

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний аерокосмічний університет ім. М. Є. Жуковського
«Харківський авіаційний інститут»

Кафедра систем управління літальними апаратами

Лабораторна робота № 5

з дисципліни «Алгоритмізація та програмування»

Тема: «Реалізація циклічних алгоритмів на мові C ++»

Виконав:
студент(ка) гр. 319а

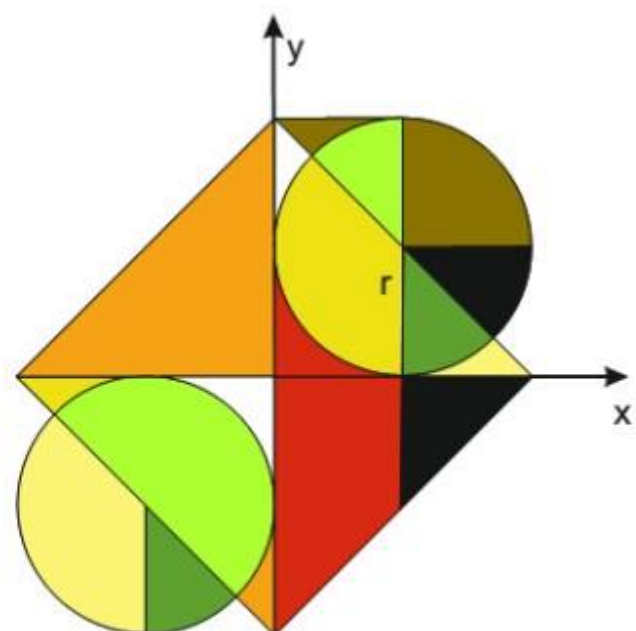
Перевірив:
к.т.н., доцент каф.301
Гавриленко О.В.

МЕТА РОБОТИ

Вивчити теоретичний матеріал із синтаксису мовою C++ і поданням у вигляді блок-схем циклічних алгоритмів і реалізувати алгоритми з використанням інструкцій циклу з перед-умовою, циклу з після-умовою і параметричного циклу мовою C++ в середовищі Visual Studio.

ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Дано дійсні числа (x_i, y_i) , $i = 1, 2, \dots, n$, – координати точок на площині. Визначити кількість точок, що потрапляють в фігуру заданого кольору (або групу фігур). Варіанти фігур представлено в табл.1.

11-19	<p>известно: r - радиус окружностей.</p> <div data-bbox="319 828 574 1456"> <p>11 вариант</p> <p>12 вариант</p> <p>13 вариант</p> <p>14 вариант</p> <p>15 вариант</p> <p>16 вариант</p> <p>17 вариант</p> <p>18 вариант</p> <p>19 вариант</p> </div> 
-------	--

Завдання 2. Дано дійсне число x і натуральне число n . Необхідно:

- Обчислити значення виразу при заданих x і n для виразу з табл.2.
- Вивести: для парних варіантів – значення кожного третього елемента, для непарних – значення кожного четвертого елемента.

15	$(x^{1/3} - x) + (x^{1/5} - x^{1/3}) + (x^{1/7} - x^{1/5}) + \dots + (x^{1/(2n+1)} - x^{1/(2n-1)})$
----	---

Завдання 3. Дослідити ряд на збіжність. Умова закінчення циклу обчислення суми прийняти у вигляді: $|u_n| < \epsilon$ або $|u_n| > g$, де ϵ – мала

величина для переривання циклу обчислення суми збіжного ряду ($\epsilon = 10^{-5} \dots 10^{-20}$); g – величина для переривання циклу обчислення суми розбіжного ряду ($g = 10^2 \dots 10^5$). Варіанти представлено в табл.3.

38.	$\sum_{n=1}^{\infty} \frac{x^{n-1} + 3n}{(2n-1)!}$
-----	--

Завдання 4. Організувати меню в командному вікні для багаторазового виконання завдань та для перевірки вхідних даних на коректність описати функції, що повертають логічне значення (true – в разі коректного значення переданих параметрів і false – в іншому випадку).

ВИКОНАННЯ РОБОТИ

Вирішення задачі: 1

Вхідні дані (ім'я, опис, тип):

int n, x, y - цілочислений

n – кількість точок

x, y – координати точок

Вирішення задачі: 2

Вхідні дані (ім'я, опис, тип):

int - x – цілочислений - змінна

int – n – цілочислений – кількість

елементів

Вихідні дані (ім'я, опис, тип):

int counter – кількість точок, що

попадуть у фігуру

Вихідні дані (ім'я, опис, тип):

int z – з плаваючою точкою -

результат

Вирішення задачі: 3

Вхідні дані (ім'я, опис, тип):

int n – цілочислений – кількість

елементів

int x – цілочислений – змінна виразу

Вирішення задачі: 4

Вхідні дані (ім'я, опис, тип):

int menu – вибір завдання

Вихідні дані (ім'я, опис, тип):

double - sum – з плаваючою точкою

Вихідні дані (ім'я, опис, тип):

void tab11, tab12, tab13 – пустий тип-
варіанти завдань

Лістинг файлу Source.cpp наведено в дод.А. Екран роботи програми показаний в дод. Б.

ВИСНОВКИ

Під час виконання лабораторної роботи були покращені навички роботи за структурами та більш детальніше розуміння роботи циклів. Були вирішені проблеми з ініціалізацією змінних, відточення розуміння роботи різних циклів та їх особливостей.

ДОДАТОК А

Лістинг коду програми до завдань: 1,2,3,4

```

#include<iostream>
#include<cmath>
using namespace std;

void tab11();
void tab12();
void tab13();

int main() {
    setlocale(LC_ALL, "Russian");
    int menu;
    do {
        cout << "Доступные задания: 1 / 2 / 3" << endl;
        cout << "Выбрать задание под номером:";
        cin >> menu;
        switch (menu)
        {
            case 1: tab11(); break;
            case 2: tab12(); break;
            case 3: tab13(); break;
            default: cout << "Ошибка. Необходимо выбрать одно из доступных
заданий. Повторите выбор!" << endl;
        }
    } while (menu != -1);
}

void tab11() {
    int x, y, n, a, b, couter = 0;
    cout << " \n ";
    cout << "Задание 1\n ";
    cout << "Даны действительные числа (x, y) координаты точек на
плоскости.\n Определить количество точек, попадающих в фигуру заданного
цвета." << endl;
    cin >> n;
    for (int j = 1; j <= n; j++) {
        cout << "Введем X и Y: ";
        cin >> x >> y;
        if (x < 0 && y < 0 && x + y > 0 && x - y < 0) {
            couter++;
        }
        else if (x > 0 && y > 0 && y > x && y > -x) {
            couter++;
        }
    }
    cout << "Количество точек, которые попали в фигуру: " << couter <<
" \n ";
}

void tab12() {
    int x, n;
    cout << "\n Задание 2 \n Дано действительное число x и натуральное
число n. Необходимо: \n а) Вычислить значение выражения при заданных x и
n для выражения \n б) Вывести: для парных вариантов - значение каждого
третьего \n элемента, для нечетных - значение каждого четвёртого
элемента.\n " << endl;

```

```

cout << "Введите X: ";
cin >> x;
int result = 0;
int power1 = 1; // Для хранения значения  $x^{(1/(2*n+1))}$ 
int power2 = 1; // Для хранения значения  $x^{(1/(2*n-1))}$ 
cout << "Введите N: ";
cin >> n;
for (int i = 1; i <= n; i++) {
    for (int j = 0; j < (1 / (2 * i + 1)); j++) {
        power1 *= x;
    }
    for (int j = 0; j < (1 / (2 * i - 1)); j++) {
        power2 *= x;
    }
    int element = power1 - power2;
    if (i % 4 == 0) {
        result += element;
        cout << "Элемент " << i << ": " << element << endl;
    }
    power1 = 1; // Сброс значения power1 перед новой итерацией
    power2 = 1; // Сброс значения power2 перед новой итерацией
    cout << "Сумма четвертых элементов: " << result << endl;
}
}
void tabl3() {
    double x;
    int n;
    double sum = 0.0;
    const double epsilon = 1e-5; // Малая величина для сходимости ряда
    const double gamma = 1e5;    // Большая величина для расходимости
    ряда
    cout << "Введите значение x: ";
    cin >> x;
    cout << "Введите количество элементов для суммирования: ";
    cin >> n;
    for (int i = 1; i <= n; ++i) {
        double term = pow(x, i - 1) + (3.0 * i) / tgamma(2 * i - 1);
        if (fabs(term) < epsilon || fabs(term) > gamma) {
            cout << "Сумма оборвана на " << i << " элементе.
Значение суммы: " << sum << endl;
            break;
        }
        sum += term;
    }
    cout << "Итоговая сумма ряда: " << sum << endl;
}
}

```

ДОДАТОК Б

Скрін-шоти програми

```

Задание 1
Даны действительные числа (x, y) координаты точек на плоскости.
Определить количество точек, попадающих в фигуру заданного цвета.
4
Введем X и Y: -1 -1
Введем X и Y: 5 1
Введем X и Y: 8 -2
Введем X и Y: 0 0
Количество точек, которые попали в фигуру: 0

```

Рисунок Б.1 - Экран виконання програми для вирішення завдання: 1

```

Задание 2
Дано действительное число x и натуральное число n. Необходимо:
а) Вычислить значение выражения при заданных x и n для выражения
б) Вывести: для парных вариантов – значение каждого третьего
элемента, для нечетных – значение каждого четвертого элемента.

Введите X: 6
Введите N: 3
Сумма четвертых элементов: 0
Сумма четвертых элементов: 0
Сумма четвертых элементов: 0

```

Рисунок Б.2 - Экран виконання програми для вирішення завдання: 2

```

Выбрать задание под номером:3
Введите значение x: 6
Введите количество элементов для суммирования: 3

```

Рисунок Б.3 - Экран виконання програми для вирішення завдання: 3

```

Доступные задания: 1 / 2 / 3
Выбрать задание под номером:4
Ошибка. Необходимо выбрать одно из доступных заданий. Повторите выбор!
Доступные задания: 1 / 2 / 3
Выбрать задание под номером:2

Задание 2

```

Рисунок Б.4 - Экран виконання програми для вирішення завдання: 4