**Технически университет – гр. Габрово**  
**Факултет „Електротехника и електроника“**  
**Катедра „Компютърни системи и технологии“**

**Дисциплина:**

**„Въведение в програмирането“**

### ****КУРСОВА РАБОТА****

**Тема:** Програма за намиране на общите елементи между два масива и откриване на най-малкия общ елемент.

**Разработил:***Име*: Даниел Тихомиров Трифонов  
*Специалност*: СКИ, курс: 1-ви, група: 1б  
*Факултетен номер*: 22472126

**Проверил:……………………**

**\***ас***.** Виктория Велкова**\**

### ****1. Теоретична част****

#### **1.1 Дефиниране и използване на структури масиви в програмен език C**

Масивът в програмен език C е структура от данни, която позволява съхраняването на последователни елементи от един и същ тип. Елементите на масивите се достъпват чрез индекси, започващи от 0.

**Основни характеристики:**

* **Статичен размер** – размерът на масива се задава предварително и не може да се променя по време на изпълнение.
* **Едномерни и многомерни масиви** – едномерните се използват за линейно съхранение на данни, докато двумерните и тримерните са подходящи за таблични структури.
* **Памет** – елементите се съхраняват в последователни адреси в паметта, което осигурява бърз достъп.

В програмата се използват **едномерни масиви** за съхранение на цели числа. Масивите са основният инструмент за реализиране на алгоритмите за търсене на общи елементи и намиране на най-малкия елемент.

#### **1.2 Основна идея на програмата**

Програмата използва три основни масива:

* **Първи масив** – съхранява първоначалния набор от числа.
* **Втори масив** – съхранява втория набор от числа.
* **Трети масив** – съдържа общите числа от първите два масива.

С помощта на няколко функции се реализира проверка за общи елементи и намиране на **най-малката стойност** сред тях.

### ****2. Практическо изпълнение****

#### **2.1 Функции в програмата**

1. **Array1** – събира елементи в първия масив.
2. **Array2** – събира елементи във втория масив.
3. **findCommonElements** – открива общите числа между двата масива и намира най-малкия елемент.
4. **printArray** – отпечатва съдържанието на масив.

#### **2.2 Програмен код**

#include <stdio.h>

#include <stdlib.h>

#include <stdbool.h>

// Функция за инициализиране на първия масив

void Array1(int array1[], int array1Length)

{

int array1Input;

for (int i = 0; i < array1Length; i++)

{

// Прочитаме стойностите на първия масив

printf("Please enter the number for array1[%d]: ", i);

scanf\_s("%d", &array1Input);

array1[i] = array1Input;

}

}

// Функция за инициализация на втория масив

void Array2(int array2[], int array2Length)

{

int array2Input;

for (int j = 0; j < array2Length; j++)

{

// Прочитаме стойностите на втория масив

printf("Please enter the number for array2[%d]: ", j);

scanf\_s("%d", &array2Input);

array2[j] = array2Input;

}

}

// Функция за проверка дали числата от първия масив се съдържат във втория масив и запълване на третия масив

void findCommonElements(int array1[], int array2[], int array3[], int array1Length, int array2Length, int\* index3, int\* smallestNumber, int\* smallestIndex, int\* array3Length, bool\* ifWeHaveCommons)

{

for (int i = 0; i < array1Length; i++)

{

for (int j = 0; j < array2Length; j++)

{

// Проверка дали елементите от 1вия масив са равни на елементите на 2рия масив

if (array1[i] == array2[j])

{

array3[\*index3] = array1[i];

\*ifWeHaveCommons = true;

if (array3[\*index3] < \*smallestNumber)

{

\*smallestNumber = array3[\*index3];

\*smallestIndex = \*index3;

}

(\*index3)++;

(\*array3Length)++;

}

}

}

}

void printArray(int array[], int length)

{

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

{

printf("%d ", array[i]);

}

printf("\n");

}

int main()

{

int smallestNumber = INT\_MAX;

int index3 = 0;

int smallestIndex = index3;

int array1Input;

int array2Input;

int array3[299];

int array3Length = 0;

bool ifWeHaveCommons = false;

int array1[299];

int array1Length;

printf("Please enter the first array's length (0-299): ");

scanf\_s("%d", &array1Length);

while (array1Length > 299 || array1Length < 0)

{

printf("Incorrect length, please enter a number between (0-299): ");

scanf\_s("%d", &array1Length);

}

int array2[299];

int array2Length;

printf("Please enter the second array's length (0-299): ");

scanf\_s("%d", &array2Length);

while (array2Length > 299 || array2Length < 0)

{

printf("Incorrect length, please enter a number between (0-299): ");

scanf\_s("%d", &array2Length);

}

// Въвеждаме стойностите на първия масив

Array1(array1, array1Length);

// Въвеждаме стойностите на втория масив

Array2(array2, array2Length);

printf("1st Array\n");

printArray(array1, array1Length);

printf("2nd Array\n");

printArray(array2, array2Length);

// Проверяваме дали има общи елементи и попълваме третия масив

findCommonElements(array1, array2, array3, array1Length, array2Length, &index3, &smallestNumber, &smallestIndex, &array3Length, &ifWeHaveCommons);

if (ifWeHaveCommons)

{

printf("3rd Array\n");

printArray(array3, array3Length);

printf("The smallest number in array1 and array2 is: %d, with index of array3[%d]\n", smallestNumber, smallestIndex);

}

else

{

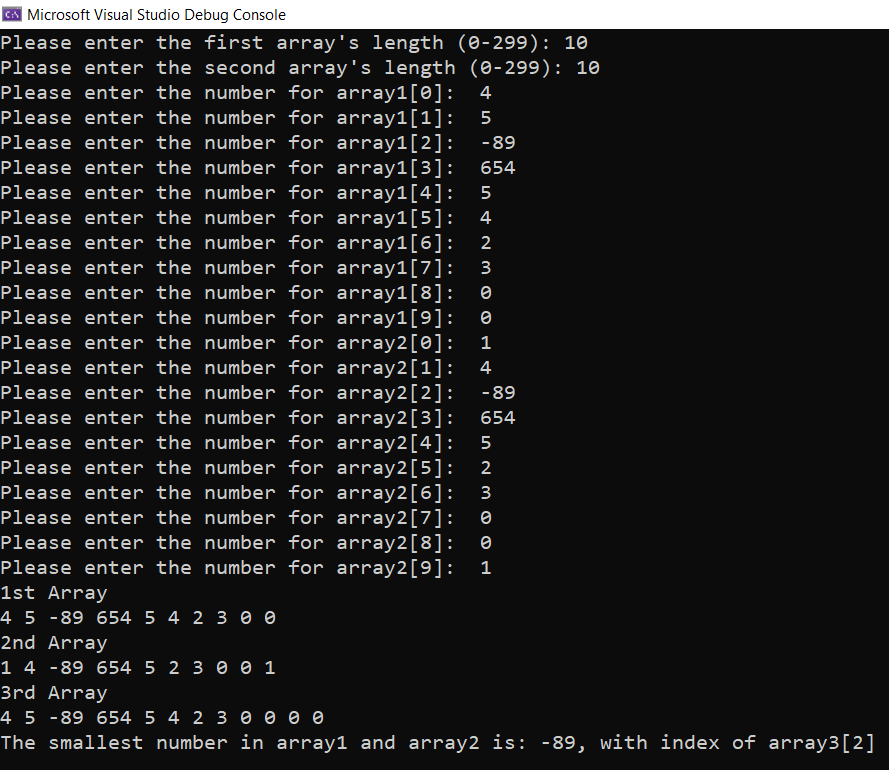
printf("We don't have common numbers!\n");

}

return 0;

}

#### **2.2 Екрани**



### ****3. Обем на чертежите****

**3.1 Блок-схема**

### ****4. Използвана литература****

* „C Programming Documentation“, ISO/IEC 9899:2018, Official C Standard.