 МІНІСТЕРСТВО ОСВІТИ I НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ   
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»

ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

**Лабораторна робота №3**

з дисципліни «Алгоритмізація та програмування

на тему: «Одновимірні масиви»

Варіант №

**Виконав:**

**Перевірив:**

Зараховано від \_\_\_.\_\_\_.\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(підпис викладача)

Київ-2022

**Завдання:**

1. Ознайомитися з теоретичними відомостями для роботи з одновимірними масивами.

2. Відповідно до свого варіанту розробити програмний застосунок, що включає реалізацію функції користувача для обробки одновимірного статичного масиву з використанням звертань до елементів через операцію індексації «[ ]»(глобальні змінні не використовувати, функція main має бути призначена тільки для виклику функцій користувача, обов’язковими є реалізація функцій користувача для введення, обробки та виведення на екран елементів масиву).

**Завдання за варіантом:** обрахувати суму елементів масиву, що розташовані між першим та другим від’ємними елементами.

3. Скласти і захистити звіт по роботі.

**Програмна реалізація:**

#include <iostream>

using namespace std;

void get\_data(int, int[]);

int tomatos\_v\_polschi(int, int[]);

int main()

{

cout << "Enter lenght of the array: ";

int lenght;

cin >> lenght;

int \*my\_array = new int[lenght];

get\_data(lenght, my\_array);

cout << "Array: ";

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

cout << my\_array[i] << " ";

int sum = tomatos\_v\_polschi(lenght, my\_array);

cout << endl << "sum = " << sum;

delete[] my\_array;

return 0;

}

void get\_data(int end, int f\_array[])

{

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

{

int j = 0;

cout << "Enter a number: ";

cin >> j;

f\_array[i] = j;

}

}

int tomatos\_v\_polschi(int lenght, int f\_array[])

{

int index1 = 0, index2 = 0, num = 0, sum = 0;

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

{

if ((f\_array[i] < 0) && (num == 0))

{

num++;

index1 = i;

}

else if ((f\_array[i] < 0) && (num == 1))

{

num++;

index2 = i;

break;

}

}

for (index1; (index1+1) < index2; index1++)

{

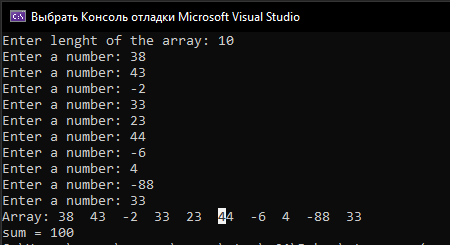
sum += f\_array[index1+1];

}

return sum;

}

**Результат виконання програми:**

****

**Контрольні питання:**

1. Що таке масив? В чому його перевага над сукупністю поодиноких змінних?

Під масивом розуміється сукупність кінцевого числа даних одного типу. Перевага над сукупністю змінних є зручність використання, менші витрати пам’яті.

1. Чи може масив містити один елемент або взагалі не містити жодного?

Масив може містити мінімум 1 елемент, оскільки розмір масива має бути більшим за 0. При створенні масива у пам’яті, виділеної для елемента зберігається «лахміття», тому не можна вважати, що масив не містить елементів.

1. Чи можуть числа «0», «1.11», «2», «2.0» бути елементами одного масиву? Поясніть.

Ні, тому що масив зберігає дані тільки одного типу, а запропоновані елементи є типами int та float.

1. Що визначає індекс елемента масиву? Які обмеження існують для його значення в C/C++?

Індекс визначає позицію елемента в масиві (пам’яті виділеної під масив). Індекс має бути додатнім цілим числом та має бути меншим за довжину масиву.

1. Які різновиди масивів існують в C/C++? Перелічіть відмінності в роботі з ними.

Існують динамічні і статичні масиви. Під час роботи з динамічними можна визначати їхню розмірність під час виконання програми. Під час роботи з статичними треба визначати їх довжину до моменту запуску програми.

1. Чим є ім'я масиву в C/C++? В чому різниця між іменами різних різновидів масивів?

Ім’я масиву – покажчик на нульовий елемент масиву.

У динамічному таким покажчиком являється змінна, якій присвоюється посилання на нульовий елемент.

У статичному це змінна, якій присвоюється вся виділена комірка пам’яті, посилання на яку співпадає з посиланням на перший елемент у цій комірці.

1. Чи може індекс при зверненні до елементу масиву бути від'ємним? Поясніть.

При використанні від’ємного індексу, виводиться «лахміття», який зберігається в пам’яті, яка розташована до масиву. Тобто використати array[-1], де array – масив, можна, але ніякого прикладного застосування цьому немає, оскільки буде виводитись «лахміття».

1. Запишіть звернення до елемента одновимірного масиву через індексацію та через зміщення.

int far[4] = { 1,2,3,4 };

cout << \*far << \*(far + 1);/\*вивід першого та другого елементу через зміщення\*/

cout << far[0] << far[1];/\*вивід першого та другого елементу через індексацію\*/

1. Які різновиди операторів «new» та «delete» є у C/C++ та в чому їх відмінність?

**new** T(/\*аргументи конструктора\*/)

**new** T[/\* довжина масиву \*/]

**delete** ptr;

**delete**[] ptr;

Тут «[]» використовується для виділення та видалення цілого масиву значень.

1. В чому особливості передачі масива у функцію?

При використанні масива, як параметра функції, у функцію передається тільки адреса масиву(першого елементу), а не вся його копія. Тому у функцію можна передавати масив, як звичайний покажчик.