# Міністерство освіти і науки України

Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"

# Факультет інформатики та обчислювальної +-\*техніки Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 7 з дисципліни

«Алгоритми та структури даних-1.

Основи алгоритмізації»

«Дослідження лінійних алгоритмів» Варіант 2

Виконав студент Боровков Іван Ігорович

(шифр, прізвище, ім'я, по батькові)

Перевірив

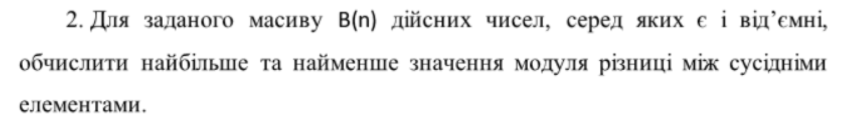
( прізвище, ім'я, по батькові)

Київ 2021

**Мета:**

Вивчити особливості обробки одновимірних масивів

**Умова задачі:**



**Постановка задачі:**

Розбити задачу на підзадачі, які реалізуємо в підпрограммах. Такими підзадачами є операції введення, виведення, формування масиву модулів різниць між сусідніми елементами та визначення найменшого та найбільшого елементу цього масиву. Для визначення найменшого та найбільшого значення використаємо стандартний алгоритм пошуку найбільшого та найменшого елемента масиву.

**Текст файла проєкту**

#include <iostream>

#include <iomanip>

#include <time.h>

using namespace std;

void input(double[], int&);

void output(double[], int, double, double);

void find\_difference\_of\_adjacent(double[], int);

double max(double[], int);

double min(double[], int);

int main()

{

setlocale(LC\_ALL, "");

double B[10], difference\_of\_advantages[9];

int size;

input(B, size);

find\_difference\_of\_adjacent(B, size);

output(B, size, max(B, size), min(B, size));

return 1;

}

void input(double B[], int& size)

{

srand(time(NULL));

//srand(5);

cout << "Введiть розмiр массиву(<=10) ";

cin >> size;

cout << "Массив:\n";

double random\_value\_1, random\_value\_2, t;

for (int i = 0; i < size; ++i)

{

random\_value\_1 = rand() % 201 - 100; //Генерація випадкового чисельника та знаменника

random\_value\_2 = rand() % 17 - 10;

while (random\_value\_2 == 0) //Перевірка знаменника на 0

{

random\_value\_2 = rand() % 17 - 10;

}

t = (int)((random\_value\_1 / random\_value\_2) \* 1000); //Округлення числа до тисячних

B[i] = t / 1000;

cout << B[i] << setprecision(3) << ' ';

}

cout << "\n";

}

void find\_difference\_of\_adjacent(double B[], int size)

{

for (int i = 0; i < size-1; ++i)

{

B[i] = abs(B[i] - B[i + 1]);

}

}

double max(double B[], int size)

{

double max = B[0];

for (int i = 1; i < size-1; ++i)

{

if (B[i] > max) max = B[i];

}

return max;

}

double min(double B[], int size)

{

double min = B[0];

for (int i = 1; i < size-1; ++i)

{

if (B[i] < min) min = B[i];

}

return min;

}

void output(double B[], int size, double max, double min)

{

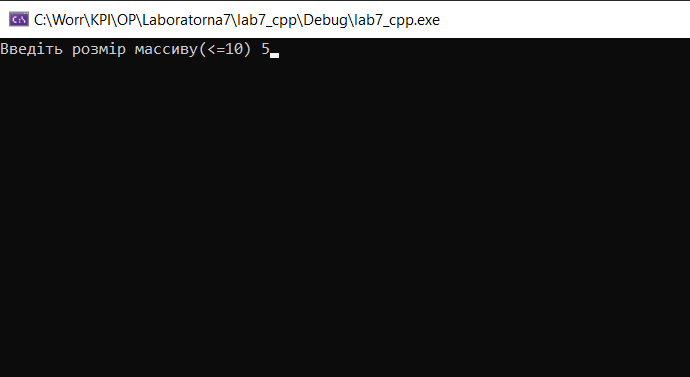
cout << "Найбiльша рiзниця двох сусiдiх чисел: " << max << '\n';

cout << "Найменша рiзниця двох сусiдiх чисел: " << min << '\n';

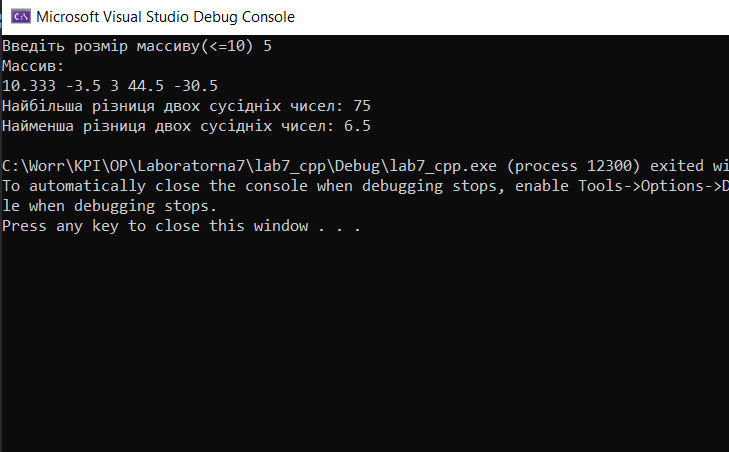
}

**Копії екранних форм результатів роботи**

Програма запитує в користувача розмір масиву, який потрібно сгенерувати...



... та виводить йому результат обчислень



**Висновок:** в ході лабораторної роботи я вивчив особливості обробки одновимірних масивів