Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій



Звіт

про виконання лабораторної роботи №1.1

Поля та методи – дії над одним (поточним) об’єктом

з дисципліни «Об’єктно-орієнтоване програмування»

студента IT-11 Лісничука Арсена

Прийняв доцент Григорович В.Г.

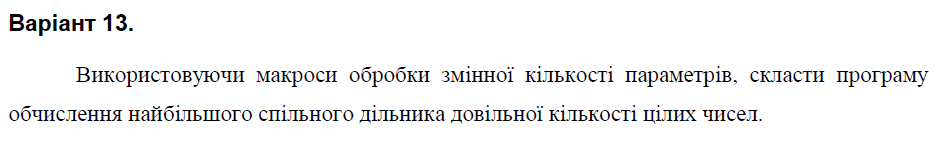
2022

Мета:

вивчити особливості використання директив препроцесора;

навчитися застосовувати директиви препроцесора під час налагодження програми;

Умова завдання:



Структурна схема програми:



Текст програми:

#include <iostream>

#include <cmath>

#include <Windows.h>

using namespace std;

#define TASK2

#define z 4

#define INPUT(x) cin >> x

#define PRINT(str) cout << (str)

#define ABS(x) (((x)<0)?(-x):(x))

#define MAX(a,b) (((a) > (b)) ? (a) : (b))

#define MIN(a,b) (((a) < (b)) ? (a) : (b))

#define POW3(x) ((x)\*(x)\*(x))

#define POW2(x) ((x)\*(x))

int gcd(int a, int b) {

return (a == 0) ? b : gcd(b % a, a);

}

int NOD(int cnt, ...)

{

int res = 0;

va\_list ptr;

va\_start(ptr, cnt);

for (; cnt > 0; cnt--) {

res = gcd(res, va\_arg(ptr, int));

}

va\_end(ptr);

return res;

}

template <typename ... Args> int ArgsCount(Args ... args) {

return sizeof...(args);

}

void main()

{

#ifdef TASK1

double x, y, w = 0;

PRINT("x = "); INPUT(x);

PRINT("y = "); INPUT(y);

#if 1 < z && z < 3

w = MIN(ABS(x) + ABS(y) + ABS(z), POW3(x));

#elif z > 3

w = MAX(2\*x+y, POW2(2\*y-z));

#endif

PRINT(w);

#endif

#ifdef TASK2

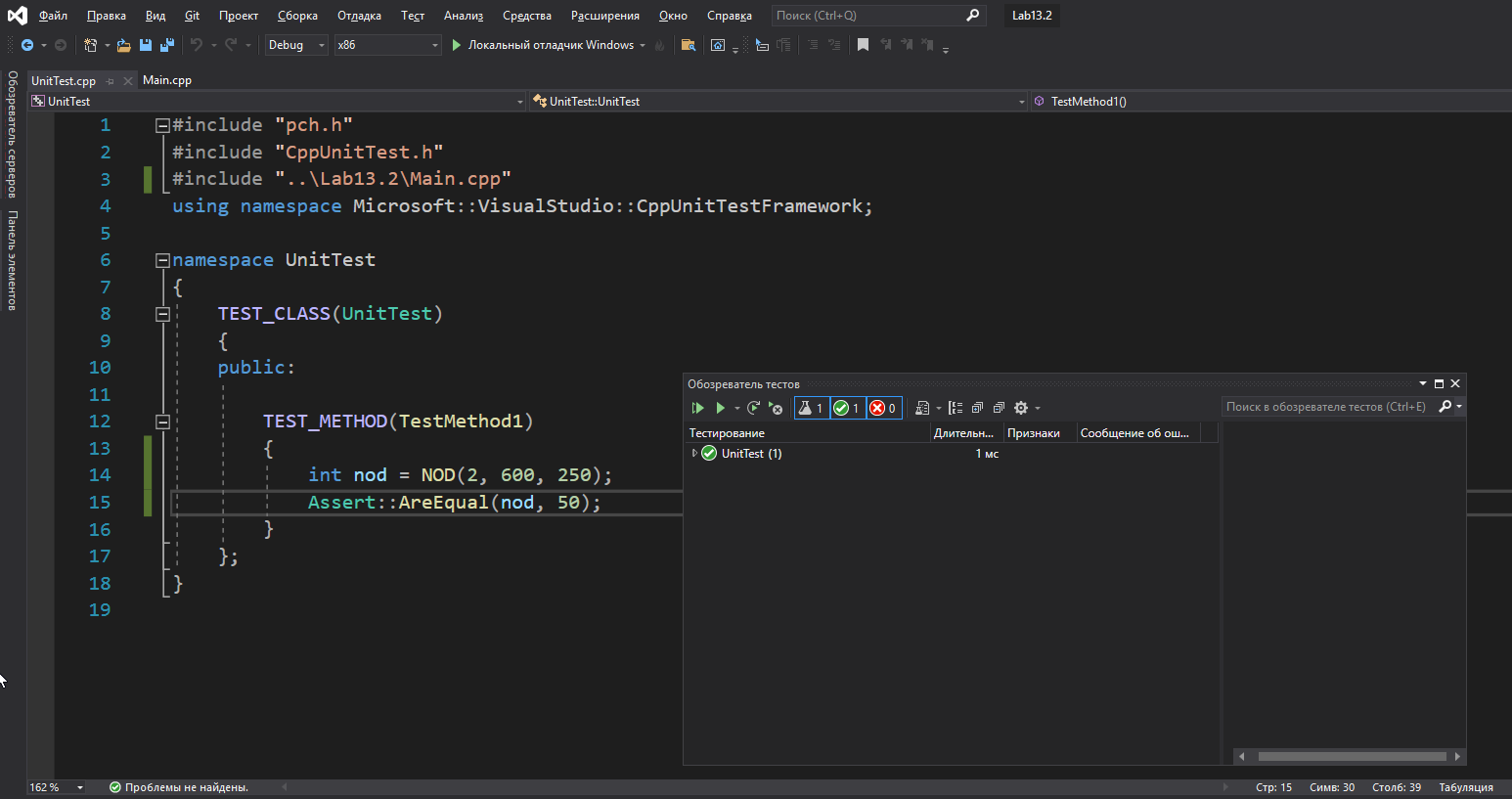
cout << NOD(3, 1024, 128, 8) << endl;

#endif

}

Git-посилання: <https://github.com/fhctyw/Lab13.2.git>

Юніт тест:



Висновок: В цій лабораторній я навчитися використовувати директиви препроцесора та застосовувати директиви препроцесора під час налагодження програми.