

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 функции main.....	8
3.2 Алгоритм метода return класса Array.....	8
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	9
5 КОД ПРОГРАММЫ.....	10
5.1 Файл Array.cpp.....	10
5.2 Файл Array.h.....	11
5.3 Файл main.cpp.....	11
6 ТЕСТИРОВАНИЕ.....	13
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	14

1 ПОСТАНОВКА ЗАДАЧИ

Дан объект следующей конструкции:

В закрытом 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 из закрытой области и каждому элементу присваивается это же значение.

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

Функциональность объекта можно расширить по усмотрению разработчика не более чем на два метода.

Спроектировать систему, которая содержит два объекта. Для построения системы последовательно, с новых строк вводятся целочисленные значения. Если значение меньше или равно 4, то создание системы прекращается и выводится сообщение. Если система построена, то посредством параметризованного конструктора создаются объекты.

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:

- объекты стандартного ввода и вывода `cin/cout`.

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 функции main

Функционал: основная функция.

Параметры: нет.

Возвращаемое значение: целое, код успеха.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		присвоение указателю a массив со значением temp, возвращаемое методом return	Ø

3.2 Алгоритм метода return класса Array

Функционал: возвращение массива.

Параметры: нет.

Возвращаемое значение: int.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода return класса Array

№	Предикат	Действия	№ перехода
1		возвращение массива	Ø

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

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

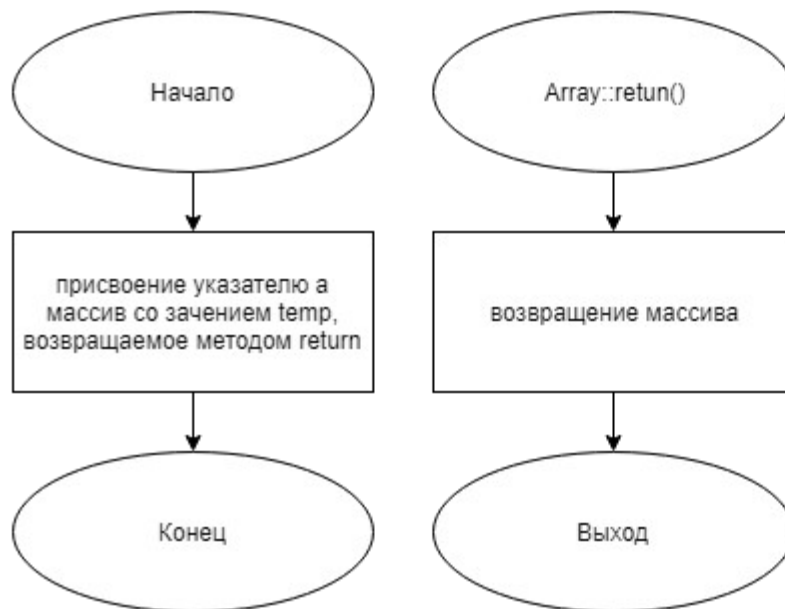


Рисунок 1 – Блок-схема алгоритма

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 razmer)
{
    arr = new int[razmer];

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

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

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

int* Array::retun()
{
    return arr;
}

void Array::Set_arr(int* new_arr)
{
    arr = new_arr;
}
```



```
}
```

5.2 Файл Array.h

Листинг 2 – Array.h

```
#ifndef __ARRAY__H
#define __ARRAY__H

class Array
{
private:
    int *arr;
public:
    Array(int razmer);
    void print();
    ~Array();
    int* return();
    void Set_arr(int* new_arr);
};

#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 g;
int main()
{
    int razmer;
    cin >> razmer;
    if (razmer <= 4)
    {
        cout << razmer << "?";
        return 0;
    }
    Array obj1(razmer);
    cin >> razmer;
    if (razmer <= 4)
```

```
    {
        cout << razmer << "?";
        return 0;
    }
    Array obj2(razmer);
    int *temp = obj1.retun();
    obj1 = obj2;
    obj1.Set_arr(temp);
    obj1.print();
    cout << endl;
    obj2.print();
    return(0);
}
```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 3.

Таблица 3 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
6 7	6 6 6 6 6 6 7 7 7 7 7 7 7	6 6 6 6 6 6 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).