

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

Виконала:

Студентка групи ШІ-11

Потапова Світлана Сергіївна

Львів 2024

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

Виконання роботи:

1. Опрацювання завдань та вимог до середовища:

Завдання №1. VNS Practice Work Task 1

Варіант 11. $p = \frac{a}{b}$, де $a = (1 - y) \frac{(x + y)^2 / (x + 4)^3}{e^{-(x-2)} + (x^3 + 4)}$;
 $b = \frac{1 + \cos(y - 2)}{x^{4/2} + \sin^2(y - 2)}$; $x=1,25$; $y=0,93$.

Завдання №2. VNS Practice Work Task 2

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

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

Завдання №3. VNS Practice Work Task 3

Варіант 16. Обчислення величини доходу по внеску. Процентна ставка (% річних) і час зберігання (днів) задаються під час роботи програми. Нижче приведений вид екрану під час виконання програми, що рекомендується (дані, введені користувачем, виділені напівжирним шрифтом).

Обчислення величини доходу по внеску.

Введіть початкові дані:

Величина внеску (грн.) > **2500**

Термін внеску (днів) > 30

Процентна ставка (річних в %) > **20**

Дохід: 41.10 грн.

Сума, після закінчення терміну внеску: 2541.10 грн.

Обчислення об'єму порожнистого циліндра $V = \pi \cdot h \cdot (r_1^2 - r_2^2)$,
де r_1 – радіус циліндра, r_2 – радіус отвору, h – висота циліндра.

Завдання №4. VNS Practice Work Task 4

Варіант 10. Написати програму, яка обчислює факторіал введеного з клавіатури числа.

Завдання №5. Algotester 0101 Дракон і кросворд

Наш дракон переконаний, що кросворд — це прямокутна таблиця $n \times m$, в кожній клітинці якої записано одну букву латинського алфавіту. Щоб

розв'язати кросворд, Віталій викреслює всі букви, які повторяються в стовпцях або в рядках. Тобто буква буде викреслена тоді і тільки тоді, коли у тому ж стовпці або рядку є така ж буква. Причому всі такі букви викреслюються одночасно. Коли викреслено всі такі букви, ті, що залишились виписують в один рядок. Спочатку виписують ті букви, які розташовані вище. Якщо букви записані в одному рядку, їх виписують в порядку зліва направо.

Вхідні дані

У першому рядку задано два натуральних числа n і m — кількість рядків і стовпців відповідно. У наступних n рядках задано по одному рядку довжиною m . Кожен рядок складається з маленьких латинських букв.

Вихідні дані

У єдиному рядку виведіть зашифроване слово. Гарантується, що відповідь складається хоча б з одної букви.

Завдання №6. Algotester 0081 Допоможе чи заб'є?

До Тойлет-мена дуже часто звертаються по допомогу різні люди. 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 k 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.

Вхідні дані

У першому рядку задано одне ціле число k — мінімальна кількість незалежних входжень рядка TOILET. Другий рядок містить рядок s — повідомлення, надіслане Тойлет-мену. Рядок містить лише великі латинські символи.

Вихідні дані

У єдиному рядку виведіть YES або NO — відповідь на задачу.

Завдання №7. Algotester 0021 Депутатські гроші

Часто-густо громадяни намагаються з'ясувати, наскільки багатими є депутати. Дехто вірить, що матеріальні статки окремих депутатів є необмеженими. Тож уявіть собі депутата, у якого є необмежена кількість купюр усіх номіналів (1, 2, 5, 10, 20, 50, 100, 200 та 500 гривень). Він хоче придбати подарунок для своєї тещі, що коштує n гривень. Незважаючи на те, що наш герой-олігарх є нескінченно багатим, він також є нескінченно скупим та педантичним. Саме тому він хоче оплатити покупку готівкою без решти. Яка мінімальна кількість купюр йому для цього знадобиться?

Вхідні дані

У єдиному рядку задано одне натуральне число n — вартість подарунку.

Вихідні дані

У єдиному рядку виведіть одне ціле число — мінімальну кількість купюр, що необхідна для покупки подарунка.

Завдання №8. Algotester 0183 Вогняне дихання

Одного разу Сеньйор розповів Дракону, що минулого вечора він k разів бачив над лісом дим. Дракон пригадав, що вчора він гуляв лісом та час від часу заради розваги дмухав вогнем, як це інколи роблять всі поважні дракони. Всього Дракон дмухав вогнем у n точках. Сеньйору раптом стало цікаво — наскільки далеко він може побачити драконів дим? Нехай відстань, на яку видно дим — r . Тоді кожного разу, коли Дракон дмухає вогнем, дим можуть побачити лише ті, хто знаходиться на відстані не більшій ніж r від дракона. Знаючи координати хатинки Сеньйора, а також усіх місць, де дмухав вогнем Дракон, скажіть, яка найменша можлива відстань r , при якій Сеньйор міг бачити дим хоча б k разів.

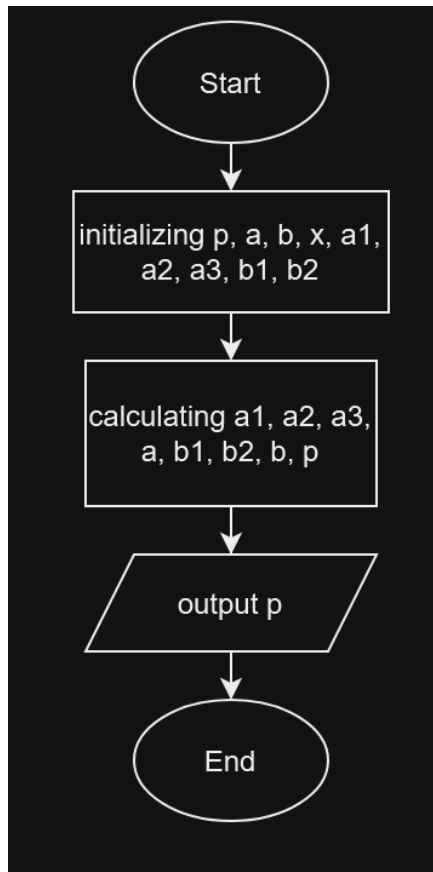
Вхідні дані

У першому рядку вхідних даних задані два цілих числа X та Y — координати хатинки Сеньйора. У наступному рядку задані два цілих числа n та k . У наступних n рядках задані пари цілих чисел x_i та y_i — координати точок, у яких Дракон дмухав вогнем.

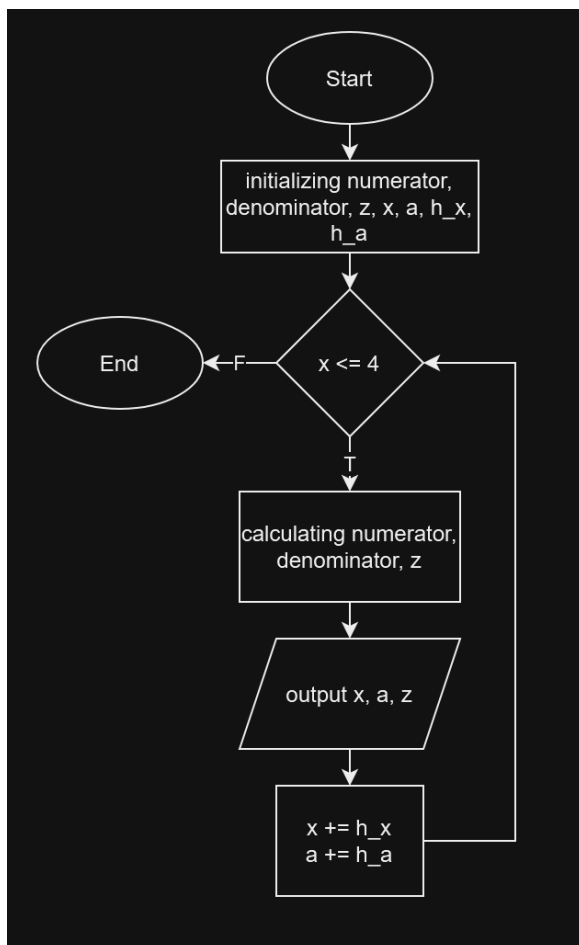
Вихідні дані

У єдиному рядку виведіть найменше можливе значення r . Відповідь буде зарахованою, якщо її відносна або абсолютна похибка не перевищуватиме 10^{-4} .

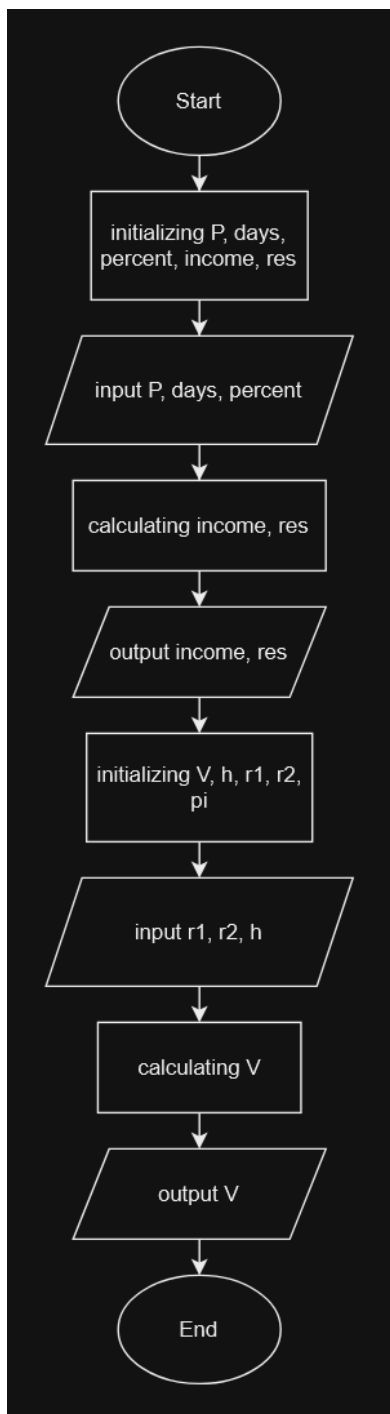
2. Дизайн виконання завдань**Завдання №1. VNS Practice Work Task 1**



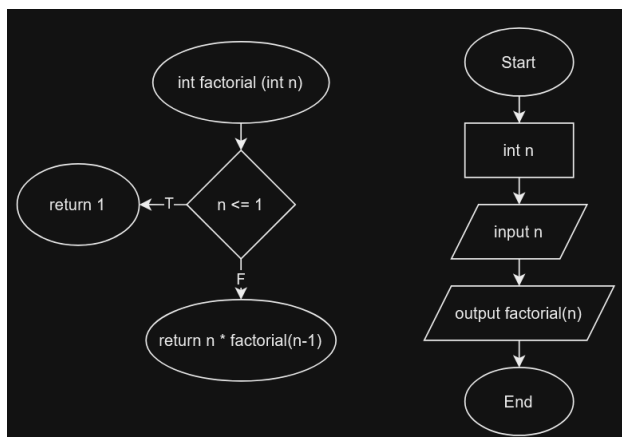
Завдання №2. VNS Practice Work Task 2



Завдання №3. VNS Practice Work Task 3



Завдання №4. VNS Practice Work Task 4

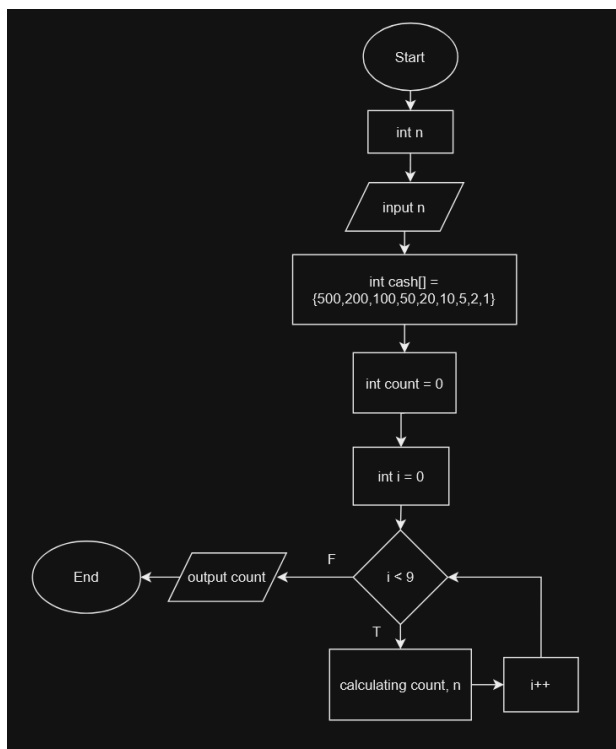


[illegible]

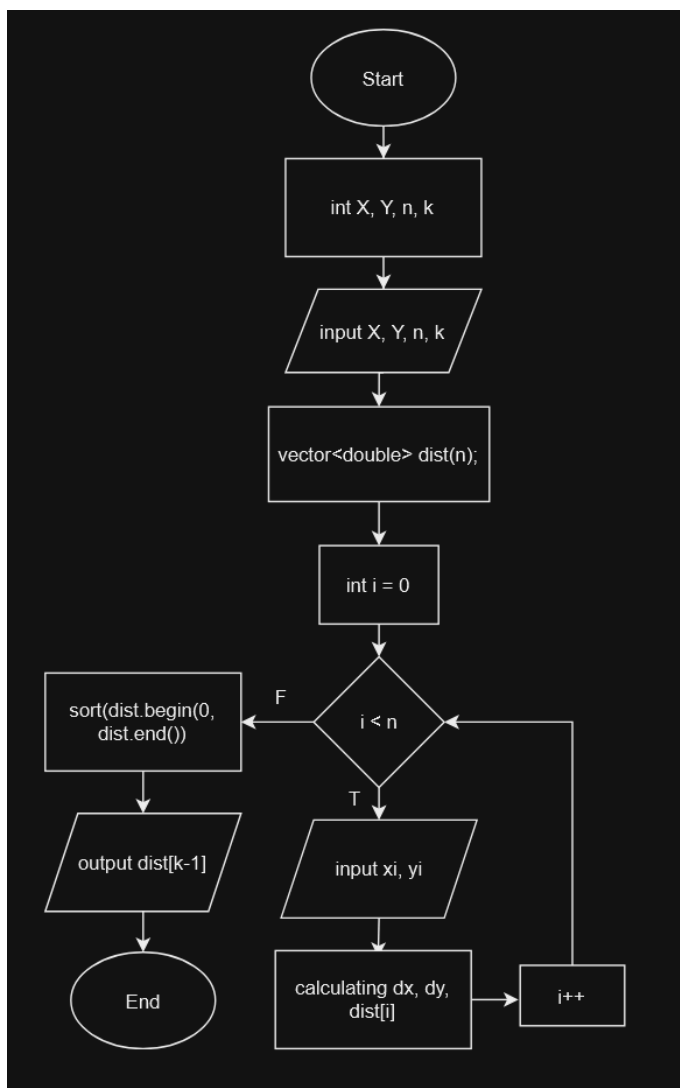
```

graph TD
    Start([Start]) --> Init[initializing k, amount, s, target]
    Init --> InputK[/input k/]
    InputK --> InputC[/input c/]
    InputC --> Find1[int found = s.find(target)]
    Find1 --> Decision1{found != string::npos}
    Decision1 -- T --> AmountInc[amount++]
    AmountInc --> Erase[s.erase(0, found+target.length())]
    Erase --> Find2[int found = s.find(target)]
    Find2 --> Decision1
    Decision1 -- F --> Decision2{amount >= k}
    Decision2 -- T --> OutputYes[/output "YES"/]
    OutputYes --> End([End])
    Decision2 -- F --> OutputNo[/output "NO"/]
    OutputNo --> End
  
```

Завдання №7. Algotester 0021 Депутатські гроші



Завдання №8. Algotester 0183 Вогняне дихання



3. Код програм

Завдання №1. VNS Practice Work Task 1

```
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  int main(){
6      double p, a, b, x = 1.25, y = 0.93, e = 2.72;
7      double a1, a2, a3, b1, b2;
8      a1 = ((x+y)*(x+y)) / ((x+4)*(x+4)*(x+4));
9      a2 = pow(e, -(x-2)) + (x*x*x + 4);
10     a3 = 1 - y;
11     a = (a3*a1) / a2;
12
13     b1 = 1 + cos(y-2);
14     b2 = x*x + pow(sin(y-2), 2);
15     b = b1 / b2;
16
17     p = a / b;
18
19     cout << p;
20
21     return 0;
22 }
```

Завдання №2. VNS Practice Work Task 2

```
1  #include <iostream>
2  #include <cmath>
3  using namespace std;
4
5  int main(){
6      double numerator, denominator, z, x = 1, a = 0.6, h_x = 0.5, h_a = 0.2;
7
8      while(x<=4){
9          numerator = pow((x-1.5), 1/2) + pow(x,a);
10         denominator = pow((x-2), 1/3);
11         z = numerator / denominator;
12         cout << "x = " << x << "\t\t a = " << a << "\t\t z = " << z << endl;
13         x += h_x;
14         a += h_a;
15     }
16
17     return 0;
18
19 }
```

Завдання №3. VNS Practice Work Task 3

```

1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int P, days;
6      double percent, income, res;
7
8      cout << "Обчислення величини доходу по внеску" << endl;
9      cout << "Введіть початкові дані:" << endl << "Величина внеску (грн): ";
10     cin >> P;
11
12     cout << "Термін внеску (днів): ";
13     cin >> days;
14
15     cout << "Процентна ставка (річних в %): ";
16     cin >> percent;
17
18     income = P * percent/100 * days/365;
19     res = P + income;
20
21     cout << "Дохід: " << income << endl;
22     cout << "Сума, після закінчення терміну внеску: " << res << endl;
23
24     cout << endl << "Обчислення об'єму порожнистого циліндра" << endl;
25
26     float V, h, r1, r2;
27     const float pi = 3.14;
28
29     cout << "Введіть радіус циліндра: ";
30     cin >> r1;
31     cout << "Введіть радіус отвору: ";
32     cin >> r2;
33     cout << "Введіть висоту циліндра: ";

```

```

34     cout << "Введіть висоту циліндра: ";
35     cin >> h;
36
37     V = pi*h*(r1*r1 - r2*r2);
38
39     cout << "Об'єм порожнистого циліндра: " << V;
40
41     return 0;
42 }

```

Завдання №4. VNS Practice Work Task 4

```

1  #include <iostream>
2  using namespace std;
3
4  int factorial(int n) {
5      if (n <= 1)
6          return 1;
7      else
8          return n * factorial(n - 1);
9  }
10
11  int main(){
12      int n;
13      cout << "Введіть число: ";
14      cin >> n;
15      cout << "Факторіал числа " << n << ": " << factorial(n);
16
17      return 0;
18  }
19

```

Завдання №5. Algotester 0101 Дракон і кросворд

```
1  √ #include <iostream>
2  #include <vector>
3  #include <algorithm>
4  using namespace std;
5
6  √ int main(){
7      int n, m;
8      cin >> n >> m;
9
10     vector<vector<char>> matrix(n, vector<char>(m));
11     vector<char> answer;
12
13     for(int i=0; i<n; i++){
14         for(int j=0; j<m; j++){
15             cin >> matrix[i][j];
16         }
17     }
18
19     for (int i = 0; i < n; i++) {
20         for (int j = 0; j < m; j++) {
21             char target = matrix[i][j];
22             bool foundInRow = false, foundInCol = false;
23
24             for (int col = 0; col < m; col++) {
25                 if (col != j && matrix[i][col] == target) {
26                     foundInRow = true;
27                     break;
28                 }
29             }
30
31             for (int row = 0; row < n; row++) {
32                 if (row != i && matrix[row][j] == target) {
33                     foundInCol = true;
```

```
34             break;
35         }
36     }
37
38     if (!foundInRow && !foundInCol) {
39         answer.push_back(matrix[i][j]);
40     }
41 }
42 }
43
44
45 for (int i = 0; i < answer.size(); i++) {
46     cout << answer[i];
47 }
48
49 return 0;
50 }
```

Завдання №6. Algotester 0081 Допоможе чи заб'є?

```

1  #include <iostream>
2  #include <string>
3  #include <algorithm>
4
5  using namespace std;
6
7  int main() {
8      int k;
9      int amount = 0;
10     string s;
11     string target = "TOILET";
12
13     cin >> k;
14     cin.ignore();
15
16     getline(cin, s);
17
18     int found = s.find(target);
19     while (found != string::npos) {
20         amount++;
21         s.erase(0, found + target.length());
22         found = s.find(target);
23     }
24
25     if(amount>=k){
26         cout << "YES";
27     }
28     else{
29         cout << "NO";
30     }
31
32     return 0;
33 }

```

Завдання №7. Algotester 0021 Депутатські гроші

```

1  #include <iostream>
2  using namespace std;
3
4  int main(){
5      int n;
6      cin >> n;
7      int cash[]={500,200,100,50,20,10,5,2,1};
8      int count=0;
9      for(int i=0; i<9; i++){
10         count += n / cash[i];
11         n %= cash[i];
12     }
13     cout << count;
14     return 0;
15 }

```

Завдання №8. Algotester 0183 Вогняне дихання

```

1  #include <iostream>
2  #include <vector>
3  #include <cmath>
4  #include <algorithm>
5  #include <iomanip>
6  using namespace std;
7
8  int main()
9  {
10     int X, Y, n, k, xi, yi;
11     cin >> X >> Y >> n >> k;
12
13     vector<double> dist(n);
14
15     for (int i = 0; i < n; i++) {
16         cin >> xi >> yi;
17         double dx = xi - X;
18         double dy = yi - Y;
19         dist[i] = sqrt(dx * dx + dy * dy);
20     }
21
22     sort(dist.begin(), dist.end());
23     cout << fixed << setprecision(9) << dist[k - 1];
24
25     return 0;
26 }

```

4. Результати виконання завдань та фактично затрачений час

Завдання №1. VNS Practice Work Task 1

0.000448772

Планований час: 20 хв, фактично: 20 хв

Завдання №2. VNS Practice Work Task 2

x = 1	a = 0.6	z = 2
x = 1.5	a = 0.8	z = 2.38316
x = 2	a = 1	z = 3
x = 2.5	a = 1.2	z = 4.00281
x = 3	a = 1.4	z = 5.65554
x = 3.5	a = 1.6	z = 8.42179
x = 4	a = 1.8	z = 13.1257

Планований час: 20 хв, фактично: 20 хв

Завдання №3. VNS Practice Work Task 3

Обчислення величини доходу по внеску
Введіть початкові дані:
Величина внеску (грн): 10000
Термін внеску (днів): 120
Процентна ставка (річних в %): 10
Дохід: 328.767
Сума, після закінчення терміну внеску: 10328.8

Обчислення об'єму порожнистого циліндра
Введіть радіус циліндра: 8
Введіть радіус отвору: 3
Введіть висоту циліндра: 5
Об'єм порожнистого циліндра: 863.5

Планований час: 30 хв, фактично: 30 хв

Завдання №4. VNS Practice Work Task 4

```
Введіть число: 5
Факторіал числа 5: 120
```

Планований час: 15 хв, фактично: 10 хв

Завдання №5. Algotester 0101 Дракон і кросворд

```
4 4
defr
aege
tofn
tdde
dragon
```

Планований час: 30 хв, фактично: 30 хв

Завдання №6. Algotester 0081 Допоможе чи заб'є?

```
2
HELPMETOILETPLEASETOILET
YES
```

Планований час: 30 хв, фактично: 40 хв

Завдання №7. Algotester 0021 Депутатські гроші

```
74
4
```

Планований час: 15 хв, фактично: 15 хв

Завдання №8. Algotester 0183 Вогняне дихання

```
5 6
4 2
3 4
8 11
6 5
17 25
2.828427125
```

Планований час: 40 хв, фактично: 40 хв

декілька секунд тому	0101 - Дракон і кросворд	C++ 23	Зараховано	0.003	1.051	1906935
декілька секунд тому	0081 - Допоможе чи заб'є?	C++ 23	Зараховано	0.021	1.191	1906933
хвилину тому	0021 - Депутатські гроші	C++ 23	Зараховано	0.002	1.176	1906932
16 хвилин тому	0183 - Вогняне дихання	C++ 23	Зараховано	0.002	2.008	1906923

Висновок: Під час виконання розрахункової роботи я закріпила свої знання у мові C/C++, набуті протягом виконання попередніх завдань.