**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ НОВОСИБИРСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Лабораторная работа №3

по дисциплине «Программирование»

Массивы указателей

Группа: АВТ-041

Студенты: Антонов А.

Преподаватель: Балакин В.

НОВОСИБИРСК 2021

**Задача**

Функция получает линейный массив целых, находит в нем последовательности подряд возрастающих значений и возвращает их в динамическом массиве указателей на линейные массивы (аналог двумерного массива). В каждом из линейных динамических массивов содержится копия возрастающей последовательности, начиная с индекса 1, а под индексом 0 содержится его длина. Невозрастающие значения включаются в отдельный массив, добавляемый в конец (или начало) массива указателей.

**Проектирование программы**

Идея: Создание функции сравнения рядом стоящих значений, в которой учитывается выделение и изменение памяти для элементов.

Составные части программы:

1. Ввод линейного массива:

for (i = 0; i < size; i++)

{

printf("arr[%d]=", i);

scanf\_s("%d", &mass[i]); // вводим элементы массива

}

1. Условие, если элементов в массиве 0 или 1:

if (i >= size) goto z2;

1. Сравнение соседних элементов, если последовательность убывает, то текущее значение возвращает в начало или конец массива указателей отдельного массива:

if (mass[i] < mass[i - 1]) // сравнение соседних элементов

{

nevozr = (int\*)realloc(nevozr, (++nevozr[0]) \* sizeof(int)); // изменение величины выделенной памяти

nevozr[nevozr[0] - 1] = mass[i]; // включение невозрастающих значений в отдельный массив в начало и конец массива указателей

}

1. Если последовательность возрастает, возвращение её значений и разделение по линейным массивам (1 массив, пока последовательность возрастает):

else if (++i < size) goto z; // иначе, если числа возрастают

if (i - 1 > a) // возврат значений на линейные массивы

{

ressize++;

result = (int\*\*)realloc(result, ressize \* sizeof(int\*));

result[ressize - 1] = (int\*)malloc((i - a + 1) \* sizeof(int));

result[ressize - 1][0] = i - a + 1;

for (j = i - a; j >= 1; j--) result[ressize - 1][j] = mass[a + j - 1];

}

1. Для продолжения сравнения соседних элементов:

a = i;

if (i + 1 < size) {

++i;

goto z;

}

1. Если элементов в массиве 0 или 1:

z2:;

result[0] = nevozr;

1. Вывод полученных массивов на консоль:

for (i = 0; i < ressize; ++i) // вывод полученных массивов на консоль

{

printf("\n%i | ", result[i][0]);

for (j = 1; j < result[i][0]; ++j)

{

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

}

}

Переменные:

size, i, j, a, ressize - переменные-счетчик и переменные-признак

mass, nevozr, result - массивы

**Текст программы с комментариями**

#include <stdio.h>

#include <stdlib.h>

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main(void)

{

int size, i, j, a, ressize = 1;

int\* mass;

int\* nevozr;

int\*\* result;

printf("Vvedite razmernost' massiva ");

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

mass = (int\*)malloc(size \* sizeof(int)); // выделенине доп. памяти

for (i = 0; i < size; i++)

{

printf("arr[%d]=", i);

scanf\_s("%d", &mass[i]); // вводим элементы массива

}

result = (int\*\*)malloc(sizeof(int\*));

nevozr = (int\*)malloc(sizeof(int));

nevozr[0] = 1; //

a = 0;

i = 1;

if (i >= size) goto z2; // если элементов массива <=1

z:;

if (mass[i] < mass[i - 1]) // сравнение соседних элементов

{

nevozr = (int\*)realloc(nevozr, (++nevozr[0]) \* sizeof(int)); // изменение величины выделенной памяти

nevozr[nevozr[0] - 1] = mass[i]; // включение невозрастающих значений в отдельный массив в начало и конец массива указателей

}

else if (++i < size) goto z; // иначе, если числа возрастают

if (i - 1 > a) // возврат значений на линейные массивы

{

ressize++;

result = (int\*\*)realloc(result, ressize \* sizeof(int\*));

result[ressize - 1] = (int\*)malloc((i - a + 1) \* sizeof(int));

result[ressize - 1][0] = i - a + 1;

for (j = i - a; j >= 1; j--) result[ressize - 1][j] = mass[a + j - 1];

}

a = i;

if (i + 1 < size) {

++i;

goto z;

}

z2:;

result[0] = nevozr; // если элементов в массиве 0 или 1 то выведем на консоль этот самый элемент

for (i = 0; i < ressize; ++i) // вывод полученных массивов на консоль

{

printf("\n%i | ", result[i][0]);

for (j = 1; j < result[i][0]; ++j)

{

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

}

}

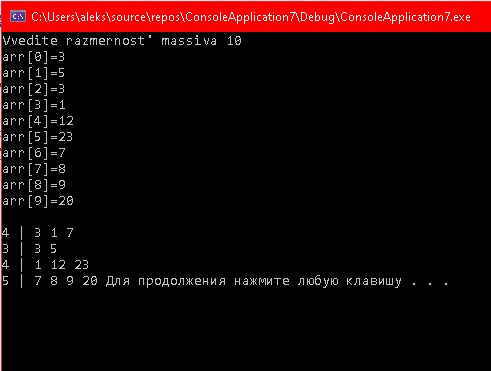
free(mass);

system("pause");

return 0;

}

**Пример работы программы**



**Вывод**  
Функция была реализована, программа работает корректно.