ERROR
PS C:\Users\Eugene\Desktop\epic_7\output> cd 'c:\Users\Eugene\Downloads\output'
PS C:\Users\Eugene\Downloads\output> & .\'algotester_practice_work_6_task_andrii_fediv.exe'
10 9 8 7 6
WIN
PS C:\Users\Eugene\Downloads\output>

затратність ~15хв

Завдання №7 – Algotester Lab 3 v1

Ви з'явилися у світі під назвою Атод посеред Пустелі Безправ'я. Так сталося, що Ви попали саме в той час і місце, де ведеться битва між чаклункою Ліною і темними силами, які хочуть знищити цей світ. На жаль, трапилась халепа, бо деякі слова із книги чар були пошкоджені під час битви. Одне таке слово можна відновити виконавши ритуал зцілення над пошкодженими буквами. Ритуал зцілення можна виконати на всіх **підряд** розташованих **пошкоджених** буквах. Вам не залишається нічого іншого як допомогти Ліні відновити ці слова і сказати скільки мінімально треба провести таких ритуалів, щоб прочитати одне з наймогутніших у цьому світі заклять - Поневолення Дракона!



затратність ~20хв

```
C: > Users > Eugene > Desktop > epic_7 > G+ algotester_practice_work_7_task_eugenie_stefanovich.cpp > main()
1  #include <iostream>
2  #include <vector>
3
4  using namespace std;
5  using vint = vector<int>;
6  using ving = vector<string>;
7
8  int main()
9  {
10     int n, acc = 0;
11     cin >> n;
12     cin.ignore();
13
14     while (n--)
15     {
16         string v;
17         getline(cin, v);
18         char prev = ' ';
19         for (int i = 0; i < v.size(); i++)
20         {
21             if (v[i] == '#' && prev != '#')
22                 acc++;
23             prev = v[i];
24         }
25     }
26     cout << acc;
27     return 0;
28 }
```

```
PS C:\Users\Eugene\Desktop\epic_7\output> cd 'c:\Users\Eugene\Desktop\epic_7\output'
PS C:\Users\Eugene\Desktop\epic_7\output> & .\algotester_practice_work_7_task_eugenie_stefanovich.cpp
2
Ukrai#e
Ukrai#e
2
PS C:\Users\Eugene\Desktop\epic_7\output>
file  Debug
```

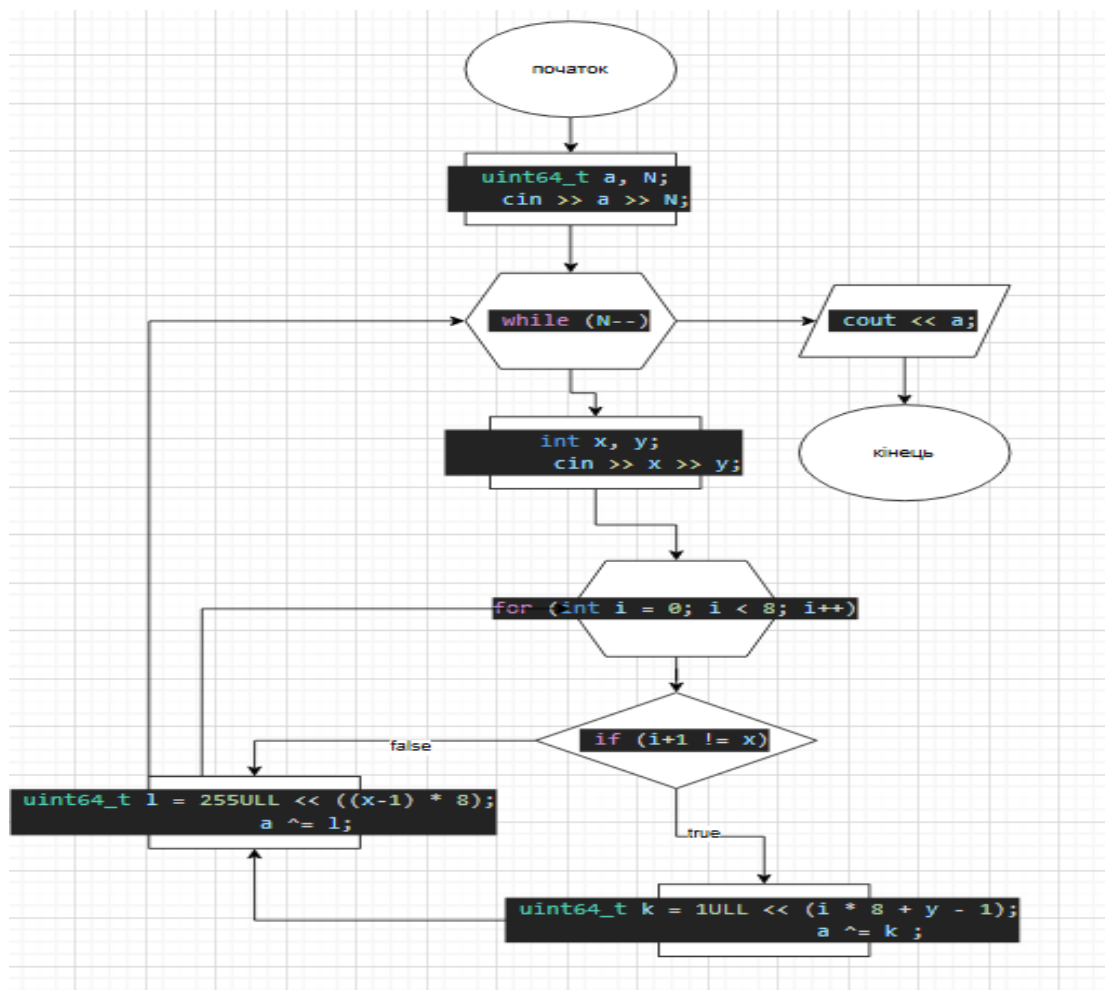
затратність ~25хв

Завдання №8 – Algotester Lab 5 v1

У світі Атод сестри Ліна і Рілай люблять грати у гру. У них є дошка із 8-ми рядків і 8-ми стовпців. На перетині ii -го рядка і jj -го стовпця лежить магічна куля, яка може світитись магічним світлом (тобто у них є 64 кулі). На початку гри деякі кулі світяться, а деякі ні... Далі вони обирають N куль і для кожної читають магічне заклиння, після чого всі кулі, які лежать на перетині стовпця і рядка обраної кулі міняють свій стан (ті що світяться - гаснуть, ті, що не світяться - загораються). Також вони вирішили трохи Вам допомогти і придумали спосіб як записати стан дошки одним числом aa із 8-ми байт, а саме (див. Примітки):

- Молодший байт задає перший рядок матриці;
- Молодший біт задає перший стовпець рядку;
- Значення біту каже світиться куля чи ні (0 - ні, 1 - так);

Тепер їх цікавить яким буде стан дошки після виконання N заклинань і вони дуже просять Вас їм допомогти.



затратність ~25хв

```
PS C:\Users\Eugene\Desktop\epic_7\output> &
0 4 1 1 2 2 1 2 2 1
771
PS C:\Users\Eugene\Desktop\epic_7\output>
```

затратність ~30хв

Висновки:

Я покращив свої практичні навички у розробці та аналізі алгоритмів для розв'язання задач.