

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:**

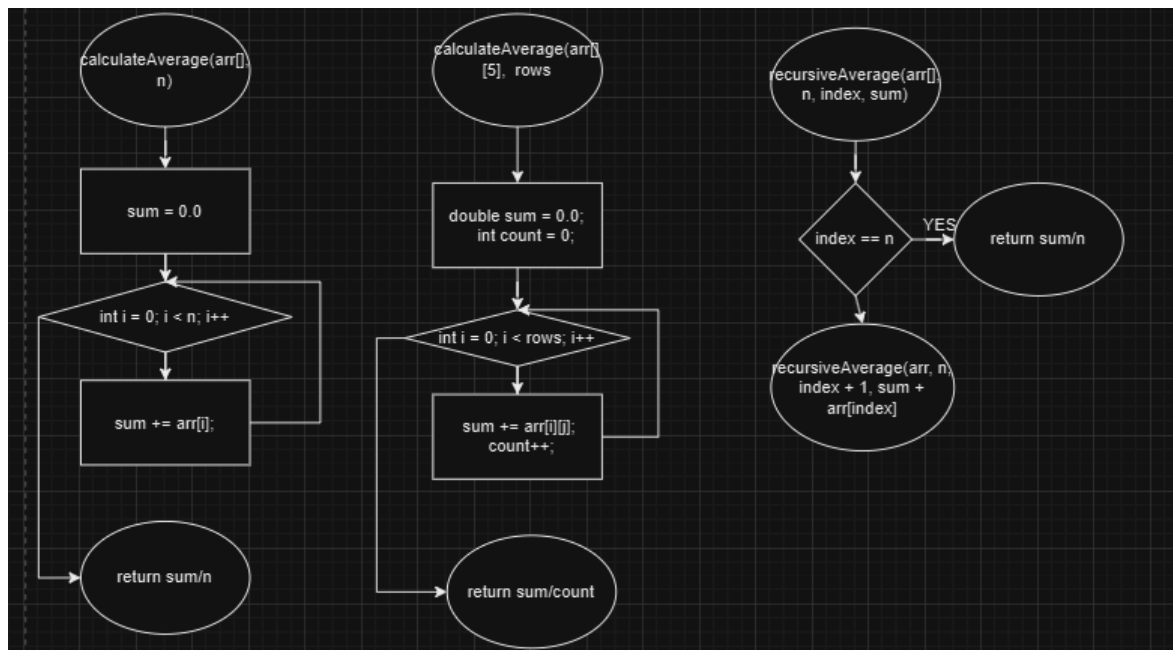
Студент групи ІІІ-12  
Кривичко Назар

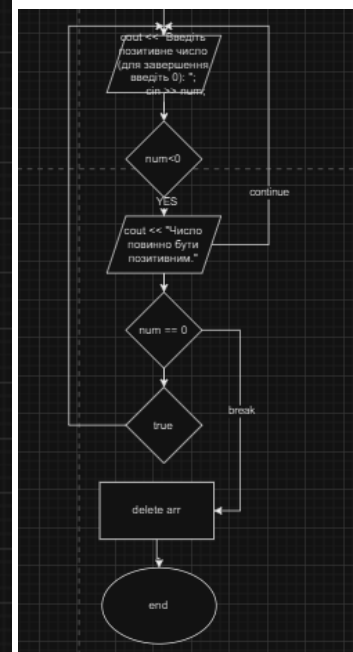
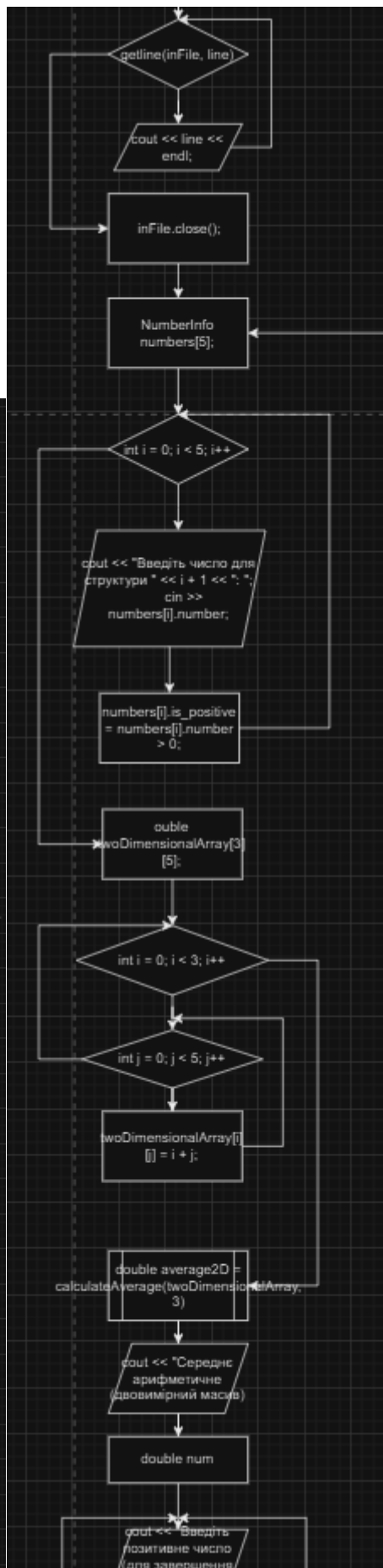
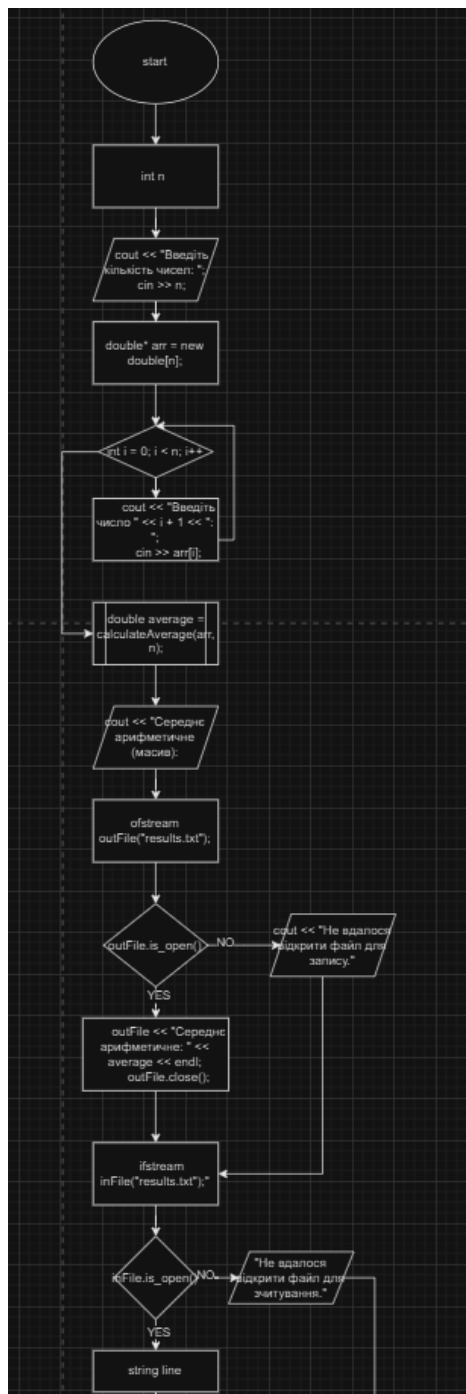
## Розв'язання

### ВНС Завдання 1

**Варіант 18.**  $K = \ln(a + x^3) + \sin^2(x/b)$ ;  $M = e^{-cx} \frac{x + \sqrt[3]{x+a}}{x - \sqrt{|x-b|}}$ , де

$a=10,2$ ;  $b=9,3$ ;  $x=2,4$ ;  $c=0,5$ .





```

1 // var 18
2 #include <iostream>
3 #include <cmath>
4 #include <fstream>
5 using namespace std;
6
7 // Оголошення структури для зберігання інформації про число
8 struct NumberInfo {
9     double number;
10    bool is_positive;
11 };
12
13 // Функція для обчислення середнього арифметичного
14 double calculateAverage(double arr[], int n) {
15     double sum = 0.0;
16     for (int i = 0; i < n; i++) {
17         sum += arr[i];
18     }
19     return sum / n;
20 }
21
22 // Переопрацьована функція для обчислення середнього арифметичного в 2D масиві
23 double calculateAverage2D(double arr[1][5], int rows) {
24     double sum = 0.0;
25     int count = 0;
26     for (int i = 0; i < rows; i++) {
27         for (int j = 0; j < 5; j++) {
28             sum += arr[i][j];
29             count++;
30         }
31     }
32     return sum / count;
33 }
34
35 // Рекурсивна функція для обчислення середнього арифметичного
36 double recursiveAverage(double arr[], int n, int index = 0, double sum = 0.0) {
37     if (index == n) return sum / n;
38     return recursiveAverage(arr, n, index + 1, sum + arr[index]);
39 }
40
41 int main() {
42     int n;
43     cout << "Введіть кількість чисел: ";
44     cin >> n;
45
46     double* arr = new double[n]; // використання динамічного масиву (вказівник)
47
48     for (int i = 0; i < n; i++) {
49         cout << "Введіть число " << i + 1 << ": ";
50         cin >> arr[i];
51     }
52
53     // Середнє арифметичне за допомогою масиву
54     double average = calculateAverage(arr, n);
55     cout << "Середнє арифметичне (масив): " << average << endl;
56
57     // Запис результату у файл
58     ofstream outfile("results.txt");
59     if (outfile.is_open()) {
60         outfile << "Середнє арифметичне: " << average << endl;
61         outfile.close();
62     } else {
63         cout << "Не вдалося відкрити файл для запису." << endl;
64     }
65
66     // Зчитування результату з файлу
67     ifstream infile("results.txt");
68     if (infile.is_open()) {
69         string line;
70         while (getline(infile, line)) {
71             cout << line << endl;
72         }
73         infile.close();
74     } else {
75         cout << "Не вдалося відкрити файл для зчитування." << endl;
76     }
77
78     // Структура для чисел
79     NumberInfo numbers[5];
80     for (int i = 0; i < 5; i++) {
81         cout << "Введіть число для структури " << i + 1 << ": ";
82         cin >> numbers[i].number;
83         numbers[i].is_positive = numbers[i].number > 0;
84     }
85
86     // Двовимірний масив
87     double twoDimensionalArray[3][5];
88     for (int i = 0; i < 3; i++) {
89         for (int j = 0; j < 5; j++) {
90             twoDimensionalArray[i][j] = i + j;
91         }
92     }
93
94     // Середнє арифметичне для двовимірної масиву
95     double average2D = calculateAverage2D(twoDimensionalArray, 3);
96     cout << "Середнє арифметичне (двовимірний масив): " << average2D << endl;
97
98     // Цикл do-while для введення числа
99     double num;
100    do {
101        cout << "Введіть позитивне число (для завершення введіть 0): ";
102        cin >> num;
103        if (num < 0) {
104            cout << "Число повинно бути позитивним." << endl;
105            continue; // Пропускаємо залишок циклу, якщо число від'ємне
106        }
107        if (num == 0) break; // Вихід з циклу
108    } while (true);
109
110    // Очищення динамічно виділеної пам'яті
111    delete[] arr;
112 }
113

```

```

Введіть кількість чисел: 3
Введіть число 1: 1
Введіть число 2: 2
Введіть число 3: 3
Середнє арифметичне (масив): 2
Середнє арифметичне: 2
Введіть число для структури 1: 5.5
Введіть число для структури 2: -1
Введіть число для структури 3: 4
Введіть число для структури 4: 6.2
Введіть число для структури 5: -6.2
Середнє арифметичне (двовимірний масив): 3
Введіть позитивне число (для завершення введіть 0): 5
Введіть позитивне число (для завершення введіть 0): 2
Введіть позитивне число (для завершення введіть 0): 7
Введіть позитивне число (для завершення введіть 0): 4
Введіть позитивне число (для завершення введіть 0): 0
PS C:\Users\user\Desktop\c++\saga1>

```

## ВНС Завдання 2

**Варіант 13.**  $z = \begin{cases} \ln \sqrt{ax^2 + by^3}, & xy > 0, \\ a^x + b^y, & xy \leq 0; \end{cases}$  де  $x \in [-1, 1]$ ;  $h_x = 0,9$ ;

$a = 1,7$ ;  $b = 2,3$ ;  $y \in [-2, 2]$ ;  $h_y = 1$ .

```

vns_practice_work_2_task_stefan_shyika.cpp > main()
1 // var 13
2 #include <iostream>
3 #include <cmath>
4
5 int main() {
6     double a = 1.7, b = 2.3;
7     double hx = 0.9, hy = 1.0;
8
9     for (double x = -1; x <= 1; x += hx) {
10         for (double y = -2; y <= 2; y += hy) {
11             double z;
12             if (x * y > 0) {
13                 z = log(sqrt(a * pow(x, 2) + b * pow(y, 3)));
14             } else {
15                 z = pow(a, x) + pow(b, y);
16             }
17             std::cout << "z = " << z << std::endl;
18         }
19     }
20 }

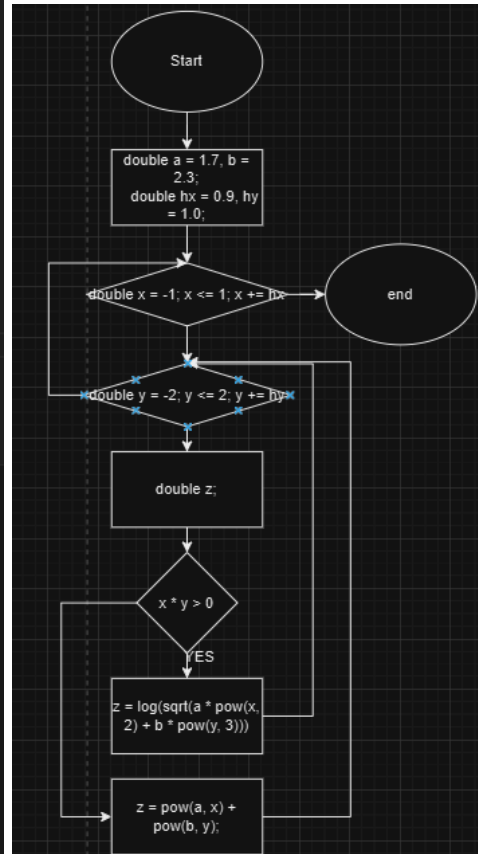
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

z = nan
z = nan
z = 1.58824
z = 2.88824
z = 5.87824
z = nan
z = nan
z = 1.94832
z = 3.24832
z = 6.23832
z = 1.71787
z = 1.96361
z = 2.52883
z = 0.61012
z = 1.4849
PS C:\Users\user\Desktop\c++\saga1>

```



## ВНС Завдання 3

**Варіант 8.** Обчислення опору електричного ланцюга, що складається з двох паралельно сполучених опорів. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Обчислення опору електричного ланцюга при паралельному з'єднанні елементів.

Введіть початкові дані:

Величина першого опору (Ом) > **15**

Величина другого опору (Ом) > **20**

Опір ланцюга: 8.57 Ом

Обчислення об'єму конуса  $S = \frac{1}{3} \cdot \pi \cdot r^2 \cdot h$ .

```

vns_practice_work_3_task_stefan_shyika.cpp > main()
1 // var 8
2 #include <iostream>
3 #include <cmath>
4 using namespace std;
5
6 int main() {
7     double R1, R2, R_total;
8     cout << "Введіть величину першого опору (Ом) > ";
9     cin >> R1;
10    cout << "Введіть величину другого опору (Ом) > ";
11    cin >> R2;
12
13    R_total = 1 / ((1 / R1) + (1 / R2));
14    cout << "Опір ланцюга: " << R_total << " Ом" << endl;
15
16    double r, h, volume;
17    cout << "Введіть радіус основи конуса (см): ";
18    cin >> r;
19    cout << "Введіть висоту конуса (см): ";
20    cin >> h;
21
22    volume = (1.0 / 3.0) * M_PI * pow(r, 2) * h;
23    cout << "Об'єм конуса: " << volume << " см^3" << endl;
24 }
25

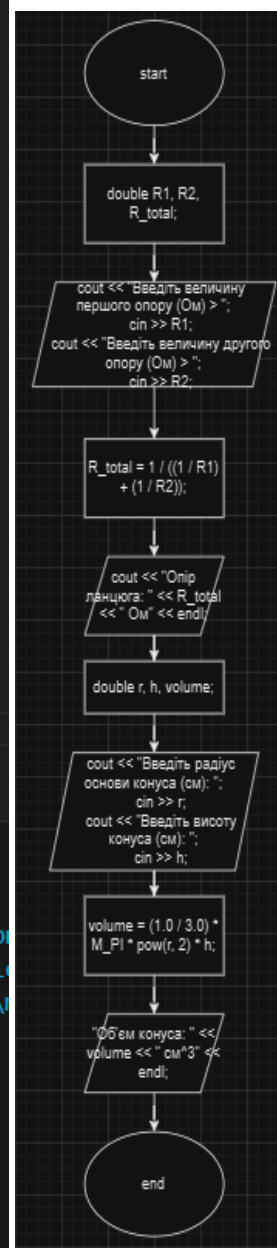
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS C:\Users\user\Desktop\c++\saga1> & 'c:\Users\user\.vscode\extension
auncher.exe' '--stdin=Microsoft-MIEngine-In-ummmmlhao.ubm' '--stdout=Mi
lb3bk1v.1or' '--pid=Microsoft-MIEngine-Pid-jiophfbu.ljp' '--dbgExe=C:\
Введіть величину першого опору (Ом) > 3
Введіть величину другого опору (Ом) > 6
Опір ланцюга: 2 Ом
Введіть радіус основи конуса (см): 2
Введіть висоту конуса (см): 7
Об'єм конуса: 29.3215 см^3
PS C:\Users\user\Desktop\c++\saga1> a

```



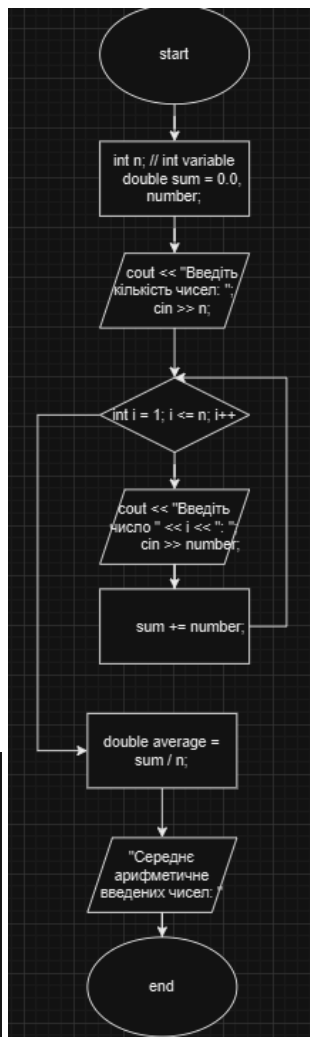
## ВНС Завдання 4

**Варіант 13.** Скласти програму, яка обчислює середнє арифметичне послідовності дробових чисел, яка вводиться з клавіатури. Кількість чисел повинна задаватися під час роботи програми.

```

1 // var 13
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     int n; // int variable
7     double sum = 0.0, number;
8
9     cout << "Введіть кількість чисел: ";
10    cin >> n;
11    int i = 1;
12
13    while(i <= n) {
14        cout << "Введіть число " << i << ": ";
15        cin >> number;
16        sum += number;
17        i++;
18    }
19
20    double average = sum / n;
21    cout << "Середнє арифметичне введених чисел: " << average << endl;
22 }
23

```



```

Введіть кількість чисел: 3
Введіть число 1: 1
Введіть число 2: 2
Введіть число 3: 3
Середнє арифметичне введених чисел: 2
PS C:\Users\user\Desktop\c++\saga1>

```



# Алготестер 132

## Назва для покемона

*Limits: 2 sec., 256 MiB*

Тарас знайшов новий вид покемонів, і тепер йому залишилося тільки придумати назву для нього. Ваше завдання — допомогти йому в цьому.

Наразі у Тараса є рядок  $t$ , який складається із символів англійського алфавіту, а також знаків запитання. Для того, щоб завершити процес придумування назви, потрібно кожен знак запитання замінити на довільний символ.

Крім цього відомо, що новий вид споріднений з іншим, давно відкритим видом під назвою  $s$ . Оскільки Тарас хоче, щоб назва нового виду відображала цю спорідненість, вона має містити  $s$  як підрядок.

Визначте, чи може Тарас замінити знаки запитання в назві  $t$  на символи таким чином, щоб відобразити спорідненість двох покемонів у назві.

### Input

У першому та другому рядках задано два рядки  $t$  та  $s$ .

Гарантується, що перший рядок складається з великих латинських літер та знаків запитання, а другий — лише з великих латинських літер.

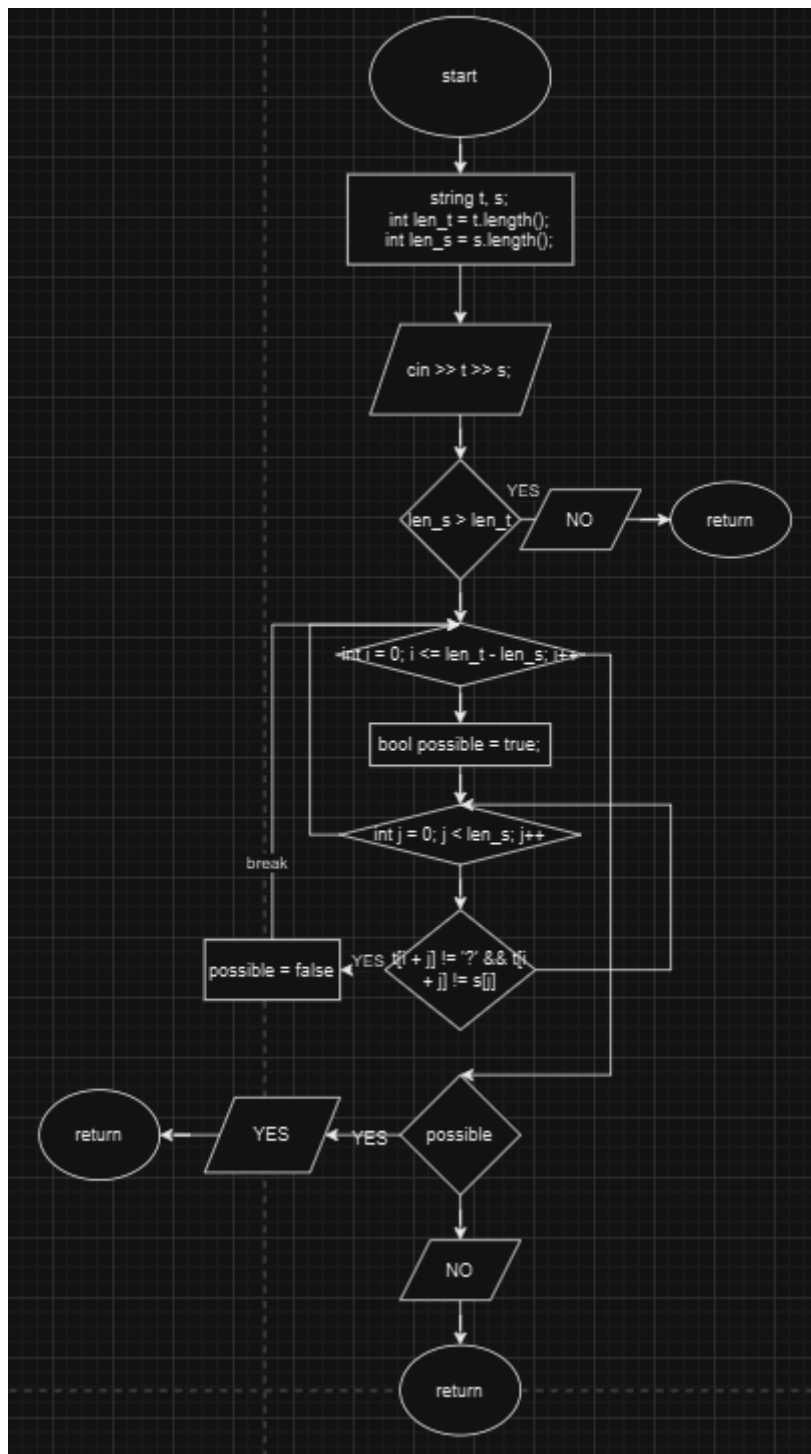
### Output

В єдиному рядку виведіть **YES**, якщо Тарас може придумати назву для нового покемона, або **NO** в протилежному разі.

```

1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  int main() {
6      string t, s;
7      cin >> t >> s;
8      int len_t = t.length();
9      int len_s = s.length();
10
11     if (len_s > len_t) {
12         cout << "NO" << endl;
13         return 0;
14     }
15
16     for (int i = 0; i <= len_t - len_s; i++) {
17         bool possible = true;
18         for (int j = 0; j < len_s; j++) {
19             if (t[i + j] != '?' && t[i + j] != s[j]) {
20                 possible = false;
21                 break;
22             }
23         }
24         if (possible) {
25             cout << "YES" << endl;
26             return 0;
27         }
28     }
29
30     cout << "NO" << endl;
31 }
32

```



15 minutes ago

0132 - Назва для покемона

C++ 23

Accepted

0.002

1.184

1864564

## Алготестер 354

### Куди бігти?

Limits: 2 sec., 256 MiB

У цей вечір Володя і Влад, як і всі геймери кожного понеділка, ішли на базу після чергового квесту. Вони саме думали, на яку локацію їм слід піти: на ту, що розташована вниз по карті, чи на ту, що вище. Якщо друзі підуть униз, то їхня звичайна швидкість подвоїться, а якщо вгору, то вона навпаки — зменшиться в два рази.

Перша локація розташована за  $s_d$  метрів від юних програмістів вниз по карті, а друга за  $s_u$  метрів відповідно вгору. Також відома звичайна швидкість Володі і Влада —  $v$  км/год. Друзі просять вас допомогти їм і сказати, до якої локації вони доберуться швидше.

#### Input

У першому рядку задано три цілих числа через пробіл  $s_d$ ,  $s_u$  та  $v$  — відстань до першої та другої локації, і звичайна швидкість геймерів.

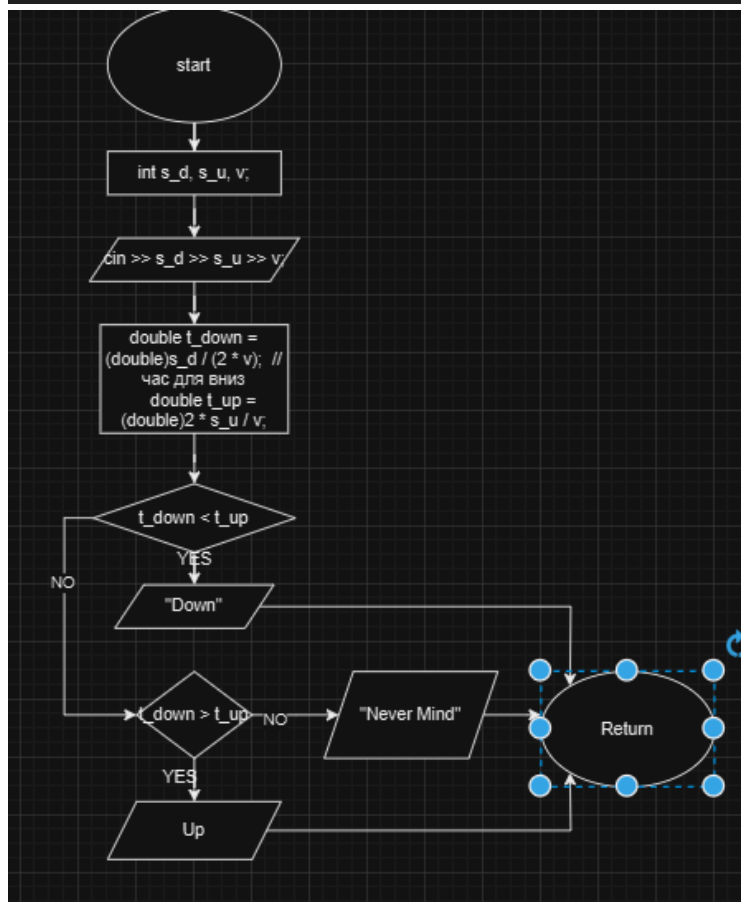
#### Output

У єдиному рядку виведіть куди варто рухатися хлопцям вгору (Up) або вниз (Down), щоб найшвидше дійти до локації. Якщо час, за який вони можуть дійти до обох локацій рівний, виведіть Never mind.

```

1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int s_d, s_u, v;
6      cin >> s_d >> s_u >> v;
7
8      double t_down = (double)s_d / (2 * v); // час для вниз
9      double t_up = (double)2 * s_u / v;      // час для вгору
10
11     if (t_down < t_up) {
12         cout << "Down" << endl;
13     } else if (t_down > t_up) {
14         cout << "Up" << endl;
15     } else {
16         cout << "Never mind" << endl;
17     }
18 }

```



a few seconds ago

0354 - Куди бігти?

C++ 23

Accepted

0.002

1.000

1864565

## Алготестер 541

Мегапопулярний Дзідзьо дуже любить складати мегапопулярні пісні. Запис його пісень є специфічним: він записує всі слова своєї пісні без пропусків та розділових знаків у єдиний рядок.

Одного разу, переглядаючи текст пісні, Дзідзьо видалив один символ з неї і помітив, що пісня стала зовсім іншою. Допитливому Дзідзьо стало цікаво, скільки різних пісень може утворитись із заданої, якщо видалити з неї тільки один символ. Але Дзідзьо не має часу для того, щоб це рахувати. Допоможіть йому, напишіть для цього програму.

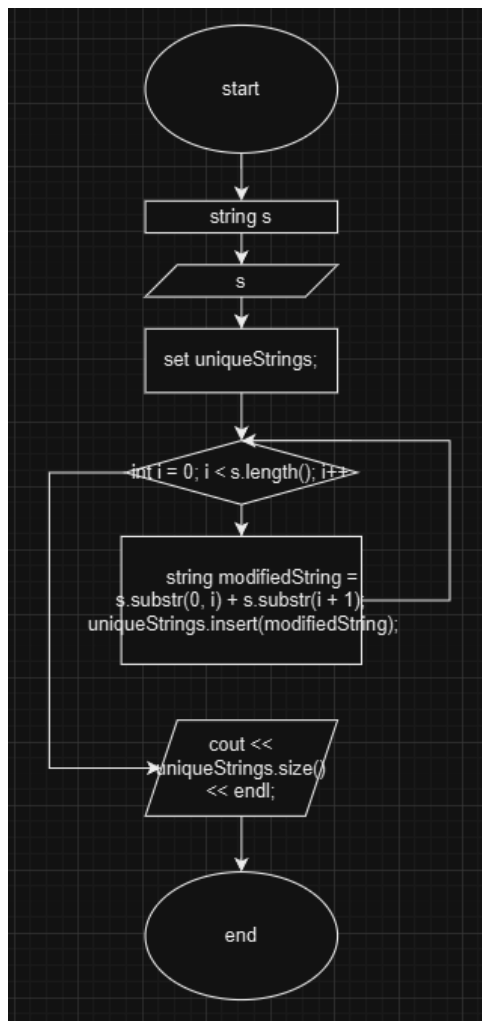
### Input

У першому і єдиному рядку задано рядок  $s$ . Він складається тільки з малих літер англійського алфавіту ( $a - z$ ).

### Output

У єдиному рядку виведіть одне ціле число — відповідь на задачу.

```
1  #include <iostream>
2  #include <set>
3  using namespace std;
4
5  int main() {
6      string s;
7      cin >> s;
8
9      set<string> uniqueStrings;
10
11     for (int i = 0; i < s.length(); i++) {
12         string modifiedString = s.substr(0, i) + s.substr(i + 1); // Видаляємо i-й символ
13         uniqueStrings.insert(modifiedString);
14     }
15     cout << uniqueStrings.size() << endl;
16 }
17
```



2 minutes ago

0541 - Дзідзьо і його пісня

C++ 23

Accepted

0.003

1.188

1864566

# Алготестер 877

Одного разу Дмитрик знайшов пристрій, який при натисканні кнопки «Показати» показує на екрані якесь ціле додатне число. Кожного разу, коли він бачить на екрані число менше або рівне за усі попередні числа (або перше число) він вигукує «Вау!».

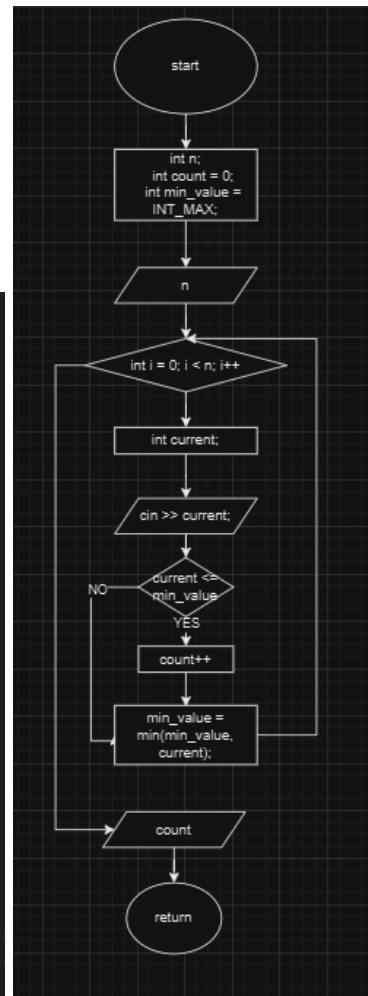
Відомо, що Дмитрик натискав кнопку «Показати»  $n$  раз. Скільки раз він вигукнув «Вау!»?

## Input

У першому рядку задано одне ціле число  $n$  — кількість натискань на кнопку.

У другому рядку задано  $n$  цілих чисел у тій послідовності, в якій вони появлялися на екрані в Дмитрика.

```
1  ✓ #include <iostream>
2  ✓ #include <limits>
3  ✓ using namespace std;
4
5  ✓ int main() {
6      int n;
7      cin >> n;
8      int count = 0;
9      int min_value = INT_MAX;
10
11  ✓ for (int i = 0; i < n; i++) {
12      int current;
13      cin >> current;
14
15  ✓      if (current <= min_value) {
16          count++;
17      }
18
19      min_value = min(min_value, current);
20  }
21
22  cout << count << endl;
23  }
24
```



a few seconds ago	Стефан Шийка	0877 - Мінімальні вигукки	C++ 23	Accepted	0.408	1.535	1864571
-------------------	--------------	---------------------------	--------	----------	-------	-------	---------

## Pull

**Висновок:** на цій лабораторній я навчився писати базові алгоритми та блок схеми до задач на c++.