**Министерство образования и наук РФ**

**Федеральное государственное образовательное учреждение**

**Высшего образования**

**Рязанский государственный радиотехнический университет**

**Кафедра ЭВМ**

Отчет по лабораторной работе №5:

“Одномерные массивы и указатели”

Вариант 12

Выполнила: ст. гр 640

Тограева К. О.

Проверила:

Доц. к.т.н. Елесина С.И.

Рязань 2017

**Цель работы:**  
 получение навыков обработки одномерных статических динамических

массивов. Использование указателей при работе с массивами.

**Вариант №12:**В одномерном массиве, состоящем из n целых элементов, вычислить:

1) номер максимального элемента масcива;  
2) произведение элементов массива, расположенных между первым и вторым нулевыми элементами.  
Преобразовать массив таким образом, чтобы в первой его половине  
располагались элементы, стоявшие в нечетных позициях, а во второй половине — элементы, стоявшие в четных позициях.

**Программирование**

**#include <stdio.h>  
#include <stdlib.h>  
#include <time.h>  
#include <stdio.h>  
#include <stdint.h>  
  
uint32\_t n;  
  
void initializeArray(int \*array, uint32\_t);  
  
const uint32\_t c = 10;  
//Вариант 12  
//В одномерном массиве, состоящем из n целых элементов, вычислить:  
//1) номер максимального элемента масcива;  
//2) произведение элементов массива, расположенных между первым и вторым нулевыми элементами.  
//Преобразовать массив таким образом, чтобы в первой его половине  
//располагались элементы, стоявшие в нечетных позициях, а во второй  
// половине — элементы, стоявшие в четных позициях.  
  
int getMaxArrayNumber(int \*arPointer) {  
 int maxIndex = 0;  
 for (int i = 0; i < n; ++i) {  
 if (arPointer[i] > maxIndex) {  
 maxIndex = i;  
 }  
 }  
 printf("Максимальное число %i\n ", maxIndex);  
 return maxIndex;  
}  
  
int \*chetNechet(int \*pInt) {  
 int newArrPointer = 0;  
 int array[c];  
 for (int i = 0; i < c; ++i) {  
 if (i % 2) {// нечётные  
 array[newArrPointer] = pInt[i];  
 newArrPointer++;  
 }  
 }  
 for (int i = 0; i < c; ++i) {  
 if (!(i % 2)) {// чётные  
 array[newArrPointer] = pInt[i];  
 newArrPointer++;  
 }  
 }  
 return array;  
}  
  
int getMultiplyBetweenZeros(const int \*pInt) {  
 int zeroNumber = 0;  
 int firstZeroPointer = -1;  
 for (int i = 0; i < c; ++i) {  
 if (pInt[i] == 0) {  
 zeroNumber++;  
 if (firstZeroPointer == -1) {  
 firstZeroPointer = i;  
 }  
 }  
 }  
  
 if (zeroNumber != 2) {  
 printf("Неверное число нулей");  
 return 1;  
 }  
  
 int mult = 1;  
 int lIndex = firstZeroPointer + 1;  
 int fIndex = firstZeroPointer;  
 firstZeroPointer += 1;// сл. элемент  
 while (pInt[firstZeroPointer] != 0) {// обеспечивается безопасность, т.к. два нуля уже гаранитрованно есть  
 mult \*= pInt[firstZeroPointer];  
 lIndex = firstZeroPointer;  
 firstZeroPointer++;  
 }  
 printf("Число между двумя нулями это %i\n", (lIndex - fIndex != 1) ? mult : 0);  
 return mult;  
}  
  
  
void printArray(int \*array) {  
 for (int i = 0; i < c; ++i) {  
 printf("%i ", array[i]);  
 }  
 printf("\n");  
}  
  
int main() {  
#ifdef WIN32  
 system("chcp 65001");  
#endif  
 srand(time(NULL));  
 printf("Введите число элементов массива\n");  
 scanf("%u", &n);  
 int \*arr = calloc(n, sizeof(int));  
 if (arr == 0) {  
 return -1;  
 }  
 int array[c];  
 // initialization  
 // заполняем случайными числами с 1 до 10  
 initializeArray(array, c);  
 initializeArray(arr, n);  
  
 getMaxArrayNumber(arr);  
 getMultiplyBetweenZeros(array);  
  
 printArray(chetNechet(array));  
 return 0;  
}  
  
void initializeArray(int \*array, uint32\_t c) {  
 for (int i = 0; i < c; ++i) {  
 array[i] = rand() % 7;  
 }  
 printArray(array);  
}**

Тестовый пример:

Пример

Вывод:

Мы научились работать со статическими и нестатическими массивами, а также со ссылками и указателями