Фаховий коледж ракетно-космічного машинобудування

Дніпровського національного університету імені Олеся Гончара

ЗВІТ

з лабораторних робіт

з дисципліни «Технології (Програмування)»

Спеціальність 123 Комп’ютерна інженерія

Група КС-21-1

Виконав Колосов Ігор

Перевірив М.М.Гапоненко

2021-2022

ЗМІСТ

Лабораторна робота № 14

ЛАБОРАТОРНА РОБОТА № 14

Мета: Придбати практичні навички визначення та виклику функцій з передачею параметрів за посиланням, використання масивів в якості параметрів функцій.

Хід роботи

1 Постановка задачі.

Загальна постановка завдання:

Обробити згідно завдання Вашого варіанту псевдодинамічний масив даних, використовуючи функції, які приймають масив я якості параметру. Програма повинна задовольняти наступним вимогам:

* програма повинна містити наступні функції:

1. введення значень елементів масиву;
2. функції, що відповідають пунктам а) та б) завдання;
3. виведення значень елементів масиву, яка буде викликатись з функції, що відповідає пункту б) завдання;

* організувати меню для вибору користувачем варіанту виконання вказаних вище функцій. Забезпечити повернення в меню програми після виконання чергового пункту меню. Вихід із програми здійснювати по відповідному пункту меню: «Вихід з програми»;
* при обробці масиву враховувати, що шукані елементи можуть бути відсутні. В цьому випадку вивести користувачеві відповідне повідомлення;
* введення та виведення вхідних та вихідних даних повинно містити необхідні для користувача повідомлення.

У одномірному масиві, який складається з n цілих елементів, обчислити:

а) добуток елементів масиву, розташованих між передостаннім та останнім нульовими елементами;

б) перетворити масив таким чином, щоб у першій його половині розташовувались парні елементи, а у другій половині – непарні елементи.

2 Блок-схема алгоритму рішення задачі.

3 Текст програми мовою програмування С++.

#include <stdio.h>

#include <iostream>

#include <windows.h>

*void* transformArray(*int array[], int ARRSIZE*) {

*/\*  Пояснення:*

*У першому циклі ми рахуємо кількість парних елементів.*

*i = 0 Лічильник для переміщення по масиву*

*j = 0 Лічильник для вставки парних елементів з початку масиву (кожна вставка j++)*

*k > 0 Цикл працює поки лічильник парних елементів не дорівнюватиме 0*

*i++ Крок лічильника для переміщення \*/*

*// перетворити масив таким чином, щоб у першій його половині розташовувались парні елементи,*

*// а у другій половині – непарні елементи.*

*// transformArray(int array[], int ARRSIZE)*

*int* temp, k = 0;

    for (*int* i = 0; i < *ARRSIZE*; i++) {

        if (*array*[i] % 2 == 0) *// для перетворення*

            k++;

    }

    for (*int* i = 0, j = 0; k > 0; i++) {

        if (*array*[i] % 2 == 0) {

*// Заміна знайденного парного елементу на непарний*

            temp = *array*[i];

*array*[i] = *array*[j];

*array*[j] = temp;

            k--;

            j++;

        }

    }

}

*// int calculateArray(int array[], int ARRSIZE)*

*int* calculateArray(*int array[], int ARRSIZE*) {

*// добуток елементів масиву, розташованих між передостаннім та останнім нульовими елементами*

*/\* Пояснення:*

*у нас є змінна перемикач - flag,*

*якщо зустрівся 0 то ми починаємо рахувати добуток,*

*якщо зустрівся другий 0, то ми закінчуємо рахувати добуток.*

*кожного разу як ми зустрічаємо 0 вперше (коли flag == 0) ми рахуємо добуток наново (див. завд.) \*/*

*int* product = 1, flag = 0;

    for (*int* i = 0; i < *ARRSIZE*; i++) {

        if (*array*[i] == 0 && flag == 1) {

            flag = 0;

        }

        else if (*array*[i] == 0 && flag == 0) {

            product = 1;

            flag = 1;

        }

        else if (flag == 1) {

            product \*= *array*[i];

        }

    }

    return product;

}

*// void arrayPrint(int array[], int ARRSIZE)*

*void* arrayPrint(*int array[], int ARRSIZE*) {

    printf("\n[");

    for (*int* i = 0; i < *ARRSIZE*-1; i++) {

        printf("%d, ", *array*[i]);

    }

    printf("%d]\n", *array*[*ARRSIZE*-1]);

}

*int* main()

{

    SetConsoleCP(65001);

    SetConsoleOutputCP(65001);

*const* *int* ARRSIZE = 6;

*int* array[ARRSIZE];

*int* product, option;

    printf("Введіть масив\n");

    for (*int* i = 0; i < ARRSIZE; i++) {

        printf("array[%d] = ", i);

        scanf("%d", &array[i]);

    }

    arrayPrint(array, ARRSIZE);

    printf("1: трансформувати масив\n2: порахувати массив\n Введіть відповіть: ");

    scanf("%d", &option);

    switch (option) {

    case 1:

        transformArray(array, ARRSIZE);

        arrayPrint(array, ARRSIZE);

        break;

    case 2:

        product = calculateArray(array, ARRSIZE);

*// Виведення добутку*

        if (product != 1)

            printf("Добуток: %d", product);

        else

            printf("Не було введено жодного нуля (0 . . . 0 рахуються елементи між нулями)\n");

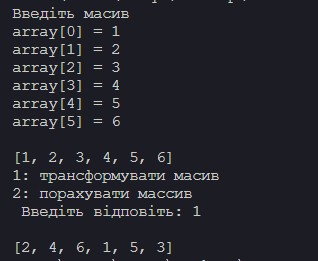
        break;

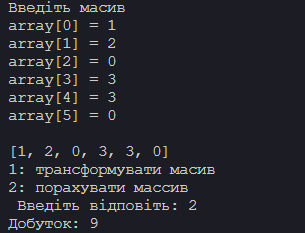
    }

    return 0;

}

4 Копія вікна виконання програми





5 Висновок

В результаті виконання лабораторної роботи опрацьовано теоретичний матеріал за темами:

* масиви