ЗВІТ З ЛАБОРАТОРНОЇ РОБОТИ

за курсом “Програмування”

студента групи ПА-22-2

Овдієнка Андрія Володимировича

кафедра комп’ютерних технологій, ДНУ

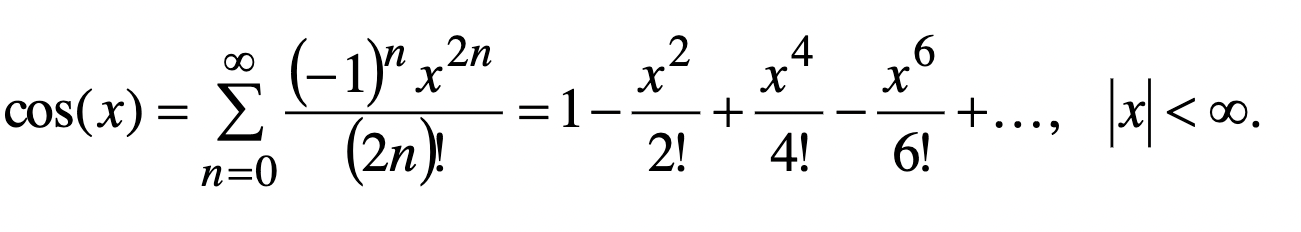
2022/2023

1.Завдання до лабораторної роботи №6.2

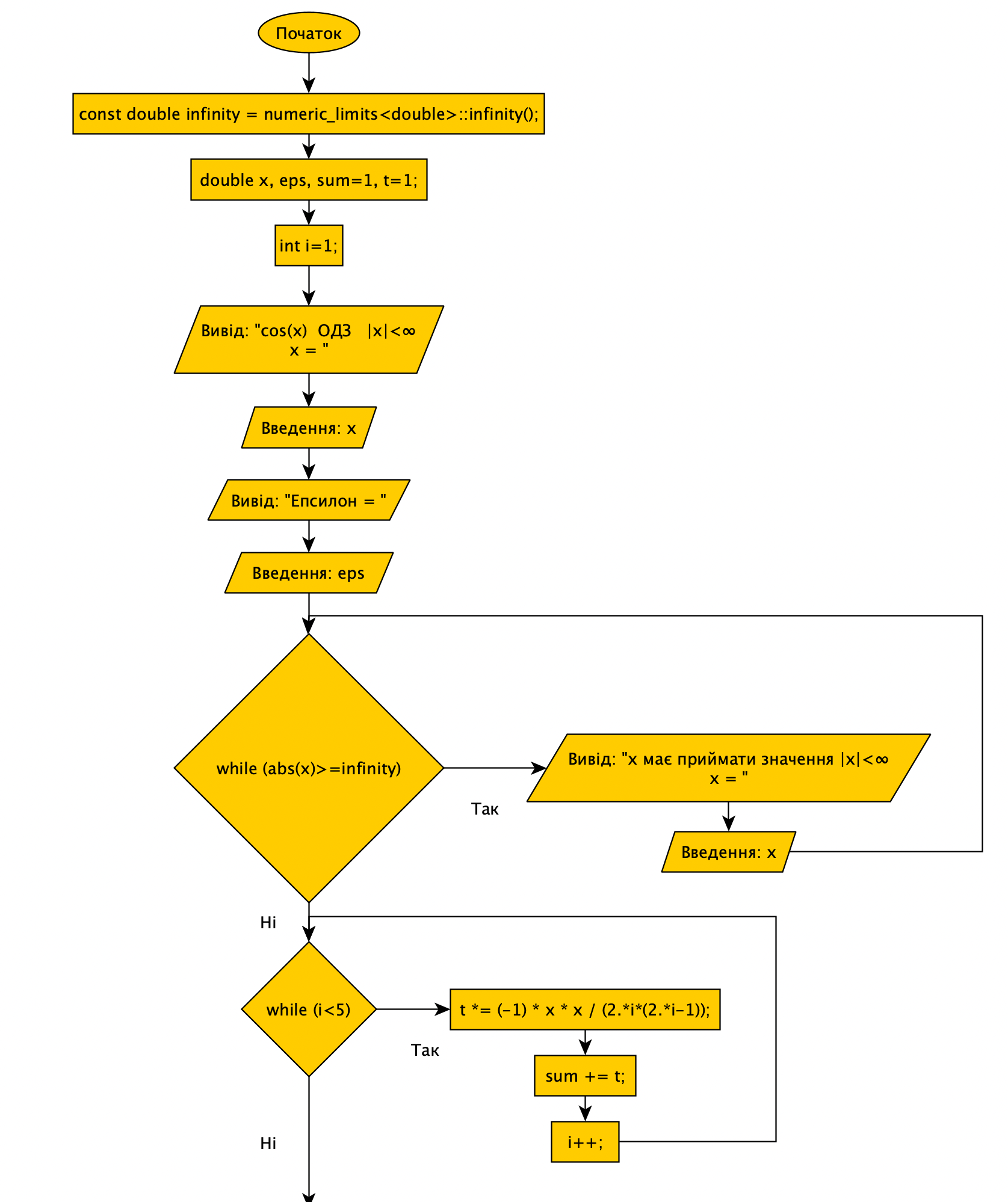
Постановка задачі:

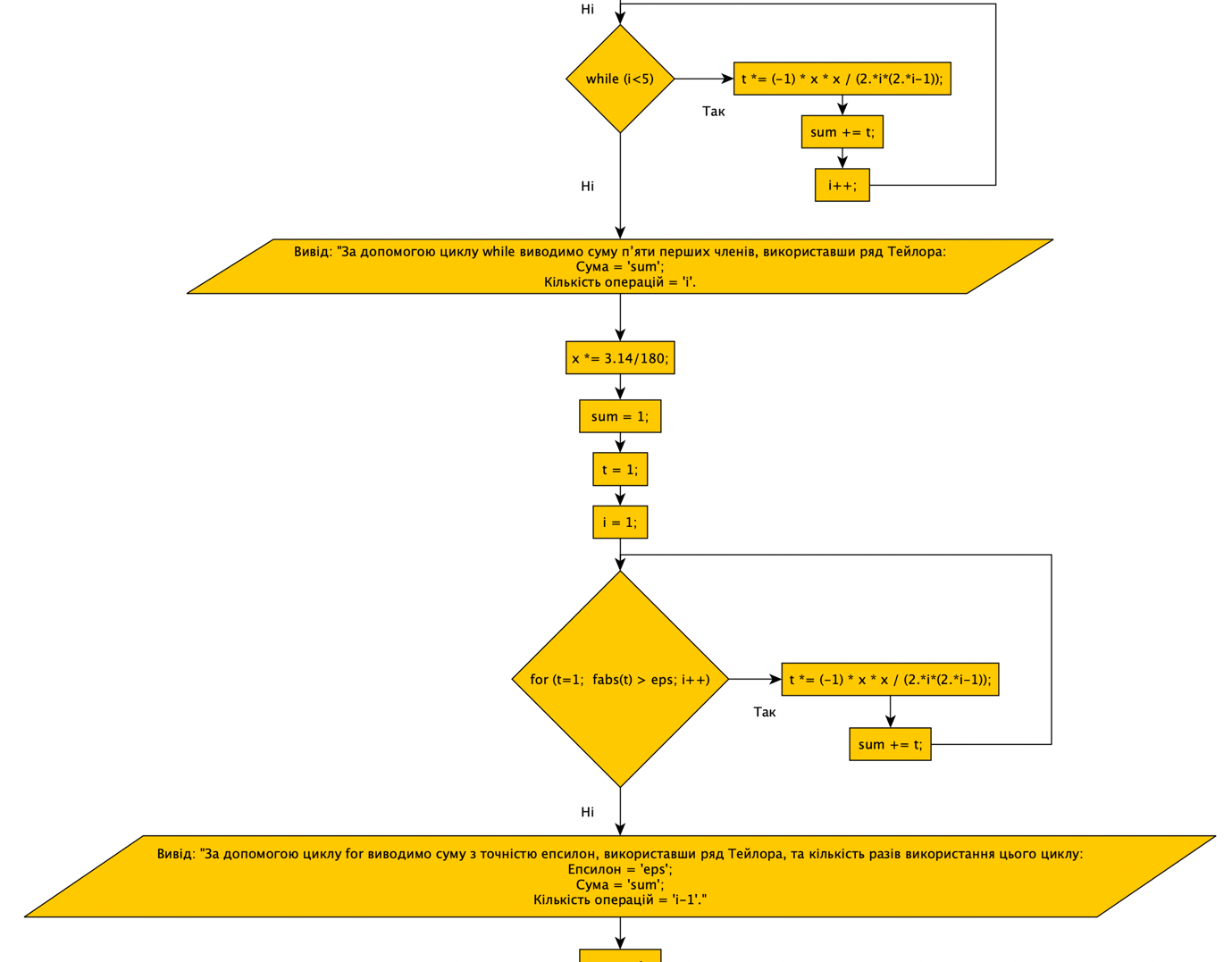
Складіть програму, яка обчислює значення функції, заданого за допомогою ряду Тейлора. Для ряду, члени якого обчислюються за формулою, що відповідає вашому індивідуальному завданню, підрахуйте та виведіть суму перших 5 членів ряду, суму членів ряду із у веденою точністю ε, кількість ітерацій та значення функції, що розрахована за допомогою стандартних математичних функцій. Організуйте вихід із програми за бажанням користувача. Завдання виконайте за допомогою трьох операторів циклу.

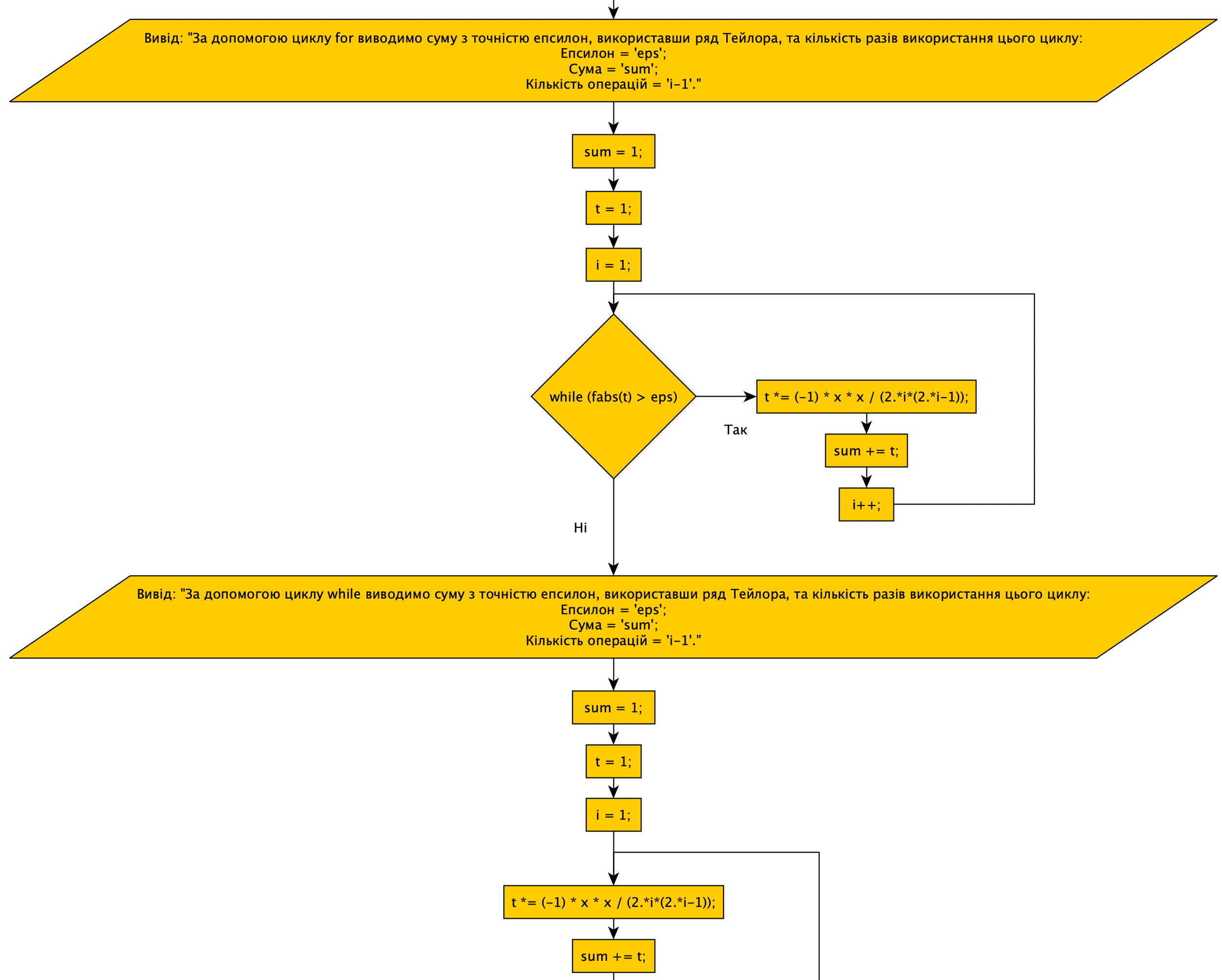
Індивідуальне завдання:

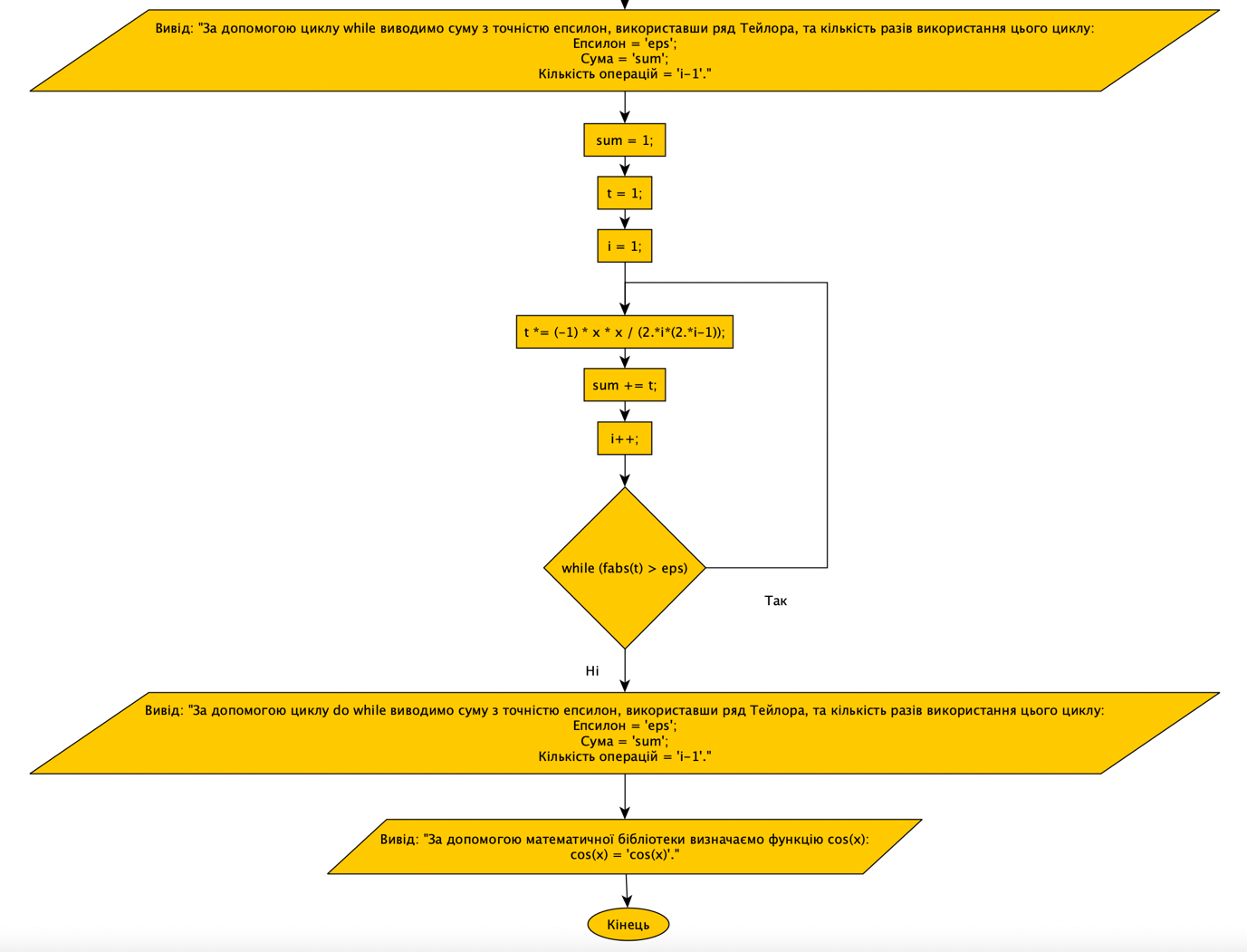


2.Опис розв’язку (блок-схема)









3.Вихідний текст програми розвʼязку задачі (код програми з коментарями)

#include <iostream> // Підключаємо бібліотека для вводу й виводу тексту на консоль.

#include <cmath> // Підключаємо бібліотека для використання математичних функцій.

#include <limits> // Підключаємо бібліотеку для використання границь.

using namespace std; // Вказуємо, який простір імен.

const double infinity = numeric\_limits<double>::infinity(); // Оголошуємо константу нескінченність.

int main() // Програма на С++ починається із виконання функції main.

{

double x, eps, sum=1, t=1; // Оголошуємо змінні: 'x', 'eps', 'sum' приймає значення одиниці,'t' приймає значення одиниці.

int i=1; // Оголошуємо змінну 'i', приймає значення одиниці.

cout<<"\ncos(x)\t\t ОДЗ\n\t\t|x|<∞\n\nx = "; // Нова строка, виводимо символи на консоль, горизонтальна табуляція (двічі), виводимо символи на консоль, нова строка, горизонтальна табуляція (двічі), виводимо символи на консоль, нові строки (дві), виводимо символи на консоль.

cin >> x; // Bводимо значення 'x'.

cout<<"Епсилон = "; // Виводимо символи на консоль.

cin >> eps; // Bводимо значення 'eps'.

while (abs(x)>=infinity) // Поки модуль від 'x' більше нескінченності.

{

cout << "х має приймати значення |x|<∞\nх = "; // Виводимо символи на консоль, нова строка, виводимо символи на консоль.

cin >> x; // Bводимо значення 'eps'.

}

while (i<5) // Поки 'i' строго менше пʼяти.

{

t \*= (-1) \* x \* x / (2.\*i\*(2.\*i-1)); // Рахуємо cos(x) через ряд тейлора.

sum += t; // 'sum' приймає своє значення та додає значення 't'.

i++; // До 'i' додаємо один.

}

cout << "\n\nЗа допомогою циклу while виводимо суму пʼяти перших членів, використавши ряд Тейлора:\nСума = " << sum << ";\nКількість операцій = "<< i <<".\n\n"; // Нові сроки (дві), виводимо на консоль символи,нова строка, виводимо на консоль символи, виводимо значення 'sum', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'і', виводимо на консоль символ, нові строки (двічі).

x \*= 3.14/180; // 'x' приймає значення радіану.

sum=1; // 'sum' приймає значення одиниці.

t=1; // 't' приймає значення одиниці.

i=1; // 'i' приймає значення одиниці.

for (t=1; fabs(t) > eps; i++) // Надаємо змінну 't', приймає значення одиниці, поки модуль 't' строго більше 'eps', до 'i' додаємо одиницю.

{

t \*= (-1) \* x \* x / (2.\*i\*(2.\*i-1)); // Рахуємо cos(x) через ряд тейлора.

sum += t; // 'sum' приймає своє значення та додає значення 't'.

}

cout << "За допомогою циклу for виводимо суму з точністю епсилон, використавши ряд Тейлора, та кількість разів використання цього циклу:\nЕпсилон =" << eps << ";\nСума = " << sum << ";\nКількість операцій = " << i-1 << ".\n\n"; // Виводимо на консоль символи, нова строка, виводимо на консоль символи, виводимо значення 'eps', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'sum', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'i' на одиницю менше, виводимо на консоль символ, нова строка.

sum=1; // 'sum' приймає значення одиниці.

t=1; // 't' приймає значення одиниці.

i=1; // 'i' приймає значення одиниці.

while (fabs(t) > eps) // Поки модуль 't' строго більше 'eps'.

{

t \*= (-1) \* x \* x / (2.\*i\*(2.\*i-1)); // Рахуємо cos(x) через ряд тейлора.

sum += t; // 'sum' приймає своє значення та додає значення 't'.

i++; // До 'i' додаємо один.

}

cout << "За допомогою циклу while виводимо суму з точністю епсилон, використавши ряд Тейлора, та кількість разів використання цього циклу:\nЕпсилон =" << eps << ";\nСума = " << sum << ";\nКількість операцій = " << i-1 << ".\n\n"; // Виводимо на консоль символи, нова строка, виводимо на консоль символи, виводимо значення 'eps', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'sum', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'i' на одиницю менше, виводимо на консоль символ, нова строка.

sum=1; // 'sum' приймає значення одиниці.

t=1; // 't' приймає значення одиниці.

i=1; // 'i' приймає значення одиниці.

do // Робити.

{

t \*= (-1) \* x \* x / (2.\*i\*(2.\*i-1)); // Рахуємо cos(x) через ряд тейлора.

sum += t; // 'sum' приймає своє значення та додає значення 't'.

i++; // До 'i' додаємо один.

} while (fabs(t) > eps); // Поки модуль 't' строго більше 'eps'.

cout << "За допомогою циклу do while виводимо суму з точністю епсилон, використавши ряд Тейлора, та кількість разів використання цього циклу:\nЕпсилон =" << eps << ";\nСума = " << sum << ";\nКількість операцій = " << i-1 << ".\n\n"; // Виводимо на консоль символи, нова строка, виводимо на консоль символи, виводимо значення 'eps', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'sum', виводимо на консоль символ, нова строка, виводимо на консоль символи, виводимо значення 'i' на одиницю менше, виводимо на консоль символ, нова строка.

cout << "За допомогою математичної бібліотеки визначаємо функцію cos(x):\ncos(x) = "<< cos(x) << ".\n"; // Виводимо но консоль символи, виводимо дійсне значення функції за допомогою математичної бібліотеки, виводимо на консоль символ, переходимо на нову строку.

return 0; // Без повторень.

}

4.Опис інтерфейсу програми (керівництво користувача)

