СОДЕРЖАНИЕ

Введение

Постановка задачи

Описание входных данных

Описание выходных данных

Метод решения

Описание алгоритма

Блок-схема алгоритма

Код программы

Тестирование

Заключение

Список используемой литературы (источников)

Введение

Постановка задачи

Создать объект, который обрабатывает массив целых чисел не более 10 элементов. Количество элементов определяются в момент конструирования объекта. Объект обладает следующей функциональностью:
- в конструкторе считывает количество элементов массива, выводит количество элементов;
- считывает элементы массива;
- выводит элементы массива;
- разворачивает последовательность элементов массива.

Написать программу, которая:
1. Создает объект и в конструкторе считывает количество элементов массива;
2. Считывает элементы массива;
3. Выводит значения элементов массива согласно исходной последовательности;
4. Разворачивает элементы массива;
5. Выводит значения элементов массива согласно новому их порядку следования.

Описание входных данных

Первая строка:
целое число в десятичном формате.

Вторая строка:
последовательность целых чисел в десятичном формате разделенных пробелом.

Описание выходных данных

Первая строка:
N = «количество элементов»

Вторая строка (исходный порядок следования элементов): Значения элементов массива, значение каждого элемента занимает 5 позиции, выравнивание по правому краю.

Третья строка (порядок следования элементов после разворота): Значения элементов массива, значение каждого элемента занимает 5 позиции, выравнивание по правому краю.

Метод решения

Потоки ввода вывода cin и cout

Класс MMenu

Объект NewMass

Поля класса:

intN; количествоэлементов

int *arr; указатель на массив

Методы:

MMenu(); Конструктор класса

MCin(); Ввод массива

MCout(); Вывод массива

MBack(); Разворот массива

Описание алгоритма

Функция: main

Функционал: основная программа

Параметры: нет

Возвращаемое значение: int,код возврата

№	Предикат	Действия	№ перехода	Комментарий
1		Создание объекта класса	2	
2		Вызов метода заполнения массива	3	
3		Вызов метода вывода массива	4	
4		Переход на новую строку	5	
5		Вызов метода разворота массива	6	
6		Вызов метода вывода массива	∅	

Класс объекта: MMenu

Модификатор доступа: public

Метод: MMenu

Функционал: Конструктор класса

Параметры: нет

Возвращаемое значение: void

№	Предикат	Действия	№ перехода	Комментарий
1		Ввод количесива элементов	2	
2	Количество элементов >=1 и <= 10	Создание массива	∅	да
		Вывод ошибки	∅	нет

Класс объекта: MMenu

Модификатор доступа: public

Метод: MCin

Функционал: Заполнение массива

Параметры: нет

Возвращаемое значение: void

№	Предикат	Действия	№ перехода	Комментарий
1		Сравниваем счетчик с нулем	2	
2	Счетчик меньше количества элементов	Вводим значение элемента	3	да
			∅	нет
3		Увеличиваем счетчик на 1	2	

Класс объекта: MMenu

Модификатор доступа: public

Метод: MCout

Функционал: Вывод массива

Параметры: нет

Возвращаемое значение: void

№	Предикат	Действия	№ перехода	Комментарий
1		Сравниваем счетчик с нулем	2	
2	Счетчик меньше количества элементов	Выводим значение элемента	3	да
			∅	нет
3		Увеличиваем счетчик на 1	2	

Класс объекта: MMenu

Модификатор доступа: public

Метод: MBack

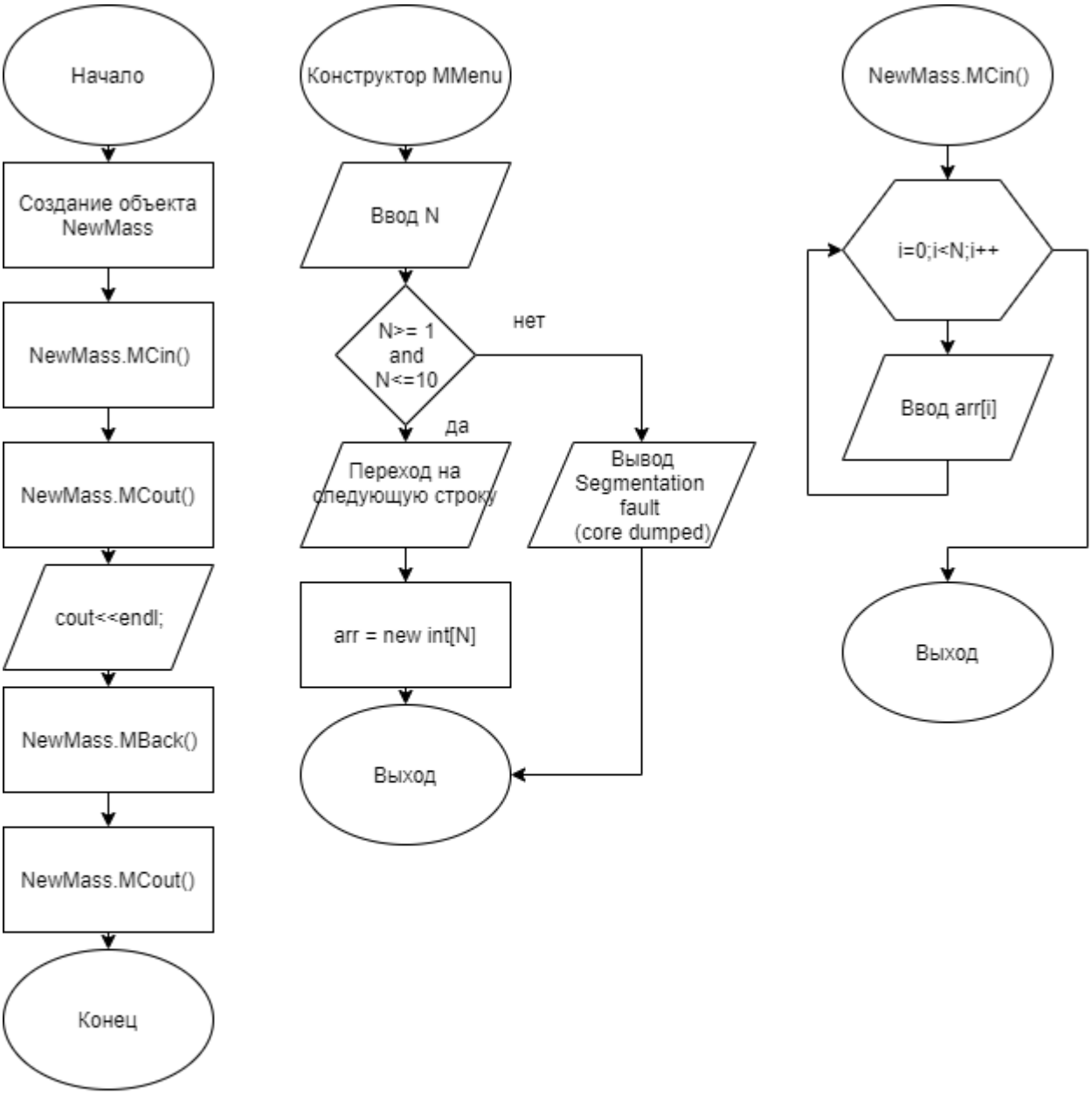
Функционал: Разворот массива

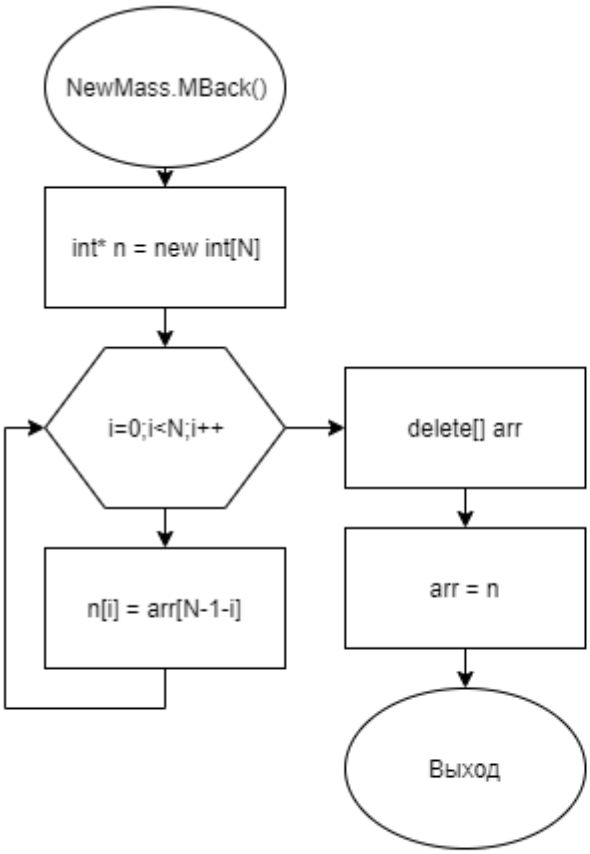
Параметры: нет

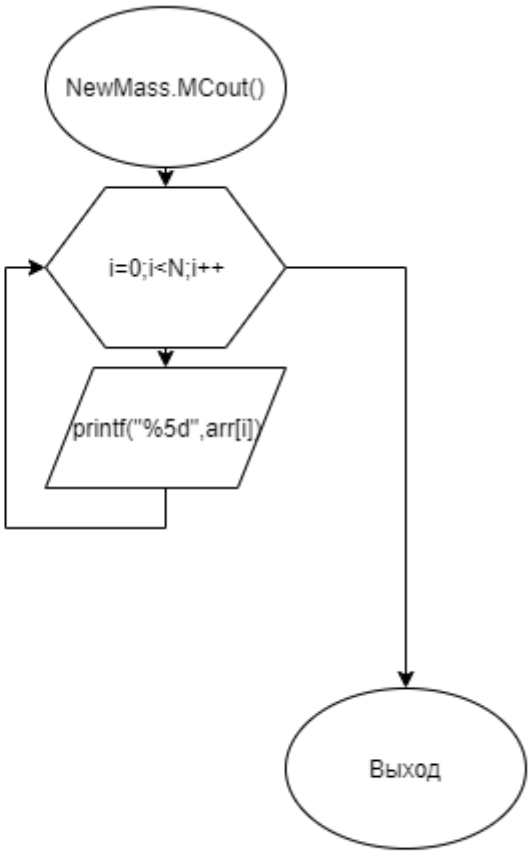
Возвращаемое значение: void

№	Предикат	Действия	№ перехода	Комментарий
1		Приравниваем счетчику 0	2	
2		Выделение памяти под int *n=new int[N]	3	
3	Счетчик меньше количества элементов	Присвоить n[i] значение arr[N-i-1]	4	
		Освободить память от начального массива	5	
4		Увеличиваем счетчик на 1	3	
5		Присвоить arr=n	∅	

Блок-схема алгоритма




NewMass.MBack()
int* n = new int[N]
i=0;i<N;i++
n[i] = arr[N-1-i]
delete[] arr
arr = n
Выход


NewMass.MCout()
i=0;i<N;i++
printf("%5d",arr[i])
Выход

Код программы

Файл main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include "MMenu.h"
using namespace std;
int main()
{
        MMenu NewMass;
        NewMass.MCin();
        NewMass.MCout();
        cout<<endl;
        NewMass.MBack();
        NewMass.MCout();
        return(0);
}
```

Файл MMenu.cpp

```
#include <iostream>
#include "MMenu.h"
using namespace std;
MMenu::MMenu()
{
        cin >> N;
        if ((N>=1) and (N<=10))
        {
                cout<<"N = "<<N<<endl;
                arr = new int[N];
        }
        else
        {
                cout<<"Segmentation fault (core dumped)";
        }
};
void::MMenu::MCin()
{
        for(int i=0;i<N;i++)
        {
                cin>>arr[i];
        }
};
void::MMenu::MCout()
{
        for(int i=0;i<N;i++)
        {
                printf("%5d",arr[i]);
```

```
        }
};
void::MMenu::MBack()
{
        int* n = new int[N];
        for(int i=0;i<N;i++)
        {
                n[i] = arr[N-1-i];
        }
        delete[] arr;
        arr=n;
};
```

Файл MMenu.h

```
#ifndef __MMENU_H
#define __MMENU_H
using namespace std;
class MMenu
{
private:
                int N;
                int *arr;
public:
                MMenu();
                void MCin();
                void MCout();
                void MBack();
};
#endif
```

Тестирование

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
16 1 2 22 4 35 6	Segmentation fault (core dumped)	Segmentation fault (core dumped)
7 1 2 3 4 5 6 7	N = 7 1 2 3 4 5 6 7 7 6 5 4 3 2 1	N = 7 1 2 3 4 5 6 7 7 6 5 4 3 2 1
6 1 2 22 4 15 6	N = 6 1 2 22 4 15 6 6 15 4 22 2 1	N = 6 1 2 22 4 15 6 6 15 4 22 2 1

Заключение

Список используемой литературы (источников)

1. Васильев А.Н. Объектно-ориентированное программирование на C++. Издательство: Наука и Техника. Санкт-Петербург, 2016г. 543 стр.

2. Шилдт Г. C++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2017. — 624 с.

3. Методическое пособие для проведения практических занятий по дисциплине «Объектно-ориентированное программирование».

4. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» в системе «Аврора».

5. Видео лекции в АСО «Аврора».