ЗВІТ

про виконання лабораторної роботи № < 6.1 >

*«Пошук елементів одновимірного масиву ітераційним*

*та рекурсивним способом»*

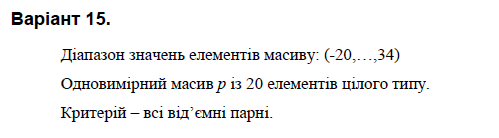
з дисципліни

«Алгоритмізація та програмування»

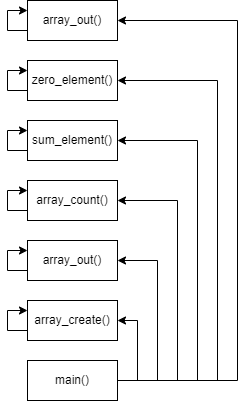
студента групи ІК-11

< *Снігура Стефана Андрійовича* >

**Умова завдання:**

****

**Структурна схема програми :**

****

**Текст програми:**

#include <iostream>

#include <iomanip>

using namespace std;

void array\_create(int\* arr, const int min, const int max, const int size, int i) {

arr[i] = min + rand() % (max - min + 1);

if (i < size - 1 )

array\_create(arr, min, max, size, i + 1);

}

void array\_out(int\* arr, const int size, int i)

{

cout << setw(4) << arr[i];

if (i < size - 1 )

array\_out(arr, size, i + 1);

}

int array\_count(int\* arr, int const size, int count, int i)

{

if (i <= size - 1) {

if (arr[i] % 2 == 0 && arr[i] < 0)

count++;

array\_count(arr, size, count, i + 1);

}

else

return count;

}

int sum\_element(int\* arr, int const size, int sum, int i)

{

if (i <= size - 1 ) {

if (arr[i] % 2 == 0 && arr[i] < 0)

sum += arr[i];

sum\_element(arr, size, sum, i + 1);

}

else

return sum;

}

void zero\_elements(int\* arr, int const size, int i)

{

if (i < size - 1) {

if (arr[i] % 2 == 0 && arr[i] < 0)

arr[i] = 0;

zero\_elements(arr, size, i + 1);

}

}

int main ()

{

srand((unsigned)time(NULL));

const int n = 20;

int p[n];

int count = 0;

const int max = 34;

const int min = -20;

array\_create(p, min, max, n, 0);

array\_out(p, n, 0);

cout << endl;

cout << endl;

array\_count(p, n, count, 0);

int num = array\_count(p, n, count, 0);

cout << "The number of negative even element = " << num << endl;;

cout << endl;

sum\_element(p, n, 0, 0);

int suma = sum\_element(p, n, 0, 0);

cout << "The sum of negative even elements = " << suma << endl;

cout << endl;

zero\_elements(p, n, 0);

array\_out(p, n, 0);

}

**UNIT-test:**

#include "pch.h"

#include "CppUnitTest.h"

#include "C:\Users\User\source\repos\lab6.1(recursion)\Source.cpp"

using namespace Microsoft::VisualStudio::CppUnitTestFramework;

namespace UnitTest1

{

TEST\_CLASS(UnitTest1)

{

public:

TEST\_METHOD(TestMethod1)

{

const int n = 20;

int arr[n] = { -1,-2,-3,-4,-5,-6,-7,-8,-9,-10,-11,-12,-13,-14,-15,-16,-17,-18,-19,-20 };

int suma = sum\_element(arr, n, 0, 0);

Assert::AreEqual(suma, -110);

}

};

}

**Посилання на git-репозиторій з проектом:**

<https://github.com/BigTrouble-Git/ashtray.git>

**Висновки**:

Виконавши цю лабораторну роботу я навчився програмувати пошук послідовним переглядом, обчислення кількості та суми заданих елементів одновимірного масиву.