

Здесь будет титульник, листай ниже

СОДЕРЖАНИЕ

1 ПОСТАНОВКА ЗАДАЧИ.....	5
1.1 Описание входных данных.....	7
1.2 Описание выходных данных.....	7
2 МЕТОД РЕШЕНИЯ.....	9
3 ОПИСАНИЕ АЛГОРИТМОВ.....	11
3.1 Алгоритм конструктора класса MyArray.....	11
3.2 Алгоритм конструктора класса MyArray.....	11
3.3 Алгоритм деструктора класса MyArray.....	12
3.4 Алгоритм метода printArray класса MyArray.....	12
3.5 Алгоритм функции function1.....	12
3.6 Алгоритм функции function2.....	13
3.7 Алгоритм функции main.....	14
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	15
5 КОД ПРОГРАММЫ.....	19
5.1 Файл main.cpp.....	19
5.2 Файл MyArray.cpp.....	20
5.3 Файл MyArray.h.....	20
6 ТЕСТИРОВАНИЕ.....	22
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	23

1 ПОСТАНОВКА ЗАДАЧИ

Сконструировать систему, которая демонстрирует особенности при возврате объекта из функции. Показывает, какого характера ошибки могут возникнуть при выполнении возврата и как их обойти.

Спроектировать объект (разработать описание класса), который имеет свойства:

- указатель на массив целого типа в открытом доступе;
- переменную целого типа в закрытом доступе, содержит количество элементов массива.

Функционал объекта:

- конструктор по умолчанию. Выдает сообщение, что работает конструктор по умолчанию;
- параметризованный конструктор, с одним целочисленным параметром. С новой строки выводит сообщение, что работает параметризованный конструктор. Конструктор создает массив целочисленного типа размерности согласно значению параметра. Каждому элементу присваивает значение равное квадрату соответствующего индекса;
- метод, в открытом доступе, который выводит значения элементов массива с новой строки, разделенным тремя пробелами;
- деструктор, выдает с новой строки сообщение, что работает деструктор. После этого освобождает память, выделенную для массива.

Разработать функцию номером 1, с параметром целого типа. Посредством данного параметра передается значение размерности массива. В описании функции реализовать алгоритм:

1. Объявить объект, использовать параметризованный конструктор. В качестве аргумента конструктору передать параметр функции.

2. Для объекта вызвать метод вывода содержимого массива.
3. Возвратить объявленный объект.

Разработать функцию номером 2, с параметром целого типа. Посредством данного параметра передается значение размерности массива. В описании функции реализовать алгоритм:

1. Создать объект, использовать параметризованный конструктор. В качестве аргумента конструктору передать параметр функции. Инициировать указатель на объект заданного класса адресом созданного объекта.
2. Для объекта вызвать метод вывода содержимого массива.
3. Вернуть значение указателя на объект заданного класса.

Первоначальный алгоритм сборки и отработки системы:

1. Объявляется целочисленная переменная.
2. Объявляется объект заданного класса.
3. Объявляется указатель на объект заданного класса.
4. Вводится целое, положительное число, его значение размещается в целочисленную переменную.
5. Объявленному объекту присваивается значение, возвращаемое функцией с номером 1. В функцию с номером 1 в качестве аргумента передается целочисленная переменная.
6. Для объявленного объекта вызывается метод вывода содержимого массива.
7. Вводится целочисленное, положительное число, его значение размещается в целочисленную переменную.
8. Объявленному указателю на объект заданного класса присваивается значение, возвращаемое функцией с номером 2. В функцию с номером 2 в качестве аргумента передается целочисленная переменная.

9. Для объекта, адрес которой является значением указателя, вызваться метод вывода содержимого массива.

10. Уничтожается объект, адрес которого является значением указателя.

11. Завершается работа системы со значением кода возврата равного 0.

При такой реализации, работа программы завершиться аварийно или значения третьей и пятой строки вывода не совпадут.

Закомментируйте один оператор так, чтобы программа завершилась нормально и выводимые значения в третьей и пятой строке совпали.

При сдаче задания, надо пояснить, почему первоначально произошла ошибка и почему выполненная коррекция кода исправляет ситуацию.

1.1 Описание входных данных

Первая строка:

«целое число»

вторая строка:

«целое число»

Пример ввода

5
8

1.2 Описание выходных данных

Конструктор по умолчанию выводит сообщение:

Default constructor

Параметризованный конструктор выводит сообщение:

Parameterized constructor

Деструктор выводит сообщение:

Destructor

Метод вывода значений элементов массива выводит последовательность чисел:

«целое число» «целое число» «целое число» . . .

Возможный пример вывода до доработки программы-системы:

```
Default constructor
Parameterized constructor
0  1  4  9  16
Destructor
2060656  2031808  4  9  16
Parameterized constructor
0  1  4  9  16  25  36  49
0  1  4  9  16  25  36  49
Destructor
Destructor
```

2 МЕТОД РЕШЕНИЯ

Для решения задачи используется:

- объект `obj` класса `MyArray` предназначен для демонстрации работы программы;
- объект `ptr` класса `MyArray` предназначен для демонстрации работы программы;
- функция `function1` для создания и возврата объекта класса `MyArray` с параметризованным конструктором;
- функция `function2` для создания и возврата указателя на объект класса `MyArray` с параметризованным конструктором;
- функция `main` для определения входной точки программы;
- стандартная библиотека ввода-вывода;
- указатели;
- класс;
- заголовочный файл.

Класс `MyArray`:

- свойства/поля:
 - поле хранит количество элементов массива `arr`:
 - наименование — `m_size`;
 - тип — `int`;
 - модификатор доступа — `private`;
 - поле хранит в себе `m_size` целочисленных элементов:
 - наименование — `arr`;
 - тип — `int*`;
 - модификатор доступа — `public`;
- функционал:

- о метод `MyArray` — стандартный конструктор;
- о метод `MyArray` — создаёт массив длины `n` и размещает в нём квадраты индексов;
- о метод `~MyArray` — освобождает память указателя, выделенную на массив `arr`;
- о метод `printArray` — выводит содержимое массива, разделённое тремя пробелами.

3 ОПИСАНИЕ АЛГОРИТМОВ

Согласно этапам разработки, после определения необходимого инструментария в разделе «Метод», составляются подробные описания алгоритмов для методов классов и функций.

3.1 Алгоритм конструктора класса MyArray

Функционал: Выводит Default constructor.

Параметры: нет.

Алгоритм конструктора представлен в таблице 1.

Таблица 1 – Алгоритм конструктора класса MyArray

№	Предикат	Действия	№ перехода
1		Вывод "Default constructor"	Ø

3.2 Алгоритм конструктора класса MyArray

Функционал: Создает массив длины n и размещает в нём квадраты индексов.

Параметры: int n.

Алгоритм конструктора представлен в таблице 2.

Таблица 2 – Алгоритм конструктора класса MyArray

№	Предикат	Действия	№ перехода
1		Вывод "Parameterized constructor"	2
2		Инициализация m_size = n	3
3		Создание динамического целочисленного массива arr длины m_size	4
4		Инициализация int i = 0	5
5	i < m_size	arr[i] = i * i	5

№	Предикат	Действия	№ перехода
			∅

3.3 Алгоритм деструктора класса MyArray

Функционал: Вывод Destructor.

Параметры: нет.

Алгоритм деструктора представлен в таблице 3.

Таблица 3 – Алгоритм деструктора класса MyArray

№	Предикат	Действия	№ перехода
1		Вывод "Destructor"	∅

3.4 Алгоритм метода printArray класса MyArray

Функционал: Выводит содержимое массива, разделённое тремя пробелами.

Параметры: нет.

Возвращаемое значение: void.

Алгоритм метода представлен в таблице 4.

Таблица 4 – Алгоритм метода printArray класса MyArray

№	Предикат	Действия	№ перехода
1		Инициализация int i = 0	2
2	i < m_size	Вывод arr[i]	3
			4
3	i != m_size - 1	Вывод трёх пробелов	2
			2
4		Вывод "\n"	∅

3.5 Алгоритм функции function1

Функционал: Создаёт и возвращает объект класса MyArray с параметризованным конструктором.

Параметры: int size.

Возвращаемое значение: MyArray.

Алгоритм функции представлен в таблице 5.

Таблица 5 – Алгоритм функции function1

№	Предикат	Действия	№ перехода
1		Создание объекта obj класса MyArray с передачей аргумента size в параметр	2
2		Вызов obj.printArray()	3
3		Возврат obj	Ø

3.6 Алгоритм функции function2

Функционал: Создаёт и возвращает указатель на объект класса MyArray с параметризованным конструктором.

Параметры: int size.

Возвращаемое значение: MyArray*.

Алгоритм функции представлен в таблице 6.

Таблица 6 – Алгоритм функции function2

№	Предикат	Действия	№ перехода
1		Создание указателя на объект pObj класса MyArray с передачей аргумента size в параметр	2
2		Вызов pObj->printArray();	3
3		Возврат pObj	Ø

3.7 Алгоритм функции main

Функционал: Входная точка программы.

Параметры: нет.

Возвращаемое значение: int.

Алгоритм функции представлен в таблице 7.

Таблица 7 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		Инициализация целочисленной переменной number	2
2		Создание объекта obj класса MyArray	3
3		Инициализация указателя ptr нулевым указателем	4
4		Вывод number	5
5		Вызов obj.printArray()	6
6		Вывод number	7
7		Инициализация ptr = function2(number);	8
8		Вызов ptr->printArray();	9
9		Удаление объекта ptr с помощью оператора delete с освобождением памяти указателя	∅

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-4.

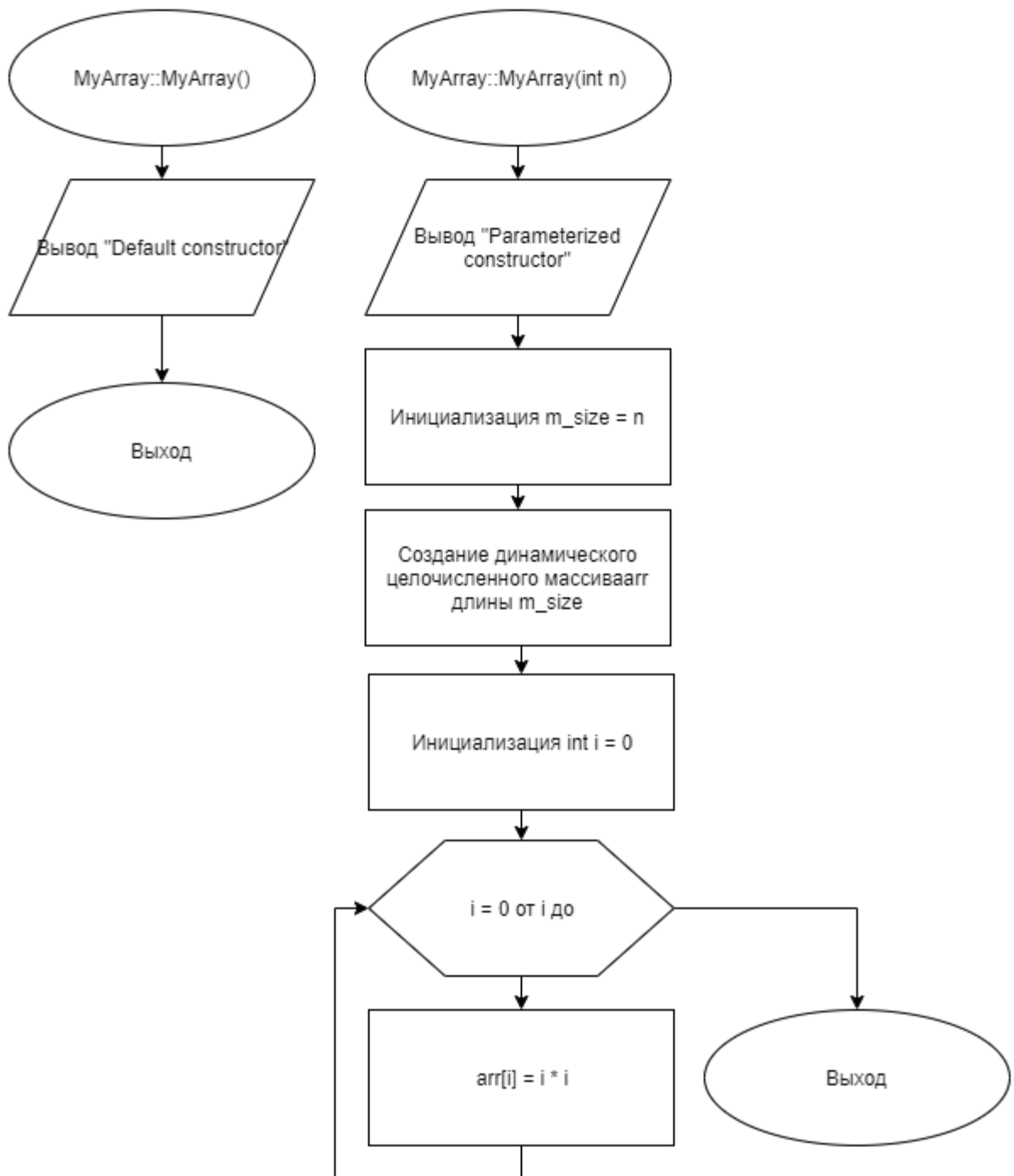


Рисунок 1 – Блок-схема алгоритма

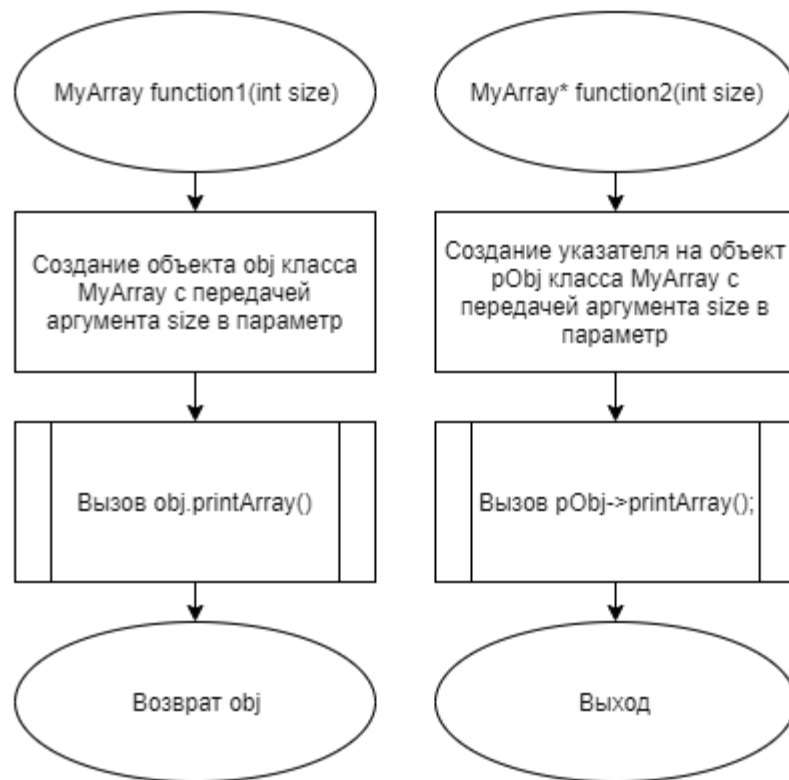


Рисунок 2 – Блок-схема алгоритма

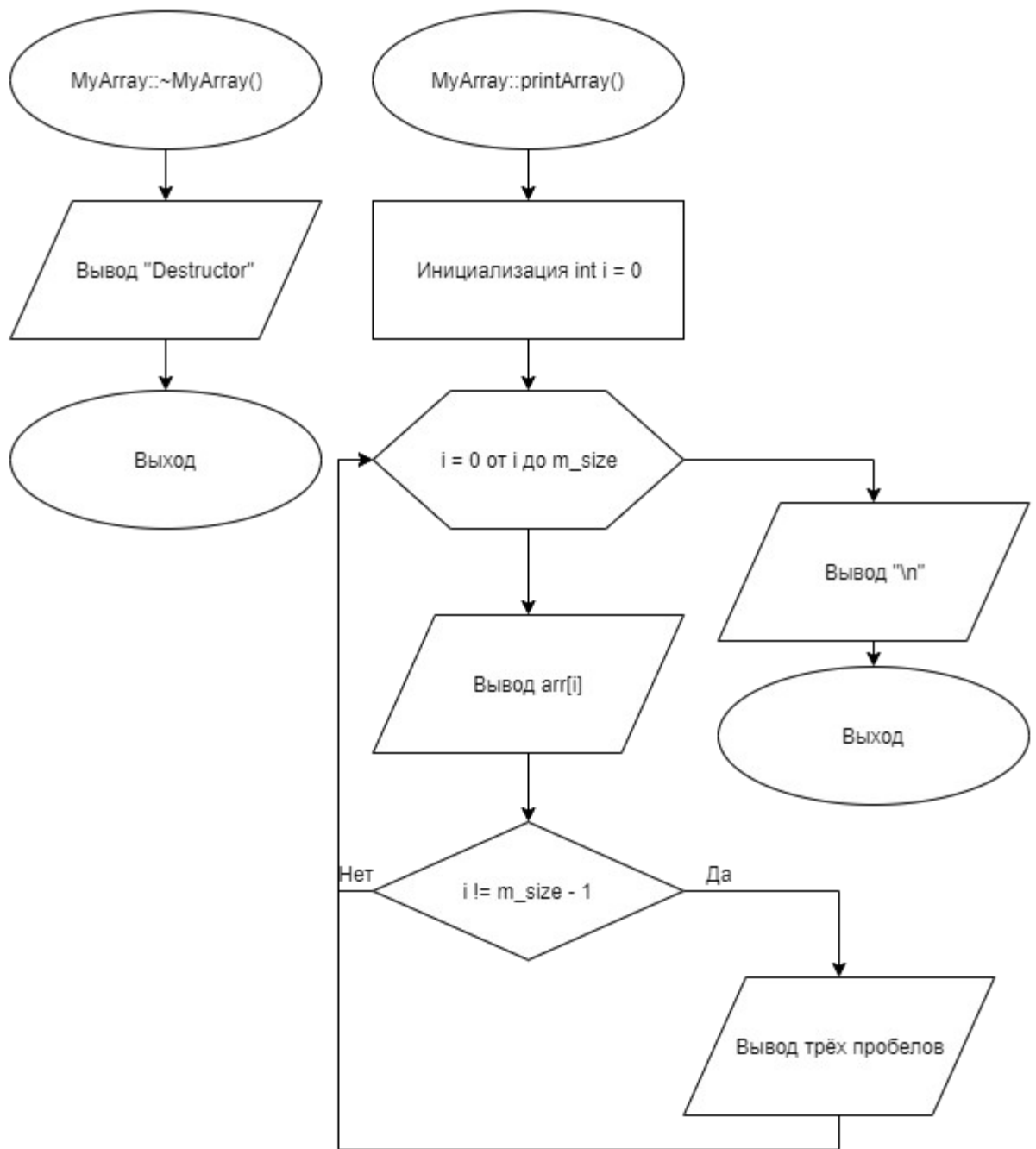


Рисунок 3 – Блок-схема алгоритма

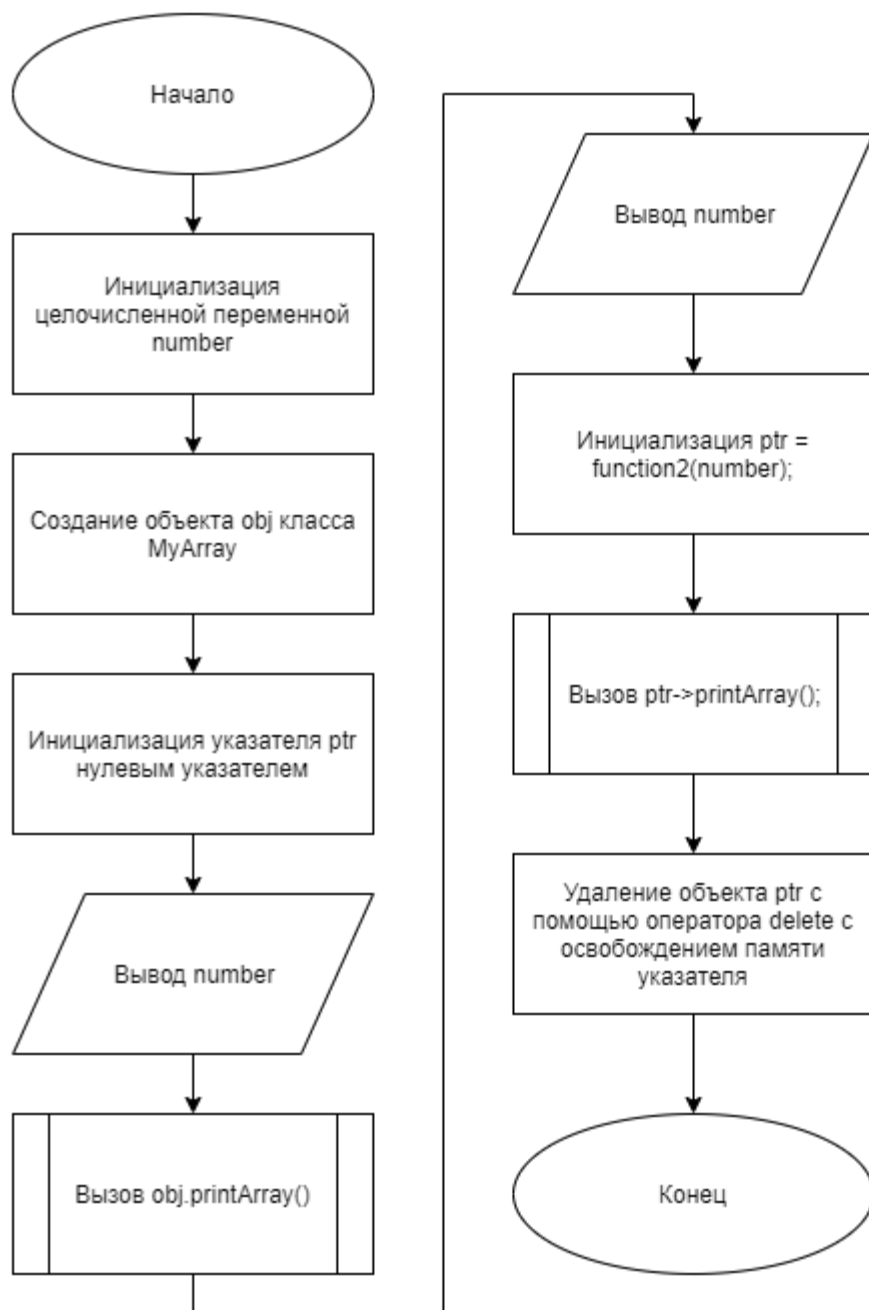


Рисунок 4 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл main.cpp

Листинг 1 – main.cpp

```
#include "MyArray.h"

MyArray function1(int size)
{
    MyArray obj(size);
    obj.printArray();
    return obj;
}

MyArray* function2(int size)
{
    MyArray* pObj = new MyArray(size);
    pObj->printArray();
    return pObj;
}

int main()
{
    int number;
    MyArray obj;
    MyArray* ptr = nullptr;

    std::cin >> number;
    obj = function1(number);
    obj.printArray();

    std::cin >> number;
    ptr = function2(number);
    ptr->printArray();

    delete ptr;

    return 0;
}
```

5.2 Файл MyArray.cpp

Листинг 2 – MyArray.cpp

```
#include "MyArray.h"

MyArray::MyArray() : arr(nullptr), m_size(0)
{
    std::cout << "Default constructor" << std::endl;
}

MyArray::MyArray(int n)
{
    std::cout << "Parameterized constructor" << std::endl;
    m_size = n;

    arr = new int[m_size];
    for (int i = 0; i < m_size; i++)
    {
        arr[i] = i * i;
    }
}

MyArray::~MyArray()
{
    std::cout << "Destructor" << std::endl;

    // delete[] arr;
}

void MyArray::printArray()
{
    for (int i = 0; i < m_size; i++)
    {
        std::cout << arr[i];
        if (i != m_size - 1) std::cout << " ";
    }
    std::cout << std::endl;
}
```

5.3 Файл MyArray.h

Листинг 3 – MyArray.h

```
#ifndef __MYARRAY__H
#define __MYARRAY__H

#include <iostream>
```

```
class MyArray
{
    int m_size;
public:
    int* arr;

    MyArray();
    MyArray(int);
    ~MyArray();

    void printArray();
};

#endif
```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 8.

Таблица 8 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
5 8	Default constructor Parameterized constructor 0 1 4 9 16 Destructor 0 1 4 9 16 Parameterized constructor 0 1 4 9 16 25 36 49 0 1 4 9 16 25 36 49 Destructor Destructor	Default constructor Parameterized constructor 0 1 4 9 16 Destructor 0 1 4 9 16 Parameterized constructor 0 1 4 9 16 25 36 49 0 1 4 9 16 25 36 49 Destructor Destructor

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. ГОСТ 19 Единая система программной документации.
2. Методическое пособие студента для выполнения практических заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] – URL: https://mirea.aco-avvora.ru/student/files/methodichescoe_posobie_dlya_laboratornyh_rabot_3.pdf (дата обращения 05.05.2021).
3. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» [Электронный ресурс]. URL: https://mirea.aco-avvora.ru/student/files/Prilozheniye_k_methodichke.pdf (дата обращения 05.05.2021).
4. Шилдт Г. С++: базовый курс. 3-е изд. Пер. с англ.. — М.: Вильямс, 2019. — 624 с.
5. Видео лекции по курсу «Объектно-ориентированное программирование» [Электронный ресурс]. АСО «Аврора».
6. Антик М.И. Дискретная математика [Электронный ресурс]: Учебное пособие /Антик М.И., Казанцева Л.В. — М.: МИРЭА — Российский технологический университет, 2018 — 1 электрон. опт. диск (CD-ROM).