

Здесь будет титульник, листай ниже

СОДЕРЖАНИЕ

1 ПОСТАНОВКА ЗАДАЧИ.....	5
1.1 Описание входных данных.....	6
1.2 Описание выходных данных.....	6
2 МЕТОД РЕШЕНИЯ.....	7
3 ОПИСАНИЕ АЛГОРИТМОВ.....	8
3.1 Алгоритм конструктора класса Array.....	8
3.2 Алгоритм деструктора класса Array.....	8
3.3 Алгоритм метода print класса Array.....	9
3.4 Алгоритм функции main.....	9
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	11
5 КОД ПРОГРАММЫ.....	14
5.1 Файл Array.cpp.....	14
5.2 Файл Array.h.....	15
5.3 Файл main.cpp.....	15
6 ТЕСТИРОВАНИЕ.....	17
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	18

1 ПОСТАНОВКА ЗАДАЧИ

Дан объект следующей конструкции:

В открытом доступе имеется указатель на массив целого типа.

Конструктору объекта передается целочисленный параметр. Параметр должен иметь значение больше 4. По значению параметра определяется размерность целочисленного массива и каждому элементу присваивается это же значение.

Объект имеет функциональность, по которой выводит содержимое целочисленного массива. Вывод производит последовательно, разделяя значения двумя пробелами.

Спроектировать систему, которая содержит два объекта. Для построения системы последовательно, с новых строк вводятся целочисленные значения. Если очередное введенное значение меньше или равно 4, то создание системы прекращается, выводится сообщение и программа завершает работу. Если очередное введенное значение больше 4, то посредством параметризованного конструктора создается первый объект, а затем, второй.

Далее система функционирует по алгоритму:

1. ...
2. Первому объекту присвоить второй объект.
3. ...
4. С первой строки вывести содержимое массива первого объекта.
5. ...
6. Со второй строки вывести содержимое массива второго объекта.

1.1 Описание входных данных

Первая строка:

«Целое число»

Вторая строка:

«Целое число»

Пример.

5
8

1.2 Описание выходных данных

Если система была построена, то в первой строке:

«Целое число» «Целое число» . . .

Во второй строке:

«Целое число» «Целое число» . . .

Если система не была построена, то в первой строке выводится некорректное значение и вопросительный знак:

«Целое число»?

Пример вывода.

5 5 5 5 5
8 8 8 8 8 8 8 8

2 МЕТОД РЕШЕНИЯ

Для решения задачи используется:

- объект `a` класса `Array` предназначен для создание объекта класса;
- объект `b` класса `Array` предназначен для создание объекта класса;
- функция `main` для основная функция;
- Объект стандартного потока ввода `cin`;
- Объект стандартного потока вывода `cout`.

Класс `Array`:

- функционал:
 - метод `Array` — конструктор класса;
 - метод `~Array` — деструктор класса;
 - метод `print` — вывод массива.

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 Array

Функционал: конструктор класса.

Параметры: value, целое число, размер массива.

Алгоритм конструктора представлен в таблице 1.

Таблица 1 – Алгоритм конструктора класса Array

№	Предикат	Действия	№ перехода
1		создается массив размером value	2
2		переменной size присваивается value	3
3	$i < \text{value}$	каждому элементу массива присваивается value	3
			Ø

3.2 Алгоритм деструктора класса Array

Функционал: деструктор класса.

Параметры: отсутствуют.

Алгоритм деструктора представлен в таблице 2.

Таблица 2 – Алгоритм деструктора класса Array

№	Предикат	Действия	№ перехода
1		удаляет объект класса	Ø

3.3 Алгоритм метода print класса Array

Функционал: вывод массива.

Параметры: отсутствуют.

Возвращаемое значение: отсутствует.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода print класса Array

№	Предикат	Действия	№ перехода
1	$i < \text{size}$	вывод элемента с индексом i	1
			Ø

3.4 Алгоритм функции main

Функционал: основная функция.

Параметры: нет.

Возвращаемое значение: целое число, код успеха.

Алгоритм функции представлен в таблице 4.

Таблица 4 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		объявление целочисленных переменных n, m	2
2		ввод n, m	3
3	$n \leq 4$	вывод n	Ø
			4
4	$m \leq 4$	вывод m	Ø
			5
5		создание объектов a и b класса Array	6
6		создание временного указателя на массив $a.\text{arr}$	7
7		$a = b$	8

№	Предикат	Действия	№ перехода
8		a.arr = temp	9
9		вызов метода a.print	10
10		вызов метода b.print	Ø

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-3.

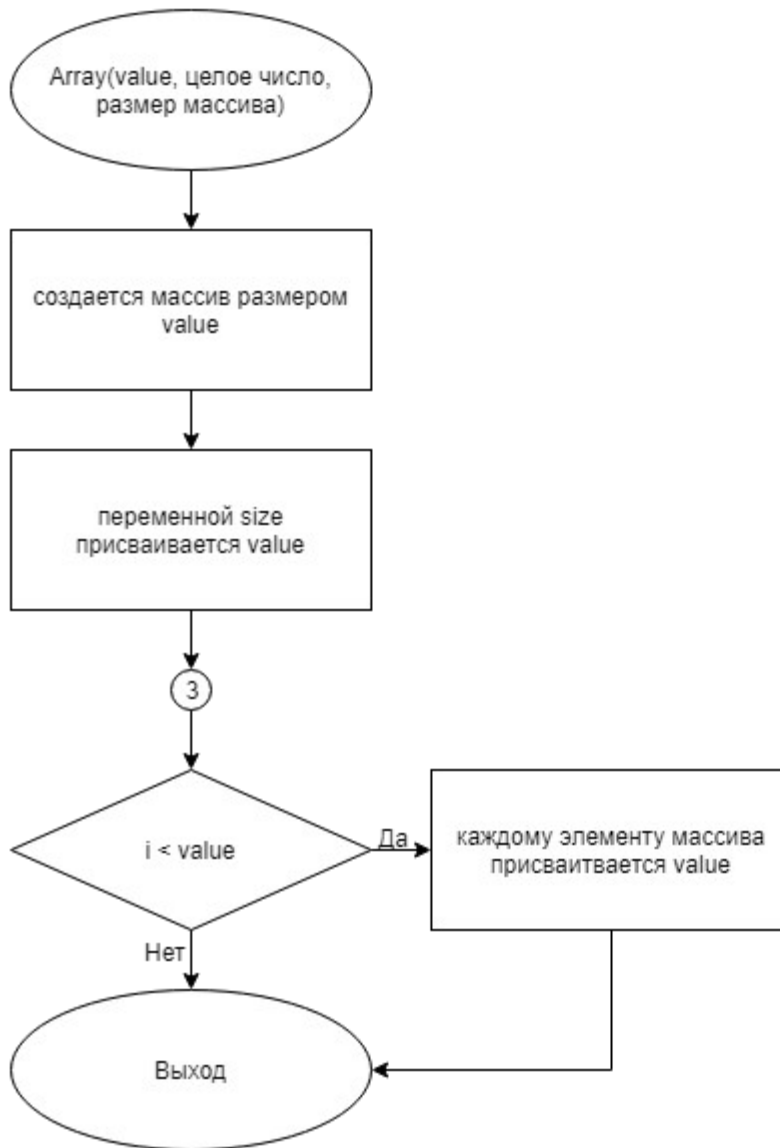


Рисунок 1 – Блок-схема алгоритма

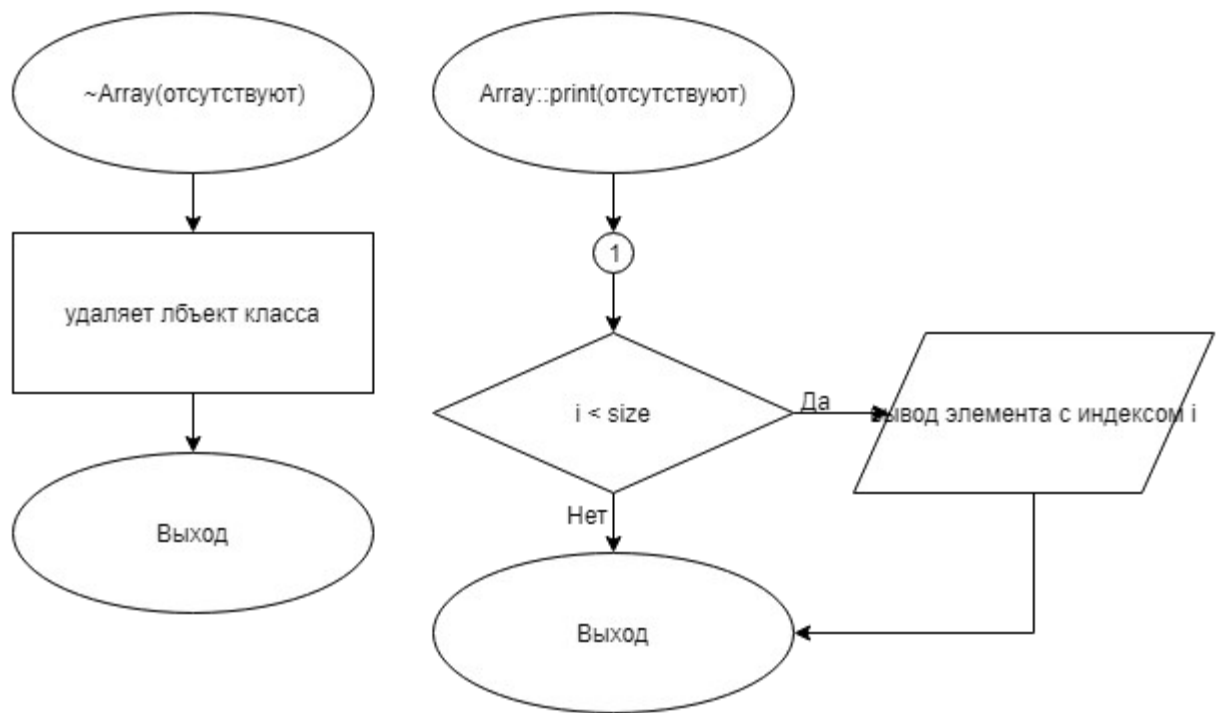


Рисунок 2 – Блок-схема алгоритма

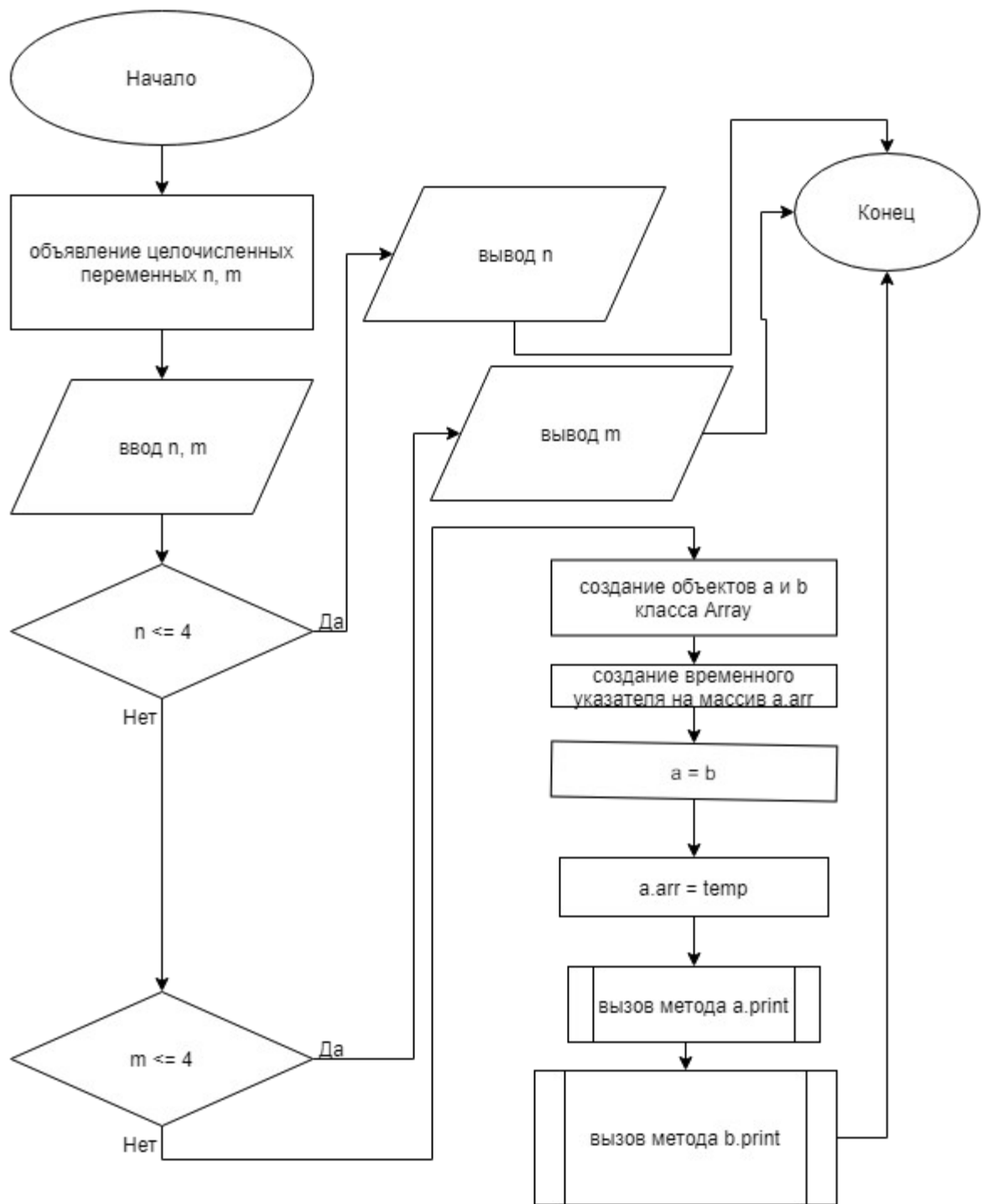


Рисунок 3 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл Array.cpp

Листинг 1 – Array.cpp

```
#include "Array.h"
#include <iostream>

using namespace std;

Array::Array(int value)
{
    arr = new int[value];

    for(int i = 0; i < value; i++)
    {
        arr[i] = value;
    }
}

Array::~Array()
{
    delete[] arr;
}

void Array::print()
{
    for (int i = 0; i < arr[0]; i++)
    {
        cout << arr[i];
        if ( i < arr[0] - 1)
        {
            cout << " ";
        }
    }
    cout << endl;
}
```

5.2 Файл Array.h

Листинг 2 – Array.h

```
#ifndef __ARRAY__H
#define __ARRAY__H

class Array
{
public:
    int* arr;
    Array (int value);
    ~Array();
    void print();
};

#endif
```

5.3 Файл main.cpp

Листинг 3 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include "Array.h"
#include <iostream>

using namespace std;

int main()
{
    int n, m;
    cin >> n >> m;
    if (n <= 4)
    {
        cout << n << "?";
        return 1;
    }
    else if (m <= 4)
    {
        cout << m << "?";
        return 1;
    }
    Array a(n);
    Array b(m);

    int* temp = a.arr;
    a = b;
```

```
    a.arr = temp;  
    a.print();  
    b.print();  
    return(0);  
}
```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 5.

Таблица 5 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
5 7	5 5 5 5 5 7 7 7 7 7 7 7	5 5 5 5 5 7 7 7 7 7 7 7

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).