|  |
| --- |
| МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  федеральное государственное АВТОНОМНОЕ образовательное учреждение высшего образования  «Национальный исследовательский ядерный университет «МИФИ» |
| **Обнинский институт атомной энергетики –**  филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский ядерный университет «МИФИ»  **(ИАТЭ НИЯУ МИФИ)** |

Отделение ядерной физики и технологий

**Лабораторная работа 6**

«Указатели и динамические массивы. Использование указателей в качестве аргументов функций»

Выполнил:

студент гр. ЯРМ-С24 Долинов Д.А.

Проверил:

Неведин А.В.

**Обнинск, 2024 г.**

Цель: Знакомство с указателями в С++. Изучение одномерных динамических массивов в языке С++. Знакомство с функциями, возвращающими несколько значений. Рассмотрение на примерах, как указатели используются в качестве параметров функций.

Задание: В массиве Х(N) определить количество групп элементов, представляющих из себя знакочередующийся ряд. Если такие группы есть, то удалить из массива последнюю такую группу. В программе написать функции: удаления элемента, поиска последней группы знакочередующихся элементов.

Программа:

#include <iostream>

using namespace std;

void print\_arr(int array[], size\_t size)

{

while (size--)

{

cout<<" "<<\*array++;

}

cout << endl;

}

int is\_seq(const int\* x, const int\* y)

{

return ((\*x) \* (\*y) < 0);

}

int last\_seq(int array[], size\_t size, int\*\* begin, int\*\* end)

{

int\* cur = array + size - 1;

for (; (array < cur) && !is\_seq(cur, cur - 1); --cur) { ; }

\*end = cur;

for (; (array < cur) && is\_seq(cur, cur - 1); --cur) { ; }

\*begin = cur;

return \*end - \*begin;

}

size\_t rmlastseq(int array[], size\_t size)

{

int\* begin;

int\* end;

int len = 0;

if ((len = last\_seq(array, size, &begin, &end)) > 0)

{

end++;

len++;

array += size;

while (end < array)

{

\*begin++ = \*end++;

}

}

return size - len;

}

int main()

{

int arrsize;

cin >> arrsize;

int arr = new int[arrsize];

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

{

cin >> arr[i];

}

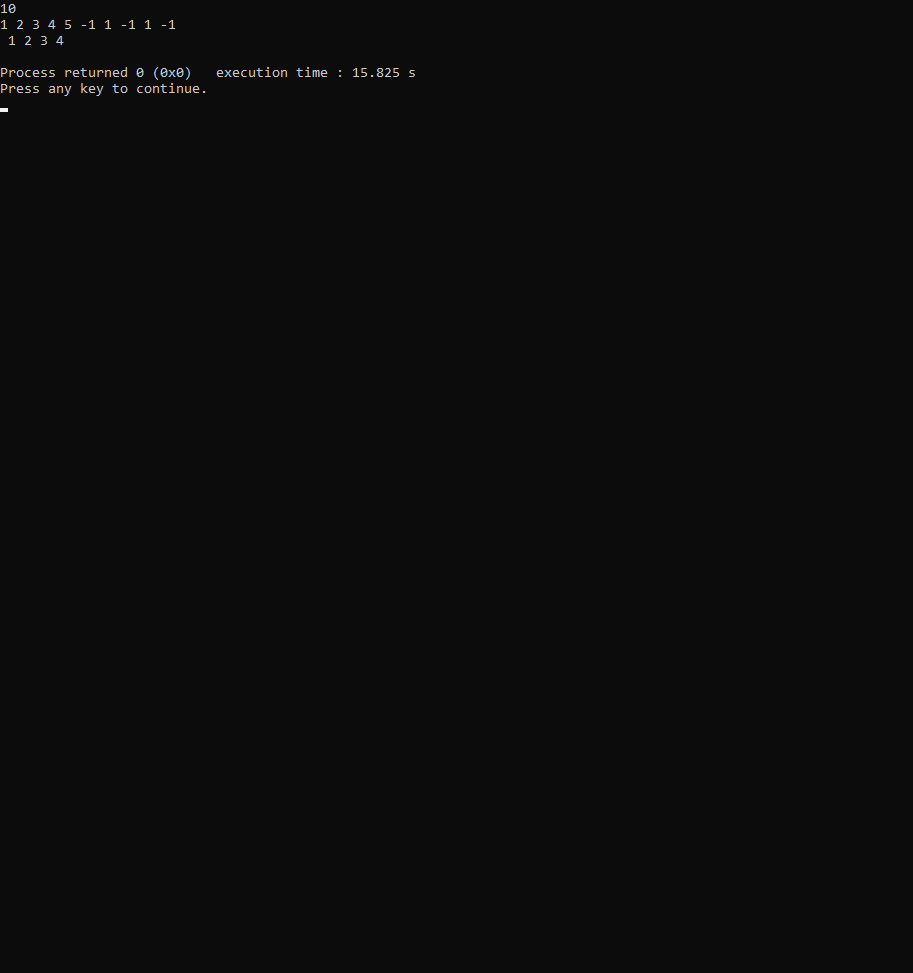
print\_arr(arr, rmlastseq(arr, arrsize));

delete [] arr;

return 0;

}

Результат:



Вывод: Я познакомился с указателями в С++. Изучил одномерные динамические массивы в языке С++. Познакомился с функциями, возвращающими несколько значений. Рассмотрел на примерах, как указатели используются в качестве параметров функций.

Блок схемы:

+-------------------+

| Начало |

+-------------------+

|

v

+-------------------+

| Ввод размера массива |

+-------------------+

|

v

+-------------------+

| Ввод элементов массива |

+-------------------+

|

v

+-------------------+

| Вызов rmlastseq() |

+-------------------+

|

v

+-------------------+

| last\_seq() |

+-------------------+

|

v

+-------------------+

| Инициализация cur|

+-------------------+

|

v

+-------------------+

| Цикл: |

| while (array < cur) |

| && !is\_seq(cur, cur - 1) |

+-------------------+

|

v

+-------------------+

| Установка \*end |

+-------------------+

|

v

+-------------------+

| Цикл: |

| while (array < cur) |

| && is\_seq(cur, cur - 1) |

+-------------------+

|

v

+-------------------+

| Установка \*begin |

+-------------------+

|

v

+-------------------+

| Возврат длины |

+-------------------+

|

v

+-------------------+

| Условие: len > 0 |

+-------------------+

|

v

+-------------------+

| Увеличение end |

+-------------------+

|

v

+-------------------+

| Цикл: |

| while (end < array) |

+-------------------+

|

v

+-------------------------------+

| Копирование оставшихся элементов |

+-------------------------------+

|

v

+-------------------------------+

| Возврат нового размера массива |

+-------------------------------+

|

v

+-------------------+

| Вызов print\_arr() |

+-------------------+

|

v

+-------------------+

| Конец |

+-------------------+

Блок схемы для вызываемых функций:

Блок-схема для last\_seq

+-------------------+

| last\_seq() |

+-------------------+

|

v

+------------------------+

| Инициализация |

| cur = array + size - 1 |

+--------------------------+

|

v

+-----------------------------+

| Цикл: |

| while (array < cur) |

| && !is\_seq(cur, cur - 1) |

+-----------------------------+

|

v

+-------------------+

| Установка \*end |

| = cur |

+-------------------+

|

v

+------------------------------+

| Цикл: |

| while (array < cur) |

| && is\_seq(cur, cur - 1) |

+------------------------------+

|

v

+-------------------+

| Установка \*begin |

| = cur |

+-------------------+

|

v

+-------------------+

| Возврат длины |

| \*end - \*begin |

+-------------------+

|

v

+-------------------+

| Конец |

+-------------------+

Блок-схема для rmlastseq

+-------------------+

| rmlastseq() |

+-------------------+

|

v

+-------------------+

| Объявление |

| int\* begin, end |

+-------------------+

|

v

+-------------------+

| Вызов last\_seq() |

+-------------------+

|

v

+-------------------+ +-------------------+

| Условие: len > 0 | – No - > | Конец |

+-------------------+ +-------------------+

|

| Yes

v

+-------------------+

| Увеличение end |

| end++ |

+-------------------+

|

v

+-------------------+

| Увеличение len |

| len++ |

+-------------------+

|

v

+-------------------+

| array += size |

+-------------------+

|

v

+-------------------+

| Цикл: |

| while (end < array) |

+-------------------+

|

v

+-------------------+

| Копирование |

| \*begin++ = \*end++|

+-------------------+

|

v

+-------------------+

| Возврат нового |

| размера массива |

+-------------------+

|

v

+-------------------+

| Конец |

+-------------------+

Блок-схема для print\_arr:

+-------------------+

| print\_arr() |

+-------------------+

|

v

+-------------------+

| Цикл: |

| while (size--) |

+-------------------+

|

v

+-------------------+

| Вывод элемента |

| cout << \*array++ |

+-------------------+

|

v

+-------------------+

| Печать новой строки|

| cout << endl |

+-------------------+

|

v

+-------------------+

| Конец |

+-------------------+