Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

Лабораторная работа № 3

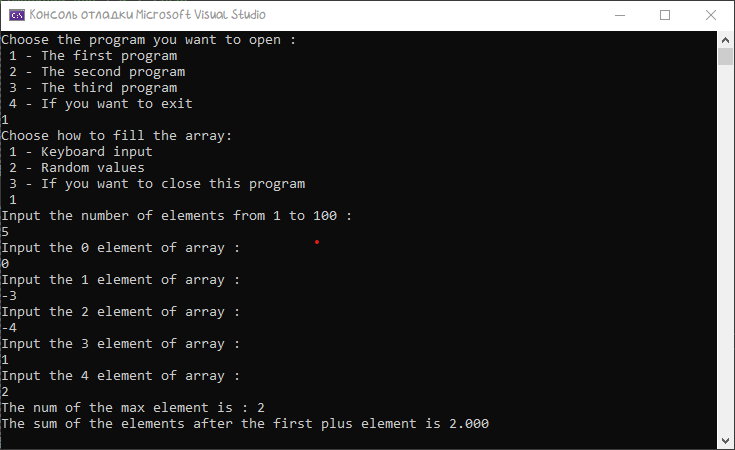
«Одномерные массивы»

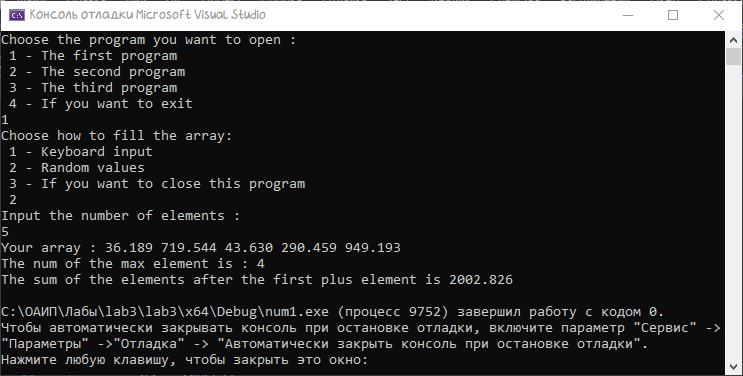
Проверил: Выполнил:   
Богдан Е.В. Зорин А.Ю.

МИНСК 2022



6.(1)В одномерном массиве, состоящем из n вещественных элементов, вычислить:  
- номер максимального по модулю элемента массива;  
- сумму элементов массива, расположенных после первого положительного элемента.







Код:

int task1(){

float arr[100], abs[100];

float max=0, sum = 0;

int choose = 0, n = 0, num = 0;

printf("Choose how to fill the array:\n 1 - Keyboard input\n 2 - Random values\n 3 - If you want to close this program \n ");

input\_choose(&choose);

if (choose == 1) keyboard\_input(&n, arr);

if (choose == 2) random\_input(&n, arr);

if (choose == 3) main();

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

abs[i] = arr[i];

//Модуль элементов

for (int i = 0; i < n; i++) {

if (abs[i] < 0) {

abs[i] = 0 - abs[i];

}

}

max = abs[0];

//Поиск максимального элемента

for (int i = 0; i < n; i++) {

if (abs[i] > max) {

max = abs[i];

num = i;

}

}

//Сумма элементов после первого положительного

for (int i = 0; i < n; i++) {

if (arr[i] > 0) {

for (int j = i + 1; j < n; j++) {

sum += arr[j];

}

break;

}

}

printf("The num of the max element is : %d\n", num);

printf("The sum of the elements after the first plus element is %.3f\n", sum);

return 0;

}

6.(2)В массиве из n элементов удалить каждый k-ый элемент



Код:

int task2() {

float arr[100];

int n = 0, fl = 0;

int k; int choose = 0;

printf("Choose how to fill the array:\n 1 - Keyboard input\n 2 - Random values\n 3 - If you want to close this program \n ");

input\_choose(&choose);

if (choose == 1) keyboard\_input(&n, arr);

if (choose == 2) random\_input(&n, arr);

if (choose == 3) main();

inputint(&k,&fl);

//Удалние каждого k-го элемента массива

for (int i = k - 1; i < n;) {

if (fl == 1) break;

for (int j = i; j < n - 1; j++)

arr[j] = arr[j + 1];

n--;

i = i + (k - 1);

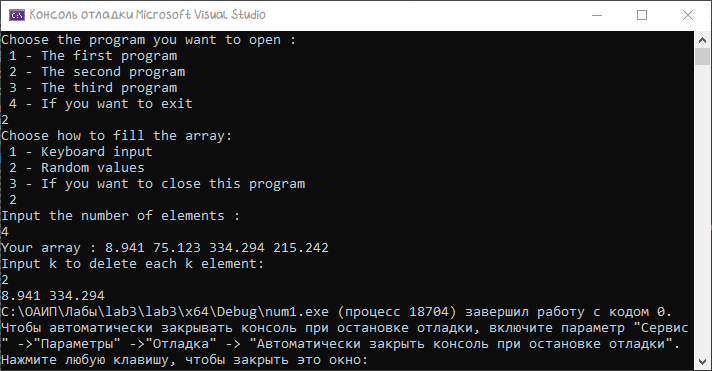
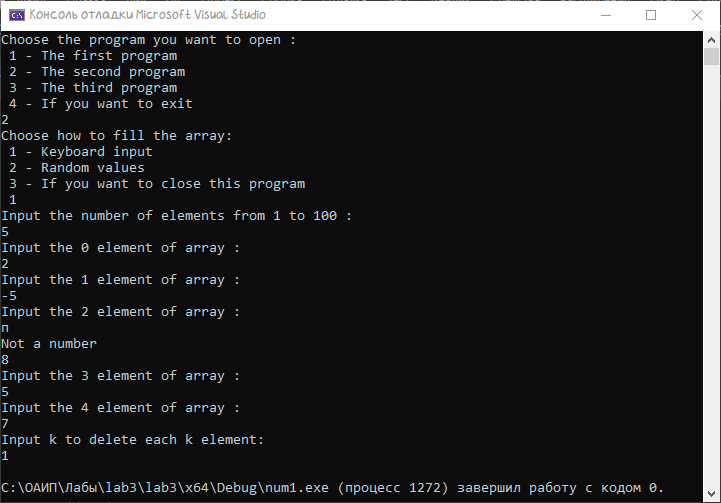
}

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

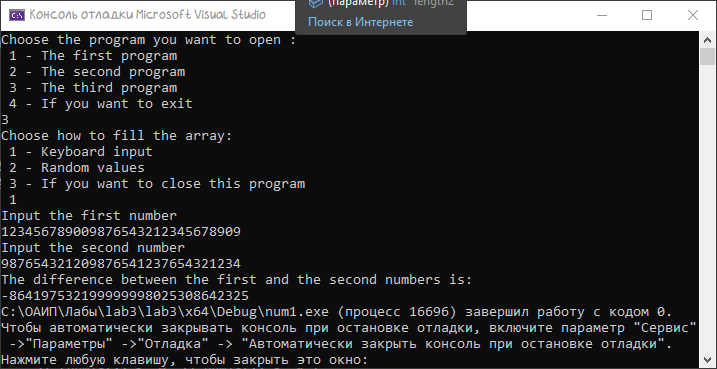
printf("%.3f " , arr[i]);

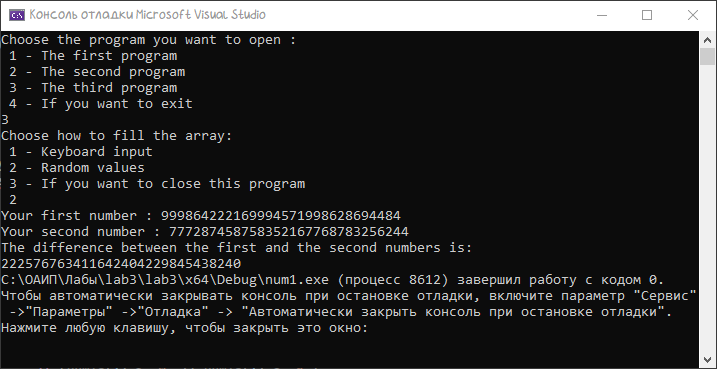
return 0;

}



6.(3) Для арифметических операций с большими числами, которые не могут быть представлены в памяти компьютера, используется следующий прием. Каждая цифра таких чисел записывается в отдельный элемент массива, и необходимые операции проводятся с элементами массива цифр. Составить программу, выполняющую вычитание 30-значных чисел.









Код:

int task3() {

int length1 = 0,length2=0,length=0,choose=0,fl=0;

int transfer = 0, nonzero = 0, col = 0;;

char num1s[100],num2s[100];

int num1[100], num2[100],result[100],temp[100];

int check0=0;

printf("Choose how to fill the array:\n 1 - Keyboard input\n 2 - Random values\n 3 - If you want to close this program \n ");

input\_choose(&choose);

if (choose == 1) keyboard\_input3(num1s, num2s, num1, num2, &length, &length1, &length2);

if (choose == 2) random\_input3(num1, num2,&length);

if (choose == 3) main();

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

if (num1[i] == num2[i]) { //Сравнение чисел исходя из элементов

col++;

continue;

}

if (num1[i] < num2[i]) { //Сравнение чисел и замена местами

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

temp[i] = num1[i];

num1[i] = num2[i];

num2[i] = temp[i];

}

fl = 1;

}

break;

}

for (int j = length; j > 0; j--) {

if (num1[j - 1] < num2[j - 1]) {

transfer++;

while (num1[j - 1 - transfer] == 0)

transfer++;

num1[j - 1 - transfer]--;

transfer--;

if (transfer == 0) {

num1[j - 1] += 10;

}

else {

while (transfer != 0) {

num1[j - 1 - transfer] += 9;

transfer--;

}

num1[j - 1] += 10;}

}

result[j - 1] = num1[j - 1] - num2[j - 1];

}

if (col == length ) {

printf("The difference between the first and the second numbers is 0");

return 0;

}

// Добавление минуса перед числом

if (fl == 1) {

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

if (result[i] != 0) {

result[i] = 0-result[i];

break;}

}

// Поиск первого ненулевого элемента

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

if (result[i] != 0) {

nonzero = result[i];

break;}

}

//Удаление всех нулей до первого ненулевого элемента

for (int i = 0; result[i] != nonzero; i++) {

for (int j = i; j < length - 1; j++)

result[j] = result[j + 1];

length--; i--;

}

printf("The difference between the first and the second numbers is:\n");

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

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

return 0;

}









