

ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ № 6

За курсом «Програмування»

Студента групи ПА-23-1

Мороза Миколи Олександровича

Кафедра комп'ютерних технологій, ДНУ

2023/2024

1.Постановка задачі

Сьогодні ми будемо робити дві програми, перша програма буде рахувати задану функцію у разі потрапляння аргументу в n -ий інтервал, якщо значення не входить в ОДЗ функції то програма виводить «не потрапляє в ОДЗ», далі у другому завданні програма буде рахувати суму ряду Тейлора заданої функції, але точність до якої буде рахувати вказує користувач

2.Опис розв'язку

По-перше ми підключаємо 4 бібліотеки (iostream, Windows.h, math.h, iomanip), у першій програмі ми оголошуємо змінну в якій прописуємо нашу задану функцію за допомогою математичних функцій які входять в бібліотеку math.h, потім ми просимо користувача ввести початок та кінець інтервалу і шаг, далі ми передаємо ці значення у функцію і потім програма виводить значення цієї функції. У другому завданні програма, за допомогою циклів ми знаходимо значення ряду Тейлора перший 5 членів, та потім і усього ряду, після цього ми знаходимо суму ряду Тейлора за допомогою математичних функцій, після цього всього ми виводимо всі знайдені значення.

3.Вихідний текст програми розв'язку задачі

```

#include <math.h>
#include <Windows.h>
#include <iostream>
#include <iomanip>
using namespace std;

double Function(double x) {
    double Sqrt_Sin, Sqrt_of_Exp, Drib;
    Sqrt_Sin = sqrt(pow(sin(x / 2), 3));
    Sqrt_of_Exp = pow(exp(1.3 * x) + exp(-1.3 * x), 1.0 / 3);
    Drib = 1.0 / abs(x + 5.0 / 2);
    return (Sqrt_Sin + Sqrt_of_Exp) * Drib;
}

void first_task(double x_start, double x_end, double x_shag) {
    double x, y;
    cout << "|-----|-----|" << endl;
    cout << "  x  |      y      |" << endl;
    cout << "|-----|-----|" << endl;

    for (x = x_start; x <= x_end; x += x_shag) {
        if (x > 5 && x < 10 && x != -5.0 / 2) {
            y = Function(x);

        }
        else if (x > -5 && x <= 2) {
            y = x;

        }
        else {
            y = 1.0 / pow(x, 2);

        }

        cout << " | " << setw(8);
        if (x < 0.0000001 && x > -0.0000001) cout << "0";
        else cout << x;
        cout << " | " << setw(19);
        if (x >= 6 && x <= 10) cout << "Не входить в ОДЗ";
        else if (x < 0.0000001 && x > -0.0000001) cout << "0";
        else cout << y;
        cout << " |" << endl;

    }

    cout << "|-----|-----|" << endl;
}

void second_task(double Accuracy, int x_2) {
    double current_member = x_2;
    double sum = 1, sum_while = 1, sum_do_while = 1;
    double i, j = 0;
    for (i = 0; abs(current_member) > Accuracy; j++, sum += current_member, i++)
    {
        current_member *= -(x_2 * x_2) / (i + 1);
        if (j == 7) cout << "Сума перших 5 членів ряду Тейлора =" << setprecision(16) << sum << endl;
    }

    j = 0;
    i = 0;
    current_member = x_2;
}

```

```

while (abs(current_member) > Accuracy)
{
    current_member *= -(x_2 * x_2) / (i + 1);

    j++;
    sum_while += current_member;
    i++;
}

j = 0;
i = 0;
current_member = x_2;

do
{
    current_member *= -(x_2 * x_2) / (i + 1);

    sum_do_while += current_member;
    j++;
    i++;
} while (abs(current_member) > Accuracy);

cout << setprecision(18) << "Сума ряду Тейлора = " << sum << endl;
cout << setprecision(18) << "Сума ряду Тейлора = " << sum_while << endl;
cout << setprecision(18) << "Сума ряду Тейлора = " << sum_do_while << endl;
cout << "\t e^-x^2 = " << setprecision(18) << exp(-(x_2 * x_2)) << endl;
cout << "Кількість ітерацій = " << j << endl;
}

void Checking_number_for_first_task(double& x_start, double& x_end, double& x_shag) {
    while (((x_start || x_end || x_shag) == !cin) || x_end < x_start || x_shag <= 0) {
        if ((x_start || x_end || x_shag) == !cin) {
            cout << "Ви ввели не число" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin.clear();
            cin.ignore(32767, '\n');
            cin >> x_start >> x_end >> x_shag;
        }
        else if (x_end < x_start && x_shag <= 0) {
            cout << "Кінець інтервалу повинен бути більше за початок, та шаг повинен бути додатнім"
<< endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin >> x_start >> x_end >> x_shag;
        }
        else if (x_end < x_start) {
            cout << "Кінець інтервалу повинен бути більше за початок" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin >> x_start >> x_end >> x_shag;
        }
        else if (x_shag <= 0) {
            cout << "Шаг повинен бути додатнім (він не може бути нулем або від'ємним)" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin >> x_start >> x_end >> x_shag;
        }
    }
}
}

```

```

void Checking_number_for_second_task(double& Accuracy, int& x_2) {
    cin.clear();
    cin.ignore(32767, '\n');
    while ((Accuracy == !cin) || (x_2 == !cin)) {
        if ((Accuracy == !cin) || (x_2 == !cin)) {
            cout << "\nВи ввели не число\n";
            cout << "Введіть точність розрахунків та аргумент (x)\nУ послідовності перша - це точність,
далі друга - аргумент: ";
            cin >> Accuracy >> x_2;
        }
        cin.clear();
        cin.ignore(32767, '\n');
    }

    cout << endl;
}

int main(int argc, char* argv[])
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    cout << "\t\tЛабораторна робота № 6" << endl;

    cout << "\t\tЗавдання № 1" << endl << endl;
    double x_start, x_end, x_shag;
    int choice_to_try_again;
    if (argc == 4)
    {
        x_start = atof(argv[1]);
        x_end = atof(argv[2]);
        x_shag = atof(argv[3]);
        if ((x_start || x_end || x_shag) == !cin) {
            cout << "Ви ввели не число" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin.clear();
            cin.ignore(32767, '\n');
            cin >> x_start >> x_end >> x_shag;
        }
    }
    else
    {
        cout << "Ви не ввели всі значення, будь ласка введіть значення у цій послідовності" << endl;
        cout << "Введіть початок інтервалу, кінець інтервалу та шаг: ";
        cin >> x_start >> x_end >> x_shag;
    }

    Checking_number_for_first_task(x_start, x_end, x_shag);

    first_task(x_start, x_end, x_shag);

    cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:";
    cin >> choice_to_try_again;
    while (choice_to_try_again == 1) {
        switch (choice_to_try_again) {
            case 1:
                cout << "Введіть початок інтервалу, кінець інтервалу та шаг: ";

```

```

        cin >> x_start >> x_end >> x_shag;
        Checking_number_for_first_task(x_start, x_end, x_shag);
        cout << endl;
        first_task(x_start, x_end, x_shag);
        cout << endl;
        cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 -
закінчити:";

        cin >> choice_to_try_again;
        break;

    default:
        cout << "Програма переходить до іншого завдання ";
        break;
    }
}

cout << "\t\tЗавдання № 2" << endl << endl;
double Accuracy;
int x_2;
int choice_to_try_again_2;

cout << "Калькулятор  $e^x$ " << endl << endl;
cout << "Введіть точність розрахунків та аргумент (x)\nУ послідовності перша - це точність, далі друга -
аргумент: ";
cin >> Accuracy >> x_2;

Checking_number_for_second_task(Accuracy, x_2);
second_task(Accuracy, x_2);

cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:";
cin >> choice_to_try_again_2;
while (choice_to_try_again_2 == 1) {
    switch (choice_to_try_again_2) {
        case 1:
            cout << "Введіть точність розрахунків та аргумент (x)\nУ послідовності перша - це точність,
далі друга - аргумент: ";
            cin >> Accuracy >> x_2;
            cout << endl;
            Checking_number_for_second_task(Accuracy, x_2);
            second_task(Accuracy, x_2);
            cout << endl;
            cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:";
            cin >> choice_to_try_again_2;
            break;

        default:
            cout << "Програма закінчила свою роботу";
            break;
    }
}

system("pause");
return 0;
}#include <math.h>
#include <Windows.h>
#include <iostream>
#include <iomanip>
using namespace std;

double Function(double x) {
    double Sqrt_Sin, Sqrt_of_Exp, Drib;

```

```

    Sqrt_Sin = sqrt(pow(sin(x / 2), 3));
    Sqrt_of_Exp = pow(exp(1.3 * x) + exp(-1.3 * x), 1.0 / 3);
    Drib = 1.0 / abs(x + 5.0 / 2);
    return (Sqrt_Sin + Sqrt_of_Exp) * Drib;
}

void first_task(double x_start, double x_end, double x_shag) {
    double x, y;
    cout << "|-----|-----|" << endl;
    cout << "|    x    |    y    |" << endl;
    cout << "|-----|-----|" << endl;

    for (x = x_start; x <= x_end; x += x_shag) {
        if (x > 5 && x < 10 && x != -5.0 / 2) {
            y = Function(x);

        }
        else if (x > -5 && x <= 2) {
            y = x;

        }
        else {
            y = 1.0 / pow(x, 2);

        }

        cout << "|    " << setw(8);
        if (x < 0.0000001 && x > -0.0000001) cout << "0";
        else cout << x;
        cout << "|    " << setw(19);
        if (x >= 6 && x <= 10) cout << "Не входить в ОДЗ";
        else if (x < 0.0000001 && x > -0.0000001) cout << "0";
        else cout << y;
        cout << " |" << endl;
    }

    cout << "|-----|-----|" << endl;
}

void second_task(double Accuracy, int x_2) {
    double current_member = x_2;
    double sum = 1, sum_while = 1, sum_do_while = 1;
    double i, j = 0;
    for (i = 0; abs(current_member) > Accuracy; j++, sum += current_member, i++)
    {
        current_member *= -(x_2 * x_2) / (i + 1);
        if (j == 7) cout << "Сума перших 5 членів ряду Тейлора =" << setprecision(16) << sum << endl;
    }

    j = 0;
    i = 0;
    current_member = x_2;

    while (abs(current_member) > Accuracy)
    {
        current_member *= -(x_2 * x_2) / (i + 1);

        j++;
        sum_while += current_member;
        i++;
    }
}

```

```

j = 0;
i = 0;
current_member = x_2;

do
{
    current_member *= -(x_2 * x_2) / (i + 1);

    sum_do_while += current_member;
    j++;
    i++;
} while (abs(current_member) > Accuracy);

cout << setprecision(18) << "Сума ряду Тейлора = " << sum << endl;
cout << setprecision(18) << "Сума ряду Тейлора = " << sum_while << endl;
cout << setprecision(18) << "Сума ряду Тейлора = " << sum_do_while << endl;
cout << "\t e^-x^2 = " << setprecision(18) << exp(-(x_2 * x_2)) << endl;
cout << "Кількість ітерацій = " << j << endl;
}

void Checking_number_for_first_task(double& x_start, double& x_end, double& x_shag) {
    while (((x_start || x_end || x_shag) == !cin) || x_end < x_start || x_shag <= 0) {
        if ((x_start || x_end || x_shag) == !cin) {
            cout << "Ви ввели не число" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin.clear();
            cin.ignore(32767, '\n');
            cin >> x_start >> x_end >> x_shag;
        }
        else if (x_end < x_start && x_shag <= 0) {
            cout << "Кінець інтервалу повинен бути більше за початок, та шаг повинен бути додатнім"
<< endl;

            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin >> x_start >> x_end >> x_shag;
        }
        else if (x_end < x_start) {
            cout << "Кінець інтервалу повинен бути більше за початок" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin >> x_start >> x_end >> x_shag;
        }
        else if (x_shag <= 0) {
            cout << "Шаг повинен бути додатнім (він не може бути нулем або від'ємним)" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin >> x_start >> x_end >> x_shag;
        }
    }
}

void Checking_number_for_second_task(double& Accuracy, int& x_2) {
    cin.clear();
    cin.ignore(32767, '\n');
    while ((Accuracy == !cin) || (x_2 == !cin)) {
        if ((Accuracy == !cin) || (x_2 == !cin)) {
            cout << "\nВи ввели не число\n";

```

```

        cout << "Введіть точність розрахунків та аргумент (x)\nУ послідовності перша - це точність,
далі друга - аргумент: ";
        cin >> Accuracy >> x_2;
    }
    cin.clear();
    cin.ignore(32767, '\n');
}

cout << endl;
}

int main(int argc, char* argv[])
{
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
    cout << "\t\tЛабораторна робота № 6" << endl;

    cout << "\t\tЗавдання № 1" << endl << endl;
    double x_start, x_end, x_shag;
    int choice_to_try_again;
    if (argc == 4)
    {
        x_start = atof(argv[1]);
        x_end = atof(argv[2]);
        x_shag = atof(argv[3]);
        if ((x_start || x_end || x_shag) == !cin) {
            cout << "Ви ввели не число" << endl;
            cout << "Введіть значення знову у послідовності (початок інтервалу, кінець інтервалу та
шаг):";

            cin.clear();
            cin.ignore(32767, '\n');
            cin >> x_start >> x_end >> x_shag;
        }
    }
    else
    {
        cout << "Ви не ввели всі значення, будь ласка введіть значення у цій послідовності" << endl;
        cout << "Введіть початок інтервалу, кінець інтервалу та шаг: ";
        cin >> x_start >> x_end >> x_shag;
    }

    Checking_number_for_first_task(x_start, x_end, x_shag);

    first_task(x_start, x_end, x_shag);

    cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити.";
    cin >> choice_to_try_again;
    while (choice_to_try_again == 1) {
        switch (choice_to_try_again) {
            case 1:
                cout << "Введіть початок інтервалу, кінець інтервалу та шаг: ";
                cin >> x_start >> x_end >> x_shag;
                Checking_number_for_first_task(x_start, x_end, x_shag);
                cout << endl;
                first_task(x_start, x_end, x_shag);
                cout << endl;
                cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 -
закінчити:";

```



```

        cin >> choice_to_try_again;
        break;
    default:
        cout << "Програма переходить до іншого завдання ";
        break;
    }
}

cout << "\t\tЗавдання № 2" << endl << endl;
double Accuracy;
int x_2;
int choice_to_try_again_2;

cout << "Калькулятор  $e^{-x^2}$ " << endl << endl;
cout << "Введіть точність розрахунків та аргумент (x)\nУ послідовності перша - це точність, далі друга - аргумент: ";
cin >> Accuracy >> x_2;

Checking_number_for_second_task(Accuracy, x_2);
second_task(Accuracy, x_2);

cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:";
cin >> choice_to_try_again_2;
while (choice_to_try_again_2 == 1) {
    switch (choice_to_try_again_2) {
        case 1:
            cout << "Введіть точність розрахунків та аргумент (x)\nУ послідовності перша - це точність, далі друга - аргумент: ";
            cin >> Accuracy >> x_2;
            cout << endl;
            Checking_number_for_second_task(Accuracy, x_2);
            second_task(Accuracy, x_2);
            cout << endl;
            cout << "Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:";
            cin >> choice_to_try_again_2;
            break;
        default:
            cout << "Програма закінчила свою роботу";
            break;
    }
}

system("pause");
return 0;
}

```

4.Опис інтерфейсу програм

Користувач не може вводити букви тому, що через введення букви програма видасть помилку, аналогічно для обох програм, коли користувач вводить значення то програма виводить відповідну інформацію, далі для завершення програми користувач натискає на будь-яку клавішу клавіатури.

5.Опис тестових прикладів

Лабораторна робота № 6 Завдання № 1

Введіть початок інтервалу: 1
Введіть кінець інтервалу: 10
Введіть шаг: 1

x	y
1	1
2	2
3	0.111111
4	0.0625
5	0.04
6	Не входить в ОДЗ
7	Не входить в ОДЗ
8	Не входить в ОДЗ
9	Не входить в ОДЗ
10	Не входить в ОДЗ

Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:1

Введіть початок інтервалу: -10
Введіть кінець інтервалу: 10
Введіть шаг: 5

x	y
-10	0.01
-5	0.04
0	0
5	0.04
10	Не входить в ОДЗ

Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:0

Завдання № 2

Калькулятор e^{-x^2}

Введіть точність розрахунків та аргумент (x)

У послідовності перша - це точність, далі друга - аргумент: 0.0000000001 1

Сума перших 5 членів ряду Тейлора =0.3678571428571429

Сума ряду Тейлора = 0.367879441172162036

Сума ряду Тейлора = 0.367879441172162036

Сума ряду Тейлора = 0.367879441172162036

e^{-x^2} = 0.367879441171442334

Кількість ітерацій = 14

Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:1

Введіть точність розрахунків та аргумент (x)

У послідовності перша - це точність, далі друга - аргумент: 0.0000000000 -1

Ви ввели не число

Введіть точність розрахунків та аргумент (x)

У послідовності перша - це точність, далі друга - аргумент: 0.0000000001 -1

Сума перших 5 членів ряду Тейлора =1.632142857142857

Сума ряду Тейлора = 1.63212055882783802

Сума ряду Тейлора = 1.63212055882783802

Сума ряду Тейлора = 1.63212055882783802

e^{-x^2} = 0.367879441171442334

Кількість ітерацій = 14

Щоб продовжити роботу цієї програми, напишіть, 1 - продовжити, 0 - закінчити:1

Введіть точність розрахунків та аргумент (x)

У послідовності перша - це точність, далі друга - аргумент: 0.0000000001 3

Сума перших 5 членів ряду Тейлора =-1557.767857142858

Сума ряду Тейлора = -1.99962977058351221

Сума ряду Тейлора = -1.99962977058351221

Сума ряду Тейлора = -1.99962977058351221

e^{-x^2} = 0.000123409804086679561

Кількість ітерацій = 42

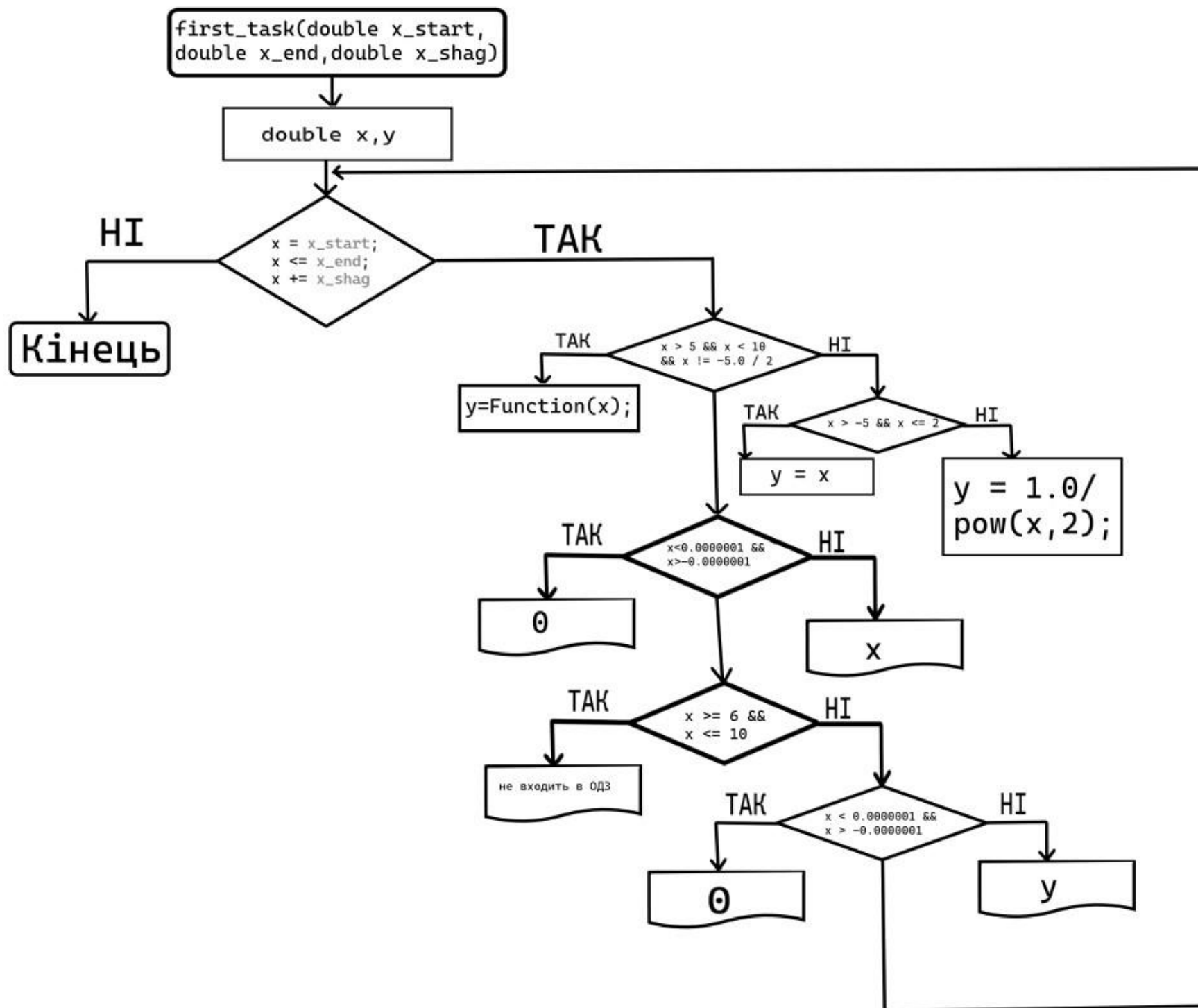
6.Аналіз помилок

Виправив 2 задачу, суму ряду розраховував за допомогою трьох видів циклу, а не суму 5 членів. Виправив значення функції і стандартної.

7.Висновок

Ця лабораторна робота була дуже корисна, тому, що я дізнався про бібліотеку `math.h/ cmath`, ми навчилися використовувати математичні функції які входять в цю бібліотеку, ми повторили використання бібліотеки `io.h`, яка дозволяє використовувати `setw()`, ще ми повторили використання трьох циклів `for`, `while`, `do while`, ще повторили розлагодження, ми дізналися для чого використовувати `int argc, char* argv[]`, ще ми дізналися як використовувати командну строку та повторили функції і передача параметрів у функцію за значенням.

8.Блок-схеми до завдань



Function(double x)



```
double Sqrt_Sin, Sqrt_of_Exp, Drib;  
Sqrt_Sin = sqrt(pow(sin(x / 2), 3));  
Sqrt_of_Exp = pow(exp(1.3 * x) + exp(-1.3 * x), 1.0 / 3);  
Drib = 1.0 / abs(x + 5.0 / 2);  
return (Sqrt_Sin + Sqrt_of_Exp) * Drib;
```