

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-11
Цяпа Остап Андрійович

Львів 2024

Мета роботи:

Опанувати вивчений матеріал на практиці, а саме попрацювати з файлами як у мові C та C++, вдосконалити роботу зі стрічками в C++.

Теоретичні відомості:

- Теми пройдені під час семестру

Джерела використані для опрацювання теоретичного матеріалу:

- Ознайомився під час навчання

Виконання роботи:

1. Опрацювання завдання до програм.

Завдання №1

VNS PRACTICE WORK TASK 1 (VARIANT 17)

Мета роботи: одержати практичні навички в розробці і дослідженні алгоритмів розв'язання задач.

Розробити лінійний алгоритм для розв'язання задачі.

Варіант 17. $Y = b^3 \operatorname{tg}^2 x - \frac{a}{\sin^2(x/a)}$; $Z = ae^{-\sqrt{3}} \cos(bx/a)$, де
 $a=3,2$; $b=17,5$; $x=-4,8$.

Завдання №2

VNS PRACTICE WORK TASK (VARIANT 9)

Розробити алгоритм, що розгалужується.

Варіант 9. $y = \begin{cases} \sin|ax + b^a|, & x < b, \\ \cos|ax - b^a|, & x \geq b; \end{cases}$

де $x \in [0,5;3,1]$; $h_x = 0,21$; $a \in [-0,1;-1,4]$; $h_a = 0,1$; $b = 1,5$.

Завдання №3

VNS PRACTICE WORK TASK 3 (VARIANT 22)

Написати програму згідно свого варіанту.

Варіант 22. Обчислення вартості покупки складається з декількох зошитів і такої ж кількості обкладинок до них. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, які вводяться користувачем, виділені напівжирним шрифтом).

Обчислення вартості покупки.

18

Введіть початкові дані:

Ціна зошита (грн.) > **2.75**

Ціна обкладинок (грн.) > **0.5**

Кількість комплектів (грн.) > **7**

Вартість покупки: 15.45 грн.

Реалізувати виконання розрахунків: Верстат –автомат виготовляє в 1 годину КОЛ шайб. Скільки шайб він виготовить за 8-ї годинну зміну, за робочий день при двозмінній роботі і за 30-ти денний місяць, якщо в місяці 4 вихідних дня і в кожній зміні верстат знаходиться 1 годину на профілактиці?

Завдання №4

VNS PRACTICE WORK TASK 4 (VARIANT 4)

Написати програму згідно свого варіанту.

Написати програму, яка виводить таблицю

квадратів перших п'яти цілих позитивних непарних чисел.

Завдання №5

SELF PRACTICE WORK 1 ALGOTESTER (Марічка і печиво)

Зібралися Зеник і Марічка разом з пластунами в похід. Похід — серйозна справа. Потрібно запастись продуктами харчування та розподілити їх споживання по днях так, щоб всім вистачило. Цього разу Зеник слідкує за тим, щоб печива вистачило аж до останнього дня походу. Зеник чітко знає, скільки пачок печива повинно залишитись кожного дня, і щовечора перераховує їх. Якщо Зеник побачить, що залишилось менше пачок, ніж повинно залишитись за його розрахунками, він неодмінно знайде того, хто з'їв забагато печива, і покарає його.

Марічка дуже любить печиво. Сьогодні, коли всі пластуни покинуть свої намети і підуть купатися в річку, Марічка планує непомітно з'їсти трохи печива. Звісно, Марічка не хоче бути покараною і дуже боїться, щоб Зеник не помітив пропажу.

Марічка підгледіла, скільки пачок печива є в рюкзаку Зеника. Також вона знає, скільки штук в кожній пачці. Марічці не терпиться дізнатися, скільки ж печива вона зможе з'їсти так, щоб Зеник не помітив. Зеник помітить пропажу печива з деякої пачки тоді і тільки тоді, коли Марічка повністю спустошить її.

Вхідні дані

У першому рядку задано одне натуральне число n — кількість пачок печива.
У другому рядку задано n натуральних чисел a_i — кількість штук печива в i -й пачці.

Вихідні дані

У єдиному рядку виведіть одне ціле число — максимальну кількість штук печива, яку зможе з'їсти Марічка так, щоб Зеник не помітив цього.

Завдання №6

SELF PRACTICE WORK 2 ALGOTESTER (Борщ, картопля і салат)

Сьогодні в пластунів свято! Привезли вдвічі більше продуктів на обід. Зеник разом із Марічкою зголосилися допомагати на кухні. Всього є три різні страви: борщ, картопля і салат.

У Пласті в Зеника та Марічки є n друзів. i -й друг хоче з'їсти a_i грамів борщу, b_i грамів картоплі та c_i грамів салату. Якщо друг Зеника та Марічки хоче з'їсти x грамів певної страви, а йому в тарілку поклали y грамів, то ступінь недовіри до Зеника з Марічкою в цього друга зростає на $|x - y|$ одиниць. Зенику з Марічкою наказали накладати всім однакові порції певної страви. Тобто кожному другу потрібно накласти A грамів борщу, B грамів картоплі та C грамів салату. Допоможіть Зенику та Марічці обрати такі A , B та C , щоб сумарний ступінь недовіри всіх друзів був якомога меншим.

Вхідні дані

У першому рядку задано одне ціле число n — кількість друзів Зеника та Марічки.
У наступних np рядках задано по три цілі числа a_i , b_i , c_i — кількість борщу, картоплі та салату, яку хоче отримати i -ий друг.

Вихідні дані

У єдиному рядку виведіть одне ціле число — сумарний ступінь недовіри всіх друзів до Зеника та Марічки.

Завдання №7

SELF PRACTICE WORK 4 ALGOTESTER (Торт для Петрика)

Зовсім скоро в Петрика день народження, і він з нетерпінням чекає цього свята. Його друзі слоненята вирішили зробити йому подарунок і приготувати ідеальний торт.

На думку слоненят, ідеальний торт має складатися з рівно n ярусів, кожен з яких є циліндром. Проте при визначенні ідеальності висоти циліндрів не важливі, важливі лише їхні радіуси, тому яруси можна вважати кругами. Яруси кладуть один на одного від найбільшого до найменшого так, щоб вони мали спільний центр. Оскільки всі яруси є однаково гарними, і важливістю жодного не можна нехтувати, торт вважається ідеальним тільки тоді, коли видимі площі всіх nn ярусів рівні, якщо дивитися на торт зверху.

Слоненята вирішили, що радіус найбільшого ярусу їхнього торта має бути r . Допоможіть слоненятам обчислити, яким же буде радіус найменшого.

Вхідні дані

У єдиному рядку задано два цілих числа n та r — відповідно кількість ярусів та радіус найбільшого ярусу в ідеальному торті.

Вихідні дані

У єдиному рядку виведіть дійсне число — радіус найменшого ярусу. Відповідь уважатиметься правильною, якщо її абсолютна чи відносна похибка не перевищуватиме 10^{-4} .

Завдання №8

SELF PRACTICE WORK 3 ALGOTESTER (Зуби)

Мале Бісеня любить гострити зуби. А Зла Тітонька любить до нього підходити і питатися: «Що, зуби гостриш?». Бісеняті таке не дуже подобається, тому воно придумало робити таке.

У Малого Бісеняти є nn зубів. Кожен зуб має коефіцієнт загостреності $a_i a_i$. Також існує межа загостреності $k k$. Якщо коефіцієнт загостреності певного зуба є більшим чи рівним межі загостреності, то такий зуб вважається загостреним. Мале Бісеня хоче наступного разу, коли Зла Тітонька його щось запитає, показати їй якнайбільше загострених зубів, що розташовані поспіль.

Допоможіть Малому Бісеняті дізнатися, скільки найбільше зубів воно зможе показати.

Вхідні дані

У першому рядку задані два цілих числа nn та $k k$ — кількість зубів та межа загостреності відповідно.

В другому рядку задано nn цілих чисел $a_i a_i$ — коефіцієнти загостреності зубів.

Вихідні дані

Єдине ціле число — відповідь на задачу.

2. Дизайн та планувальна оцінка часу виконання завдань:

Програма №1

- Блок-схема

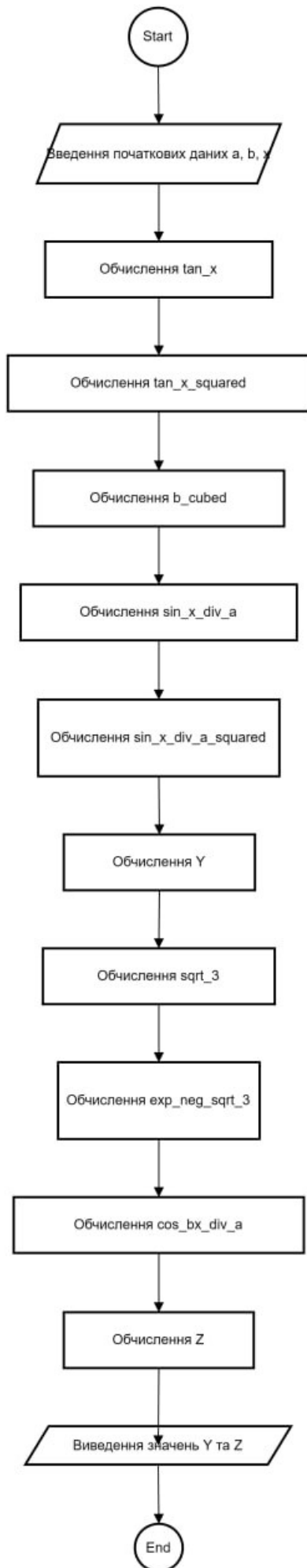


Рисунок 2.1. Блок-схема до програми №1

- Плановий час на реалізацію 25 хвилин.

Програма №2

- Блок-схема

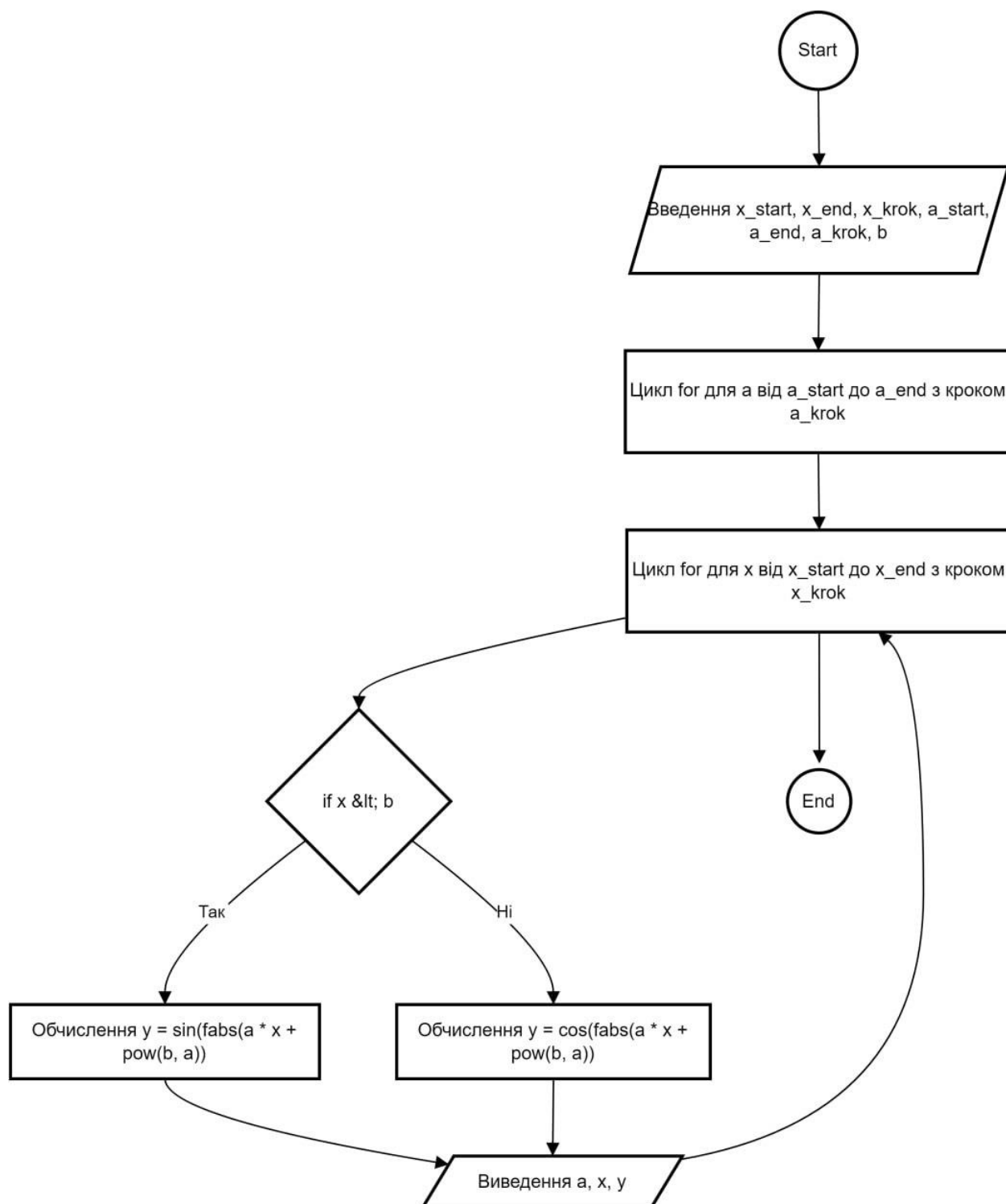


Рисунок 2.2. Блок-схема до програми №2

- Плановий час на реалізацію 35 хвилин.

Програма №3

- Блок-схема

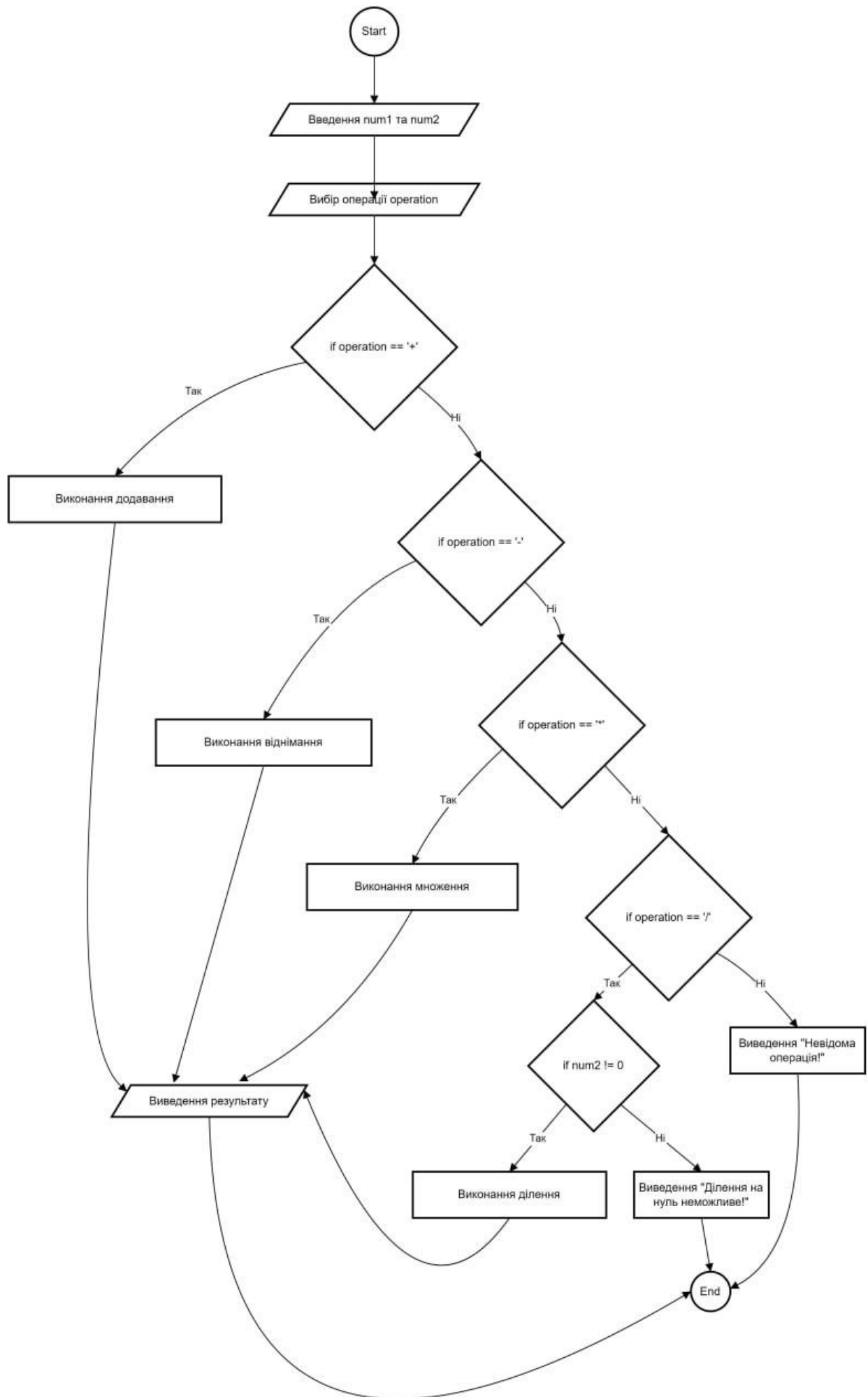


Рисунок 2.3. Блок-схема до програми №3

- Плановий час на реалізацію 55 хвилин.

Програма №4

- Блок-схема

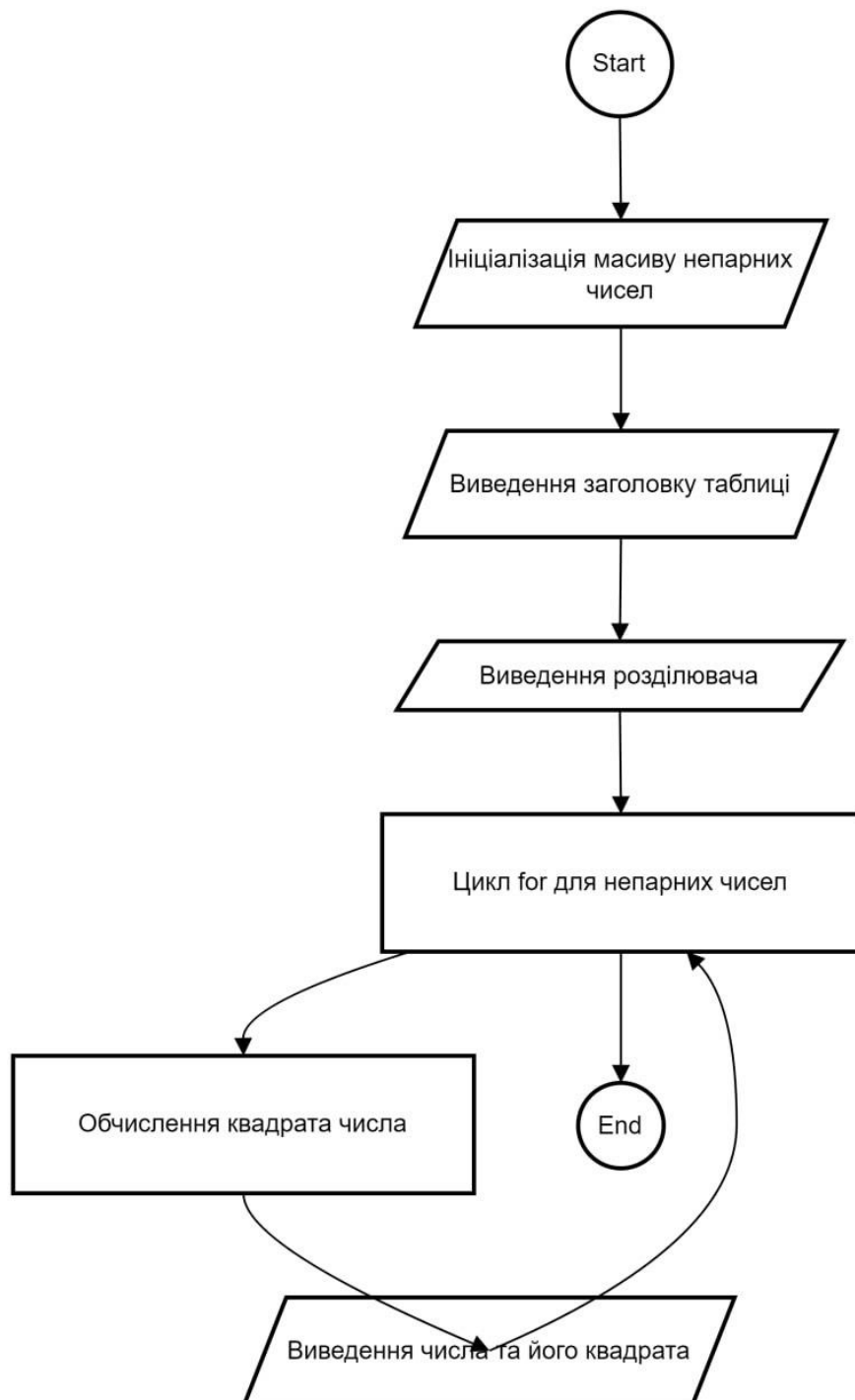


Рисунок 2.4. Блок-схема до програми №4

- Плановий час на реалізацію 40 хвилин.

Програма №5

- Блок-схема

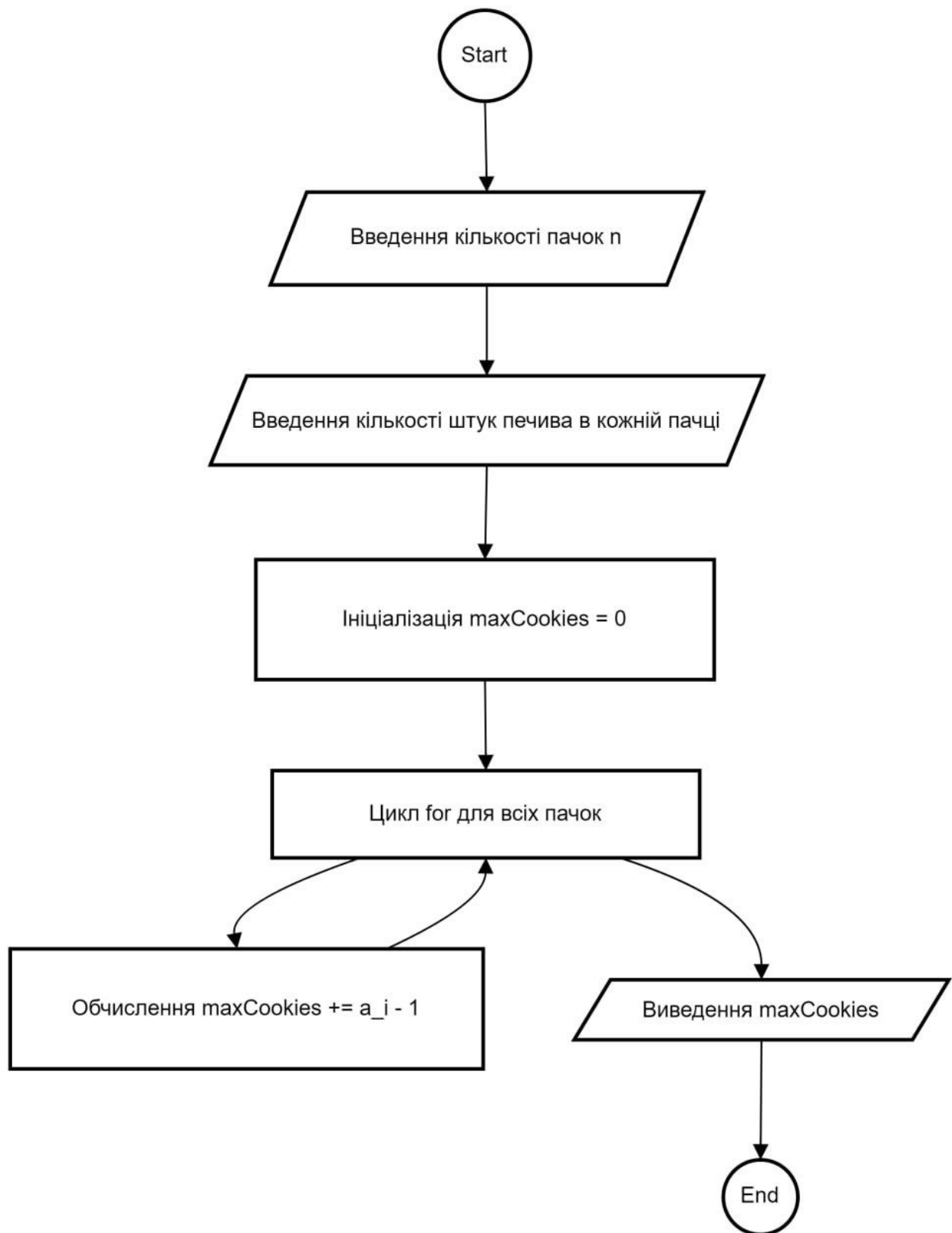


Рисунок 2.5. Блок-схема до програми №5

- Плановий час на реалізацію 20 хвилин.

Програма №6

- Блок-схема

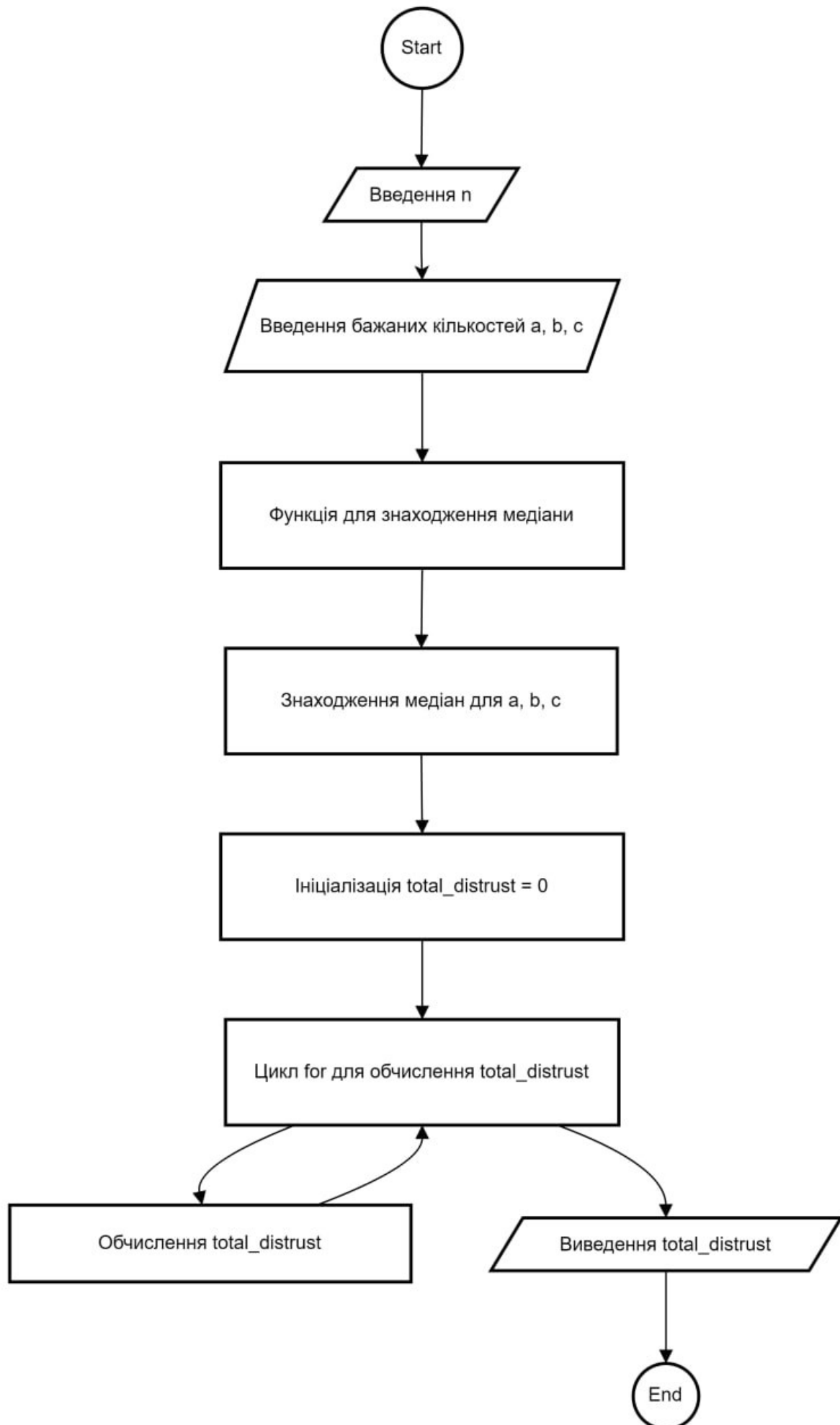


Рисунок 2.6. Блок-схема до програми №6

- Плановий час на реалізацію 40 хвилин.

Програма №7

- Блок-схема

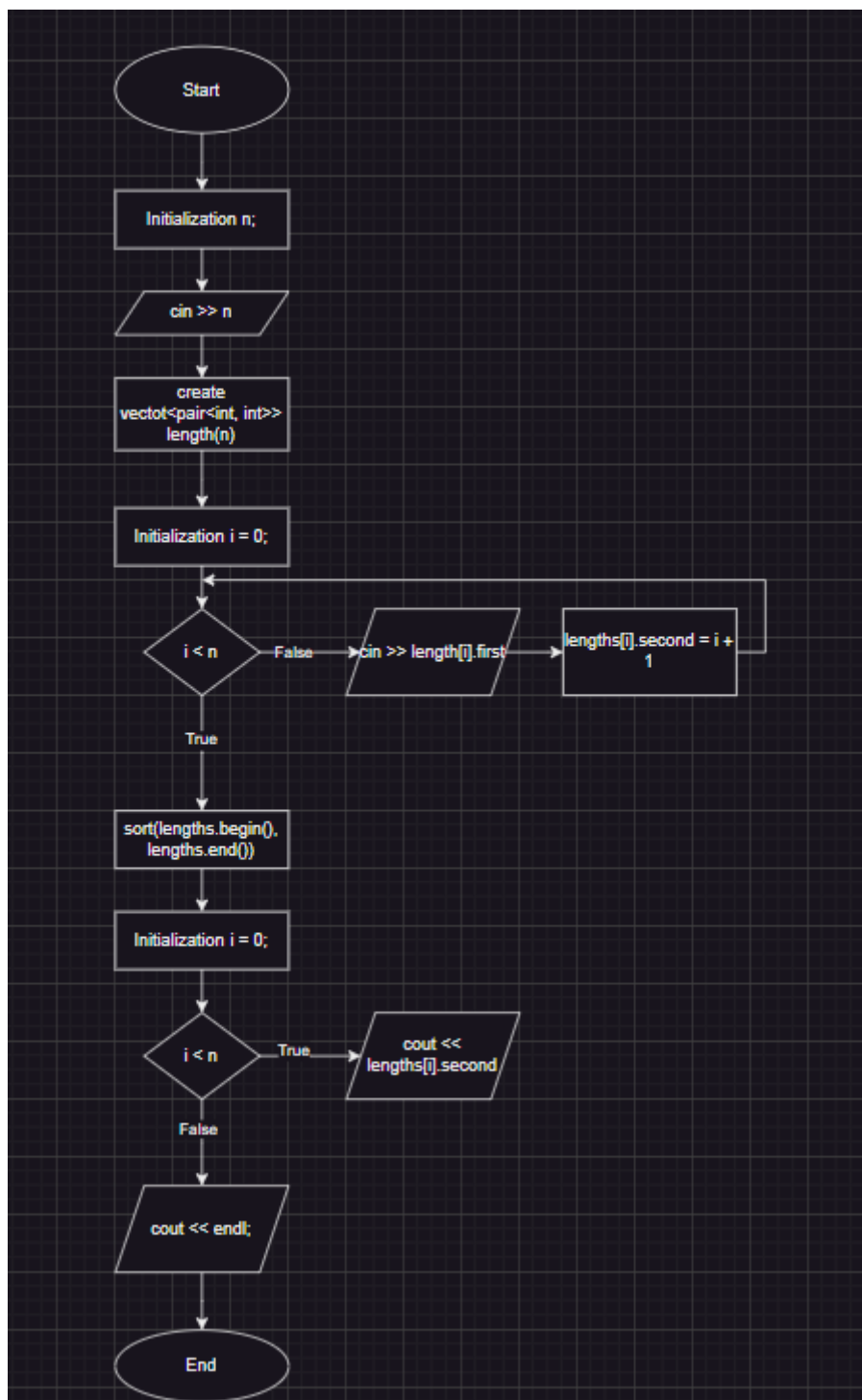


Рисунок 2.7. Блок-схема до програми 7

- Плановий час на реалізацію 25 хвилин.

Програма №8

- Блок-схема

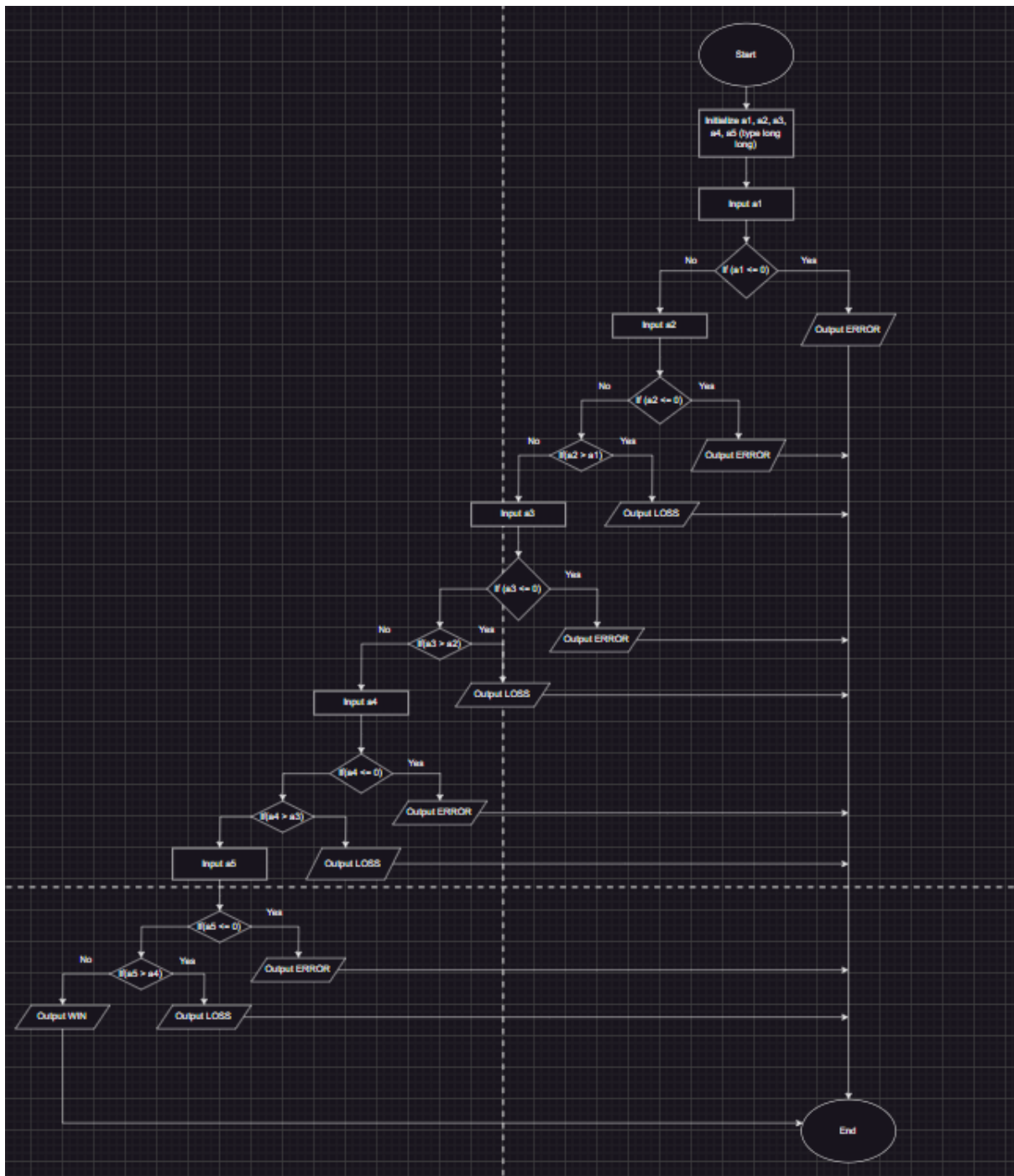


Рисунок 2.8. Блок-схема до програми 8

-Плановий час на реалізацію 35 хвилин.

3. Код програм з посиланням на зовнішні ресурси та фактично затрачений час:

Завдання №1

```
1  #include <iostream>
2  #include <cmath> // Необхідна для обчислень бібліотека
3
4  int main() {
5      // Введення початкових даних
6      double a = 3.2;
7      double b = 17.5;
8      double x = 4.8;
9
10     // Обчислення значення Y
11     double tan_x = tan(x);
12     double tan_x_squared = tan_x * tan_x;
13     double b_cubed = pow(b, 3);
14     double sin_x_div_a = sin(x / a);
15     double sin_x_div_a_squared = sin_x_div_a * sin_x_div_a;
16     double Y = b_cubed * tan_x_squared - (a / sin_x_div_a_squared);
17
18     // Обчислення значення Z
19     double sqrt_3 = sqrt(3.0);
20     double exp_neg_sqrt_3 = exp(-sqrt_3);
21     double cos_bx_div_a = cos((b * x) / a);
22     double Z = a * exp_neg_sqrt_3 * cos_bx_div_a;
23
24     // Виведення результатів
25     std::cout << "Значення Y: " << Y << std::endl;
26     std::cout << "Значення Z: " << Z << std::endl;
27
28     return 0;
29 }
30
```

Рисунок 3.1. Код до програми № 1

```
Значення Y: 694654
Значення Z: 0.248057
```

Рисунок 3.2. Приклад виконання програми № 1

Фактично затрачений час 30 хвилин.

Завдання №2


```

1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4
5  int main() {
6      double x_start = 0.5;
7      double x_end = 3.1;
8      double x_krok = 0.21;
9      double a_start = -0.1;
10     double a_end = -1.4;
11     double a_krok = 0.1;
12     double b = 1.5;
13
14     std::cout << std::fixed << std::setprecision(4); // Встановлення формату виведення
15
16     for (double a = a_start; a >= a_end; a -= a_krok) {
17         for (double x = x_start; x <= x_end; x += x_krok) {
18             double y;
19             if (x < b) {
20                 y = sin(fabs(a * x + pow(b, a)));
21             } else {
22                 y = cos(fabs(a * x - pow(b, a)));
23             }
24             std::cout << "a = " << a << ", x = " << x << ", y = " << y << std::endl;
25         }
26     }
27
28     return 0;
29 }
30

```

Рисунок 3.3. Код до програми № 2

a = -0.1000, x = 0.5000, y = 0.7897
a = -0.1000, x = 0.7100, y = 0.7766
a = -0.1000, x = 0.9200, y = 0.7632
a = -0.1000, x = 1.1300, y = 0.7495
a = -0.1000, x = 1.3400, y = 0.7354
a = -0.1000, x = 1.5500, y = 0.4399
a = -0.1000, x = 1.7600, y = 0.4210
a = -0.1000, x = 1.9700, y = 0.4018
a = -0.1000, x = 2.1800, y = 0.3825
a = -0.1000, x = 2.3900, y = 0.3630
a = -0.1000, x = 2.6000, y = 0.3434
a = -0.1000, x = 2.8100, y = 0.3236
a = -0.1000, x = 3.0200, y = 0.3037
a = -0.2000, x = 0.5000, y = 0.7326
a = -0.2000, x = 0.7100, y = 0.7034
a = -0.2000, x = 0.9200, y = 0.6729
a = -0.2000, x = 1.1300, y = 0.6412
a = -0.2000, x = 1.3400, y = 0.6085
a = -0.2000, x = 1.5500, y = 0.3323
a = -0.2000, x = 1.7600, y = 0.2924
a = -0.2000, x = 1.9700, y = 0.2519
a = -0.2000, x = 2.1800, y = 0.2111
a = -0.2000, x = 2.3900, y = 0.1699
a = -0.2000, x = 2.6000, y = 0.1283
a = -0.2000, x = 2.8100, y = 0.0866
a = -0.2000, x = 3.0200, y = 0.0447
a = -0.3000, x = 0.5000, y = 0.6709
a = -0.3000, x = 0.7100, y = 0.6229
a = -0.3000, x = 0.9200, y = 0.5724
a = -0.3000, x = 1.1300, y = 0.5197
a = -0.3000, x = 1.3400, y = 0.4649
a = -0.3000, x = 1.5500, y = 0.2186
a = -0.3000, x = 1.7600, y = 0.1567
a = -0.3000, x = 1.9700, y = 0.0942
a = -0.3000, x = 2.1800, y = 0.0313
a = -0.3000, x = 2.3900, y = -0.0317
a = -0.3000, x = 2.6000, y = -0.0945
a = -0.3000, x = 2.8100, y = -0.1570
a = -0.3000, x = 3.0200, y = -0.2189
a = -0.4000, x = 0.5000, y = 0.6054
a = -0.4000, x = 0.7100, y = 0.5365

a = -0.4000, x = 0.9200, y = 0.4638
a = -0.4000, x = 1.1300, y = 0.3878
a = -0.4000, x = 1.3400, y = 0.3091
a = -0.4000, x = 1.5500, y = 0.1003
a = -0.4000, x = 1.7600, y = 0.0165
a = -0.4000, x = 1.9700, y = -0.0674
a = -0.4000, x = 2.1800, y = -0.1509
a = -0.4000, x = 2.3900, y = -0.2333
a = -0.4000, x = 2.6000, y = -0.3141
a = -0.4000, x = 2.8100, y = -0.3926
a = -0.4000, x = 3.0200, y = -0.4684
a = -0.5000, x = 0.5000, y = 0.5367
a = -0.5000, x = 0.7100, y = 0.4453
a = -0.5000, x = 0.9200, y = 0.3490
a = -0.5000, x = 1.1300, y = 0.2489
a = -0.5000, x = 1.3400, y = 0.1460
a = -0.5000, x = 1.5500, y = -0.0207
a = -0.5000, x = 1.7600, y = -0.1254
a = -0.5000, x = 1.9700, y = -0.2287
a = -0.5000, x = 2.1800, y = -0.3294
a = -0.5000, x = 2.3900, y = -0.4266
a = -0.5000, x = 2.6000, y = -0.5190
a = -0.5000, x = 2.8100, y = -0.6057
a = -0.5000, x = 3.0200, y = -0.6858
a = -0.6000, x = 0.5000, y = 0.4654
a = -0.6000, x = 0.7100, y = 0.3505
a = -0.6000, x = 0.9200, y = 0.2300
a = -0.6000, x = 1.1300, y = 0.1059
a = -0.6000, x = 1.3400, y = 0.0199
a = -0.6000, x = 1.5500, y = -0.1428
a = -0.6000, x = 1.7600, y = -0.2660
a = -0.6000, x = 1.9700, y = -0.3850
a = -0.6000, x = 2.1800, y = -0.4980
a = -0.6000, x = 2.3900, y = -0.6030
a = -0.6000, x = 2.6000, y = -0.6985
a = -0.6000, x = 2.8100, y = -0.7829
a = -0.6000, x = 3.0200, y = -0.8548
a = -0.7000, x = 0.5000, y = 0.3921
a = -0.7000, x = 0.7100, y = 0.2531
a = -0.7000, x = 0.9200, y = 0.1087

a = -0.9000, x = 1.1300, y = 0.3172
a = -0.9000, x = 1.3400, y = 0.4897
a = -0.9000, x = 1.5500, y = -0.4955
a = -0.9000, x = 1.7600, y = -0.6499
a = -0.9000, x = 1.9700, y = -0.7811
a = -0.9000, x = 2.1800, y = -0.8845
a = -0.9000, x = 2.3900, y = -0.9564
a = -0.9000, x = 2.6000, y = -0.9942
a = -0.9000, x = 2.8100, y = -0.9967
a = -0.9000, x = 3.0200, y = -0.9636
a = -1.0000, x = 0.5000, y = 0.1659
a = -1.0000, x = 0.7100, y = 0.0433
a = -1.0000, x = 0.9200, y = 0.2506
a = -1.0000, x = 1.1300, y = 0.4469
a = -1.0000, x = 1.3400, y = 0.6236
a = -1.0000, x = 1.5500, y = -0.6019
a = -1.0000, x = 1.7600, y = -0.7551
a = -1.0000, x = 1.9700, y = -0.8752
a = -1.0000, x = 2.1800, y = -0.9568
a = -1.0000, x = 2.3900, y = -0.9964
a = -1.0000, x = 2.6000, y = -0.9922
a = -1.0000, x = 2.8100, y = -0.9444
a = -1.0000, x = 3.0200, y = -0.8551
a = -1.1000, x = 0.5000, y = 0.0901
a = -1.1000, x = 0.7100, y = 0.1404
a = -1.1000, x = 0.9200, y = 0.3633
a = -1.1000, x = 1.1300, y = 0.5670
a = -1.1000, x = 1.3400, y = 0.7405
a = -1.1000, x = 1.5500, y = -0.6993
a = -1.1000, x = 1.7600, y = -0.8444
a = -1.1000, x = 1.9700, y = -0.9446
a = -1.1000, x = 2.1800, y = -0.9947
a = -1.1000, x = 2.3900, y = -0.9919
a = -1.1000, x = 2.6000, y = -0.9364
a = -1.1000, x = 2.8100, y = -0.8312
a = -1.1000, x = 3.0200, y = -0.6818
a = -1.2000, x = 0.5000, y = 0.0147
a = -1.2000, x = 0.7100, y = 0.2350
a = -1.2000, x = 0.9200, y = 0.4700

```
a = -1.2000, x = 1.1300, y = 0.6752
a = -1.2000, x = 1.3400, y = 0.8378
a = -1.2000, x = 1.5500, y = -0.7858
a = -1.2000, x = 1.7600, y = -0.9152
a = -1.2000, x = 1.9700, y = -0.9868
a = -1.2000, x = 2.1800, y = -0.9960
a = -1.2000, x = 2.3900, y = -0.9424
a = -1.2000, x = 2.6000, y = -0.8292
a = -1.2000, x = 2.8100, y = -0.6636
a = -1.2000, x = 3.0200, y = -0.4561
a = -1.3000, x = 0.5000, y = 0.0597
a = -1.3000, x = 0.7100, y = 0.3266
a = -1.3000, x = 0.9200, y = 0.5693
a = -1.3000, x = 1.1300, y = 0.7699
a = -1.3000, x = 1.3400, y = 0.9135
a = -1.3000, x = 1.5500, y = -0.8596
a = -1.3000, x = 1.7600, y = -0.9655
a = -1.3000, x = 1.9700, y = -1.0000
a = -1.3000, x = 2.1800, y = -0.9603
a = -1.3000, x = 2.3900, y = -0.8495
a = -1.3000, x = 2.6000, y = -0.6758
a = -1.3000, x = 2.8100, y = -0.4521
a = -1.3000, x = 3.0200, y = -0.1948
```

Рисунок 3.4. Приклад виконання програми № 2

Фактично затрачений час 50 хвилин.

Завдання №3

```

1  #include <iostream>
2
3  int main() {
4      double num1, num2, result;
5      char operation;
6
7      // Введення двох чисел
8      std::cout << "Введіть перше число: ";
9      std::cin >> num1;
10     std::cout << "Введіть друге число: ";
11     std::cin >> num2;
12
13     // Вибір операції
14     std::cout << "Виберіть операцію (+, -, *, /): ";
15     std::cin >> operation;
16
17     // Виконання операції в залежності від вибору користувача
18     if (operation == '+') {
19         result = num1 + num2;
20         std::cout << "Результат додавання: " << result << std::endl;
21     } else if (operation == '-') {
22         result = num1 - num2;
23         std::cout << "Результат віднімання: " << result << std::endl;
24     } else if (operation == '*') {
25         result = num1 * num2;
26         std::cout << "Результат множення: " << result << std::endl;
27     } else if (operation == '/') {
28         if (num2 != 0) {
29             result = num1 / num2;
30             std::cout << "Результат ділення: " << result << std::endl;
31         } else {
32             std::cout << "Ділення на нуль неможливе!" << std::endl;
33         }
34     } else {
35         std::cout << "Невідома операція!" << std::endl;
36     }
37
38     return 0;
39 }

```

Рисунок 3.5. Код до програми № 3

```

Введіть перше число: 5
Введіть друге число: 7
Виберіть операцію (+, -, *, /): +
Результат додавання: 12

```

Рисунок 3.6. Приклад виконання програми № 3

Фактично затрачений час 25 хвилин.

Завдання №4

```

1  #include <iostream>
2  #include <iomanip>
3
4  int main() {
5      // Массив перших п'яти цілих позитивних непарних чисел
6      int oddNumbers[] = {1, 3, 5, 7, 9};
7
8      std::cout << std::setw(10) << "Number" << std::setw(10) << "Square" << std::endl;
9      std::cout << "======" << std::endl;
10
11     for(int i = 0; i < 5; ++i) {
12         int number = oddNumbers[i];
13         int square = number * number;
14         std::cout << std::setw(10) << number << std::setw(10) << square << std::endl;
15     }
16
17     return 0;
18 }

```

Рисунок 3.7. Код до програми № 4

Number	Square
=====	
1	1
3	9
5	25
7	49
9	81

Рисунок 3.8. Приклад виконання програми № 4

Фактично затрачений час 20 хвилин.

Завдання №5

```

1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int main() {
6      // Зчитування кількості пачок печива
7      int n;
8      cin >> n;
9
10     // Зчитування кількості штук печива в кожній пачці
11     vector<long long> a(n);
12     for (int i = 0; i < n; ++i) {
13         cin >> a[i];
14     }
15
16     // Обчислення максимальної кількості печива, яку Марічка може з'їсти
17     long long maxCookies = 0;
18     for (int i = 0; i < n; ++i) {
19         maxCookies += a[i] - 1; // Марічка може з'їсти все, крім 1 штуки в пачці
20     }
21
22     // Вивід результату
23     cout << maxCookies << endl;
24
25     return 0;
26 }

```

Рисунок 3.9. Код до програми № 5

```

5
5 7 2 10 5
24

```

Рисунок 3.10. Приклад виконання програми № 5

декілька секунд тому	C++ 23	Зараховано	0.038	1.039	Перегляд
----------------------	--------	------------	-------	-------	--------------------------

Рисунок 3.11. Статус задачі на Algotester

Фактично затрачений час 25 хвилин.

Завдання №6


```

1  #include <iostream>
2  #include <vector>
3  #include <algorithm>
4
5  using namespace std;
6
7  int main() {
8      int n;
9      cin >> n;
10
11     vector<int> a(n), b(n), c(n);
12
13     // Зчитуємо бажані кількості для кожної страви
14     for (int i = 0; i < n; ++i) {
15         cin >> a[i] >> b[i] >> c[i];
16     }
17
18     // Функція для знаходження медіани
19     auto get_median = [](vector<int>& v) {
20         sort(v.begin(), v.end());
21         return v[v.size() / 2];
22     };
23
24     // Знаходимо медіани для кожної страви
25     int A = get_median(a);
26     int B = get_median(b);
27     int C = get_median(c);
28
29     // Обчислюємо сумарний ступінь недовіри
30     long long total_distrust = 0;
31     for (int i = 0; i < n; ++i) {
32         total_distrust += abs(A - a[i]) + abs(B - b[i]) + abs(C - c[i]);
33     }
34
35     cout << total_distrust << endl;
36
37     return 0;
38 }
39

```

Рисунок 3.12. Код до програми № 6

```

2
10 50 100
70 10 50
150

```

Рисунок 3.13. Приклад виконання програми №6

декілька секунд тому	C++ 23	Зараховано	0.103	2.543	Перегляд
----------------------	--------	------------	-------	-------	----------

Рисунок 3.14. Статус задачі на Algotester

Фактично затрачений час 30 хвилин.

Завдання №7

```
1  #include <iostream>
2  #include <cmath>
3  #include <iomanip>
4  using namespace std;
5
6  int main() {
7      int n, r;
8      cin >> n >> r;
9
10     // Обчислюємо радіус найменшого ярусу
11     double r_min = sqrt(static_cast<double>(r * r) / n);
12
13     // Виводимо результат з точністю 4 знаки після коми
14     cout << fixed << setprecision(4) << r_min << endl;
15
16     return 0;
17 }
18
```

Рисунок 3.15. Код до програми № 7

```
4 7
3.5000
```

Рисунок 3.16. Приклад виконання програми №7

декілька секунд тому	C++ 23	Зараховано	0.003	1.203	Перегляд
----------------------	--------	------------	-------	-------	--------------------------

Рисунок 3.17. Статус завдання на Algotester

Фактично затрачений час 20 хвилин.

Завдання №8

```

1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  int main() {
6      int n, k;
7      cin >> n >> k;
8
9      vector<int> a(n);
10     for (int i = 0; i < n; ++i) {
11         cin >> a[i];
12     }
13
14     int max_length = 0; // Максимальна довжина серії загострених зубів
15     int current_length = 0; // Поточна довжина серії
16
17     for (int i = 0; i < n; ++i) {
18         if (a[i] >= k) {
19             current_length++; // Продовжуємо серію
20             max_length = max(max_length, current_length);
21         } else {
22             current_length = 0; // Серія переривається
23         }
24     }
25
26     cout << max_length << endl;
27     return 0;
28 }
29

```

Рисунок 3.18. Код до програми № 8

```

7 4
7 1 4 7 6 3 4
3

```

Рисунок 3.19. Приклад виконання програми №8

декілька секунд тому	C++ 23	Зараховано	0.047	1.480	Перегляд
----------------------	--------	------------	-------	-------	----------

Рисунок 3.20. Статус задачі на Algotester

Фактично затрачений час 35 хвилин.

Посилання на пул реквест: [Saga 1 - Ostap Tsiapa by Ostap2007ter · Pull Request #556 · artificial-intelligence-department/ai_programming_playground_2024](#)