Покажчики та масиви

1. Мета роботи

Метою лабораторної роботи є отримання практичних навичок в роботі з покажчиками і з адресною арифметикою в мові C.

2. Тими для попереднього опрацювання

Покажчики. Типізовані покажчики.

Покажчики та масиви.

Адресна арифметика.

Динамічне виділення пам'яті.

3. Завдання для виконання

Виконати завдання лабораторної роботи ╧8 з такими додатковими умовами:

* розмір масиву визначається на початку виконання програми як випадкове число в діапазоні 50 - 200;
* в тексті програми забороняється застосовувати операцію індексації.

|  |  |  |  |
| --- | --- | --- | --- |
| **╧ варианта** | **Размерность массива** | **Диапазон значений** | **Что нужно сделать** |
| **19** | 200 | -50 - 50 | Подсчитать количество всех, непрерывных последовательностей положительных чисел, длина которых больше 7 |

Код:

// LAB10.cpp : This file contains the 'main' function. Program execution begins and ends there.

//

#include <iostream>

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <malloc.h>

using namespace std;

int main(void) {

setlocale(LC\_ALL, "Rus");

int size; /\* размер массива \*/

int\* Ar; /\* указатель на начало массива \*/

srand(time(NULL));

size = rand() % 201 + 50;

printf("size=%d\n", size);

/\* выделения памяти \*/

Ar = (int \*)malloc(size \* sizeof(int));

int i, j; /\* индексы в массиве \*/

int nn; /\* количество эл-тов в последовательности \*/

int ib; /\* индекс начала последовательности \*/

int copy\_chek;

/\* заполнение массива случайными числами \*/

for (i = 0; i < size; Ar[i++] = rand() % 101 - 50);

/\* вывод начального массива \*/

printf("Начальный массив:\n");

for (i = 0; i < size; printf("%3d ", Ar[i++]));

putchar('\n');

putchar('\n');

for (nn = i = 0; i < size; i++) { /\* перебор массива \*/

if (Ar[i] >= 0) {//проверка на положительные числа

ib = i;

for (int chek = ib, copy\_chek = chek; Ar[chek] >= 0; chek++)

{

if (chek >= (copy\_chek + 6)) {//проверка для нахождения ровно семи подряд элементов

for (; Ar[chek] >= 0; chek++);//находим все элементы которые положительные и больше этой поседовотельности

nn++;//найденна последовательность

i = chek;//ставим точку поиска в массиве

break;

}

}

}

}

printf("результат:%d\n",nn);

putchar('\n');

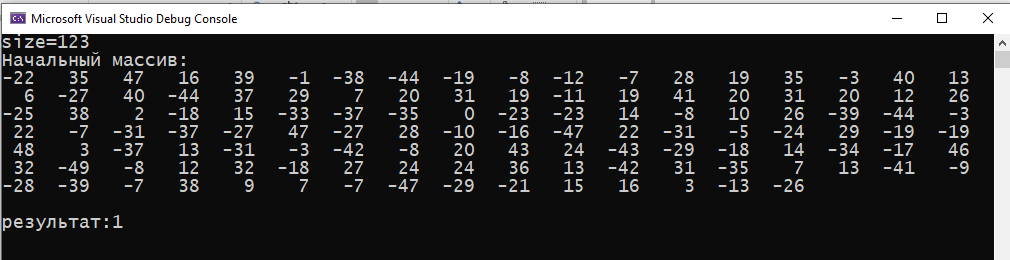
/\* освобождение памяти \*/

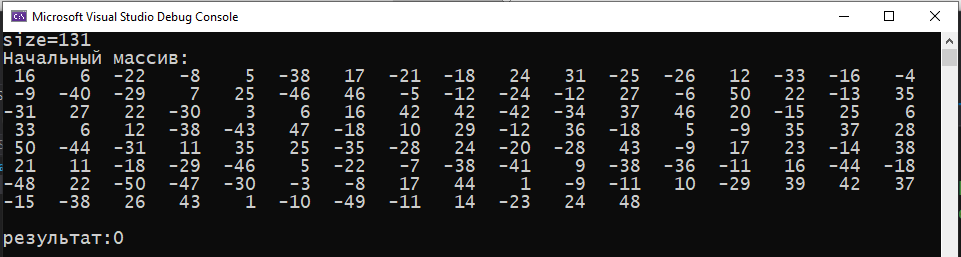
free(Ar);

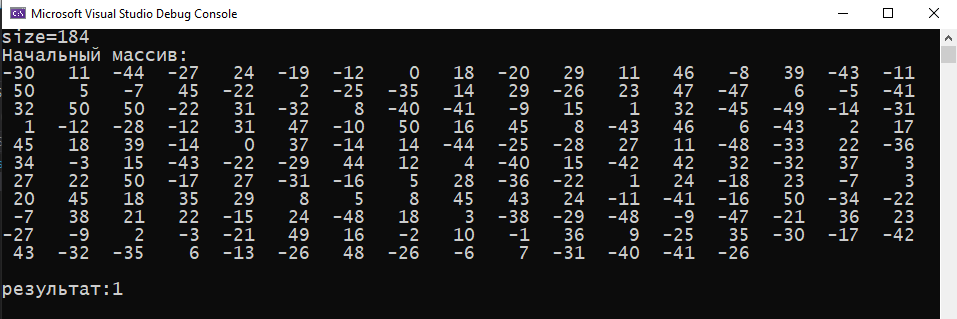
return 0;

}

Результати роботи програми:







Висновки:

При виконанні лабораторної роботи вивчені питання:

* робота з масивами через покажчики;
* динамічне виділення і звільнення пам'яті.