МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний аерокосмічний університет ім. М. Є. Жуковського

«Харківський авіаційний інститут»

Факультет систем управління літальних апаратів

Кафедра систем управління літальних апаратів

**Лабораторна робота № 5**

з дисципліни «Алгоритмізація та програмування»

на тему "Реалізація циклічних алгоритмів мовою С ++"

ХАІ.301.173.310.5ЛР

Виконав студент гр. \_\_\_\_\_310\_\_\_\_\_\_\_\_\_\_

25.12.2023\_\_\_\_\_\_\_Шуба Анастасія *\_\_\_\_\_*

(підпис, дата) (П.І.Б.)

Перевірив

\_\_\_\_\_\_\_\_\_ к.т.н., доц. Олена  ГАВРИЛЕНКО

(підпис, дата) (П.І.Б.)

2023

# МЕТА РОБОТИ

Вивчити теоретичний матеріал із синтаксису мовою С ++ і поданням у вигляді UML діаграм циклічних алгоритмів і реалізувати алгоритми з використанням інструкцій циклу з передумовою, циклу з післяумовою і параметризованого циклу мовою C ++ в середовищі Visual Studio.

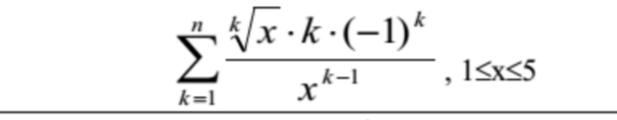
# ПОСТАНОВКА ЗАДАЧІ

Завдання 1. Дано дійсні числа (xi , yi), i = 1,2, ... n, – координати точок на площині. Визначити кількість точок, що потрапляють в фігуру заданого кольору (або групу фігур).

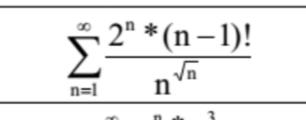
Завдання 2. Дано дійсне число х і натуральне число n. Необхідно:

a) Обчислити значення виразу при заданих x і n для виразу з табл.2.

b) Вивести: для парних варіантів – значення кожного третього елемента, для непарних – значення кожного четвертого елемента.



Завдання 3. Дослідити ряд на збіжність. Умова закінчення циклу обчислення суми прийняти у вигляді: | un | < е або | un | > g, де е – мала величина для переривання циклу обчислення суми збіжного ряду (е = 10-5 ... 10-20); g – величина для переривання циклу обчислення суми розбіжного ряду (g = 102 ... 105).



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

Завдання 1.

Вирішення задачі №9

Вхідні дані:

x - дійсний тип, 1<x<5

k - дійсний тип, k > 0

Вихідні дані:

x ( 1<x<5 ) =3

n = 5

Результат суми = -1.00502

Алгоритм вирішення:

1. Введення значень x та n
2. Ініціалізація суми
3. Обчислення чисельника
4. Обчислення знаменника
5. Додавання або віднімання елемента залежно від k
6. Виведення результату

Лістинг коду вирішення задачі розділ і номер задачі наведено в дод. А (стор. х).

Екран роботи програми показаний на рис. Б.1

Завдання 2.

Вирішення задачі №17

Вхідні дані:

n - дійсний тип, n < 0

Вихідні дані:

n = 60

Результат суми = 2.72784e+84

Алгоритм вирішення:

1. Введення значення n
2. Ініціалізація суми
3. Обчислення чисельника
4. Обчислення знаменника
5. Додавання до суми
6. Виведення результату

Лістинг коду вирішення задачі розділ і номер задачі наведено в дод. А1 (стор. х).

Екран роботи програми показаний на рис. Б.2

# ВИСНОВКИ

Вивчила теоретичний матеріал із синтаксису мовою С++. Реалізувала алгоритми з використанням інструкцій циклу з передумовою, циклу з післяумовою і параметризованого циклу мовою C ++ в середовищі Visual Studio.

ДОДАТОК А

Лістинг коду програми

#include <iostream>

#include <cmath>

int main() {

double x;

int n;

// Введення значень x та n

std::cout << "Введіть значення x (1 < x < 5): ";

std::cin >> x;

std::cout << "Введіть значення n: ";

std::cin >> n;

// Ініціалізація суми

double sum = 0.0;

for (int k = 1; k <= n; ++k) {

// Обчислення чисельника

double numerator = k \* sqrt(x);

// Обчислення знаменника

double denominator = pow(x, k - 1);

// Додавання або віднімання елемента залежно від k

sum += pow(-1, k) \* (numerator / denominator);

}

// Виведення результату

std::cout << "Результат суми: " << sum << std::endl;

return 0;

}

ДОДАТОК А1

Лістинг коду програми

#include <iostream>

#include <cmath>

double factorial(int n) {

if (n == 0 || n == 1) {

return 1.0;

} else {

return n \* factorial(n - 1);

}

}

int main() {

int n;

// Введення значення n

std::cout << "Введіть значення n: ";

std::cin >> n;

// Ініціалізація суми

double sum = 0.0;

for (int i = 1; i <= n; ++i) {

// Обчислення чисельника

double numerator = pow(2, i) \* factorial(i - 1);

// Обчислення знаменника

double denominator = pow(i, sqrt(i));

// Додавання до суми

sum += numerator / denominator;

}

// Виведення результату

std::cout << "Результат суми: " << sum << std::endl;

return 0;

}

ДОДАТОК Б

Скрін-шоти вікна виконання програми

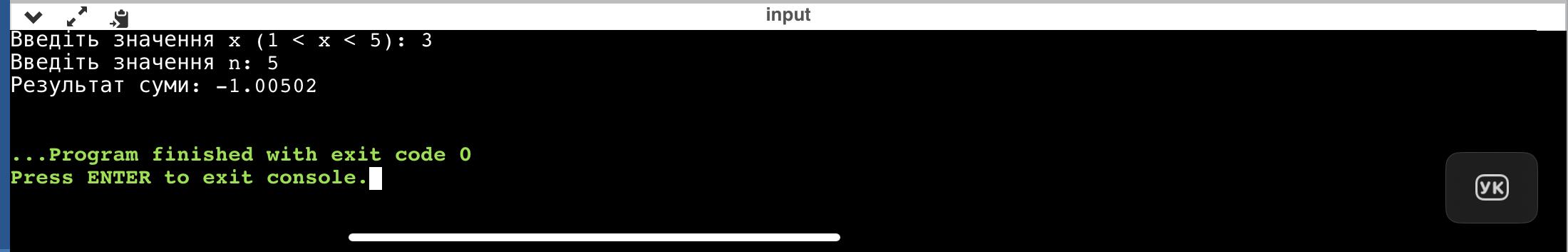


Рисунок Б.1 – Екран виконання програми для вирішення завдання 1

Задача №19

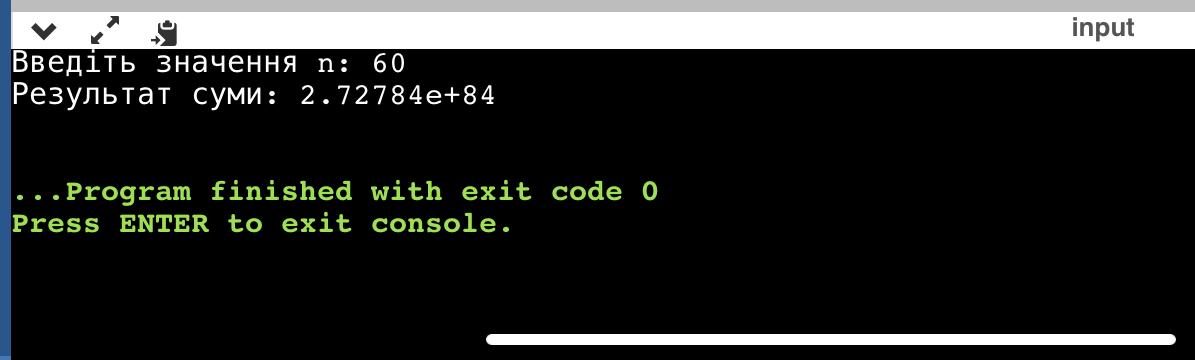


Рисунок Б.2 – Екран виконання програми для вирішення завдання 2

Задача №17