ЗВІТ  
про виконання лабораторної роботи № 5.5  
« Рекурсивні функції »  
з дисципліни  
«Алгоритмізація та програмування»  
студентки групи ІК-12  
Пилипів Яни Вікторівни

Умова завдання:

Написати рекурсивну функцію, яка методом ділення відрізання навпіл (методом дихотомії) знаходить з точністю eps корінь рівняння f (x) = sin( x) − cos(x) = 0 на відрізку [a,b] (eps < 0, a < b, f (a)⋅ f (b) < 0) . Згідно з методом дихотомії, якщо f (a) і f (b) мають різні знаки, то між точками a, b існує корінь x. Нехай m–середня точка в інтервалі [a,b] . Якщо f (m) = 0 , то корінь x = m . Якщо m ≠ 0 то або f (a) і f (m) мають різні знаки, або f (m) і f (b) мають різні знаки. Якщо f (a)⋅ f (m) < 0 , то корінь лежить в інтервалі a ≤ x ≤ m . Інакше він лежить в інтервалі m ≤ x ≤ b . Процес продовжується доти, поки інтервал не стане менший eps. Визначити глибину рекурсії під час обчислення кореня рівняння.

Блок-схеми алгоритму програми та кожної функції:

UML-діаграма дії програми та кожної функції:

Структурна схема програми:

Текст програми:

#include <iostream>

#include <cmath>

using namespace std;

double f(double x) {

return sin(x) - cos(x);

}

//дихотомія

double H(double a, double b, double eps, int &d) {

d++;

double m = (a + b) / 2.0;

if (fabs(b - a) < eps || fabs(f(m)) < eps) {

return m;

}

if (f(a) \* f(m) < 0) {

return H(a, m, eps, d);

}

else {

return H(m, b, eps, d);

}

}

int main() {

double a, b, eps;

int d = 0;

cout << "[a, b]: " << endl;

cin >> a >> b;

cout << "eps: ";

cin >> eps;

if (f(a) \* f(b) >= 0) {

cout << "x doesn't exist" << endl;

}

else {

// Виклик функції для пошуку кореня

double x = H(a, b, eps, d);

cout << "x: " << x << endl;

cout << "depth: " << d << endl;

}

return 0;

}

Посилання на git-репозиторій з проектом:

https://github.com/pylypivyana/labs\_ap.git

Результати unit-тесту:

Висновки:

В ході даної лабораторної роботи я навчилася використовувати рекурсивні функції.