

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 метода ArrGet класса Slay.....	8
3.2 Алгоритм метода ArrSet класса Slay.....	8
3.3 Алгоритм функции main.....	9
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	10
5 КОД ПРОГРАММЫ.....	11
5.1 Файл main.cpp.....	11
5.2 Файл Slay.cpp.....	11
5.3 Файл Slay.h.....	12
6 ТЕСТИРОВАНИЕ.....	14
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	15

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

Класс Slay:

- свойства/поля:
 - о поле отвечающие за объявление указателя на динамический массив:
 - наименование — arr;
 - тип — int;
 - модификатор доступа — private;
- функционал:
 - о метод те же методы , что и в 3_1_1 — ;
 - о метод ArrGet — используется для возврата значения приватного поля arr класса Slay;
 - о метод ArrSet — используется для изменения значения приватного поля arr класса Slay.

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 метода ArrGet класса Slay

Функционал: используется для возврата значения приватного поля arr класса Slay.

Параметры: нет.

Возвращаемое значение: int* temp.

Алгоритм метода представлен в таблице 1.

Таблица 1 – Алгоритм метода ArrGet класса Slay

№	Предикат	Действия	№ перехода
1		возврат значения приватного поля arr	Ø

3.2 Алгоритм метода ArrSet класса Slay

Функционал: используется для изменения значения приватного поля arr класса Slay.

Параметры: нет.

Возвращаемое значение: int* temp.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода ArrSet класса Slay

№	Предикат	Действия	№ перехода
1		Присвоение полю arr значение параметра temp	Ø

3.3 Алгоритм функции main

Функционал: Основной алгоритм программы.

Параметры: нет.

Возвращаемое значение: int.

Алгоритм функции представлен в таблице 3.

Таблица 3 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		Начало работы из предыдущего задания 3_1_1	2
2		инициализация динамического массива temp значением поля arr, путем вызова метода ArrGet объекта obj1 класса Slay	3
3		присваивание объекту obj1 значение объекта obj2	4
4		вызов метода ArrSet объекта obj1 с передачей в качестве аргумента динамического массива temp	5
5		конец работы алгоритма из предыдущей задачи 3_1_1	Ø

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

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

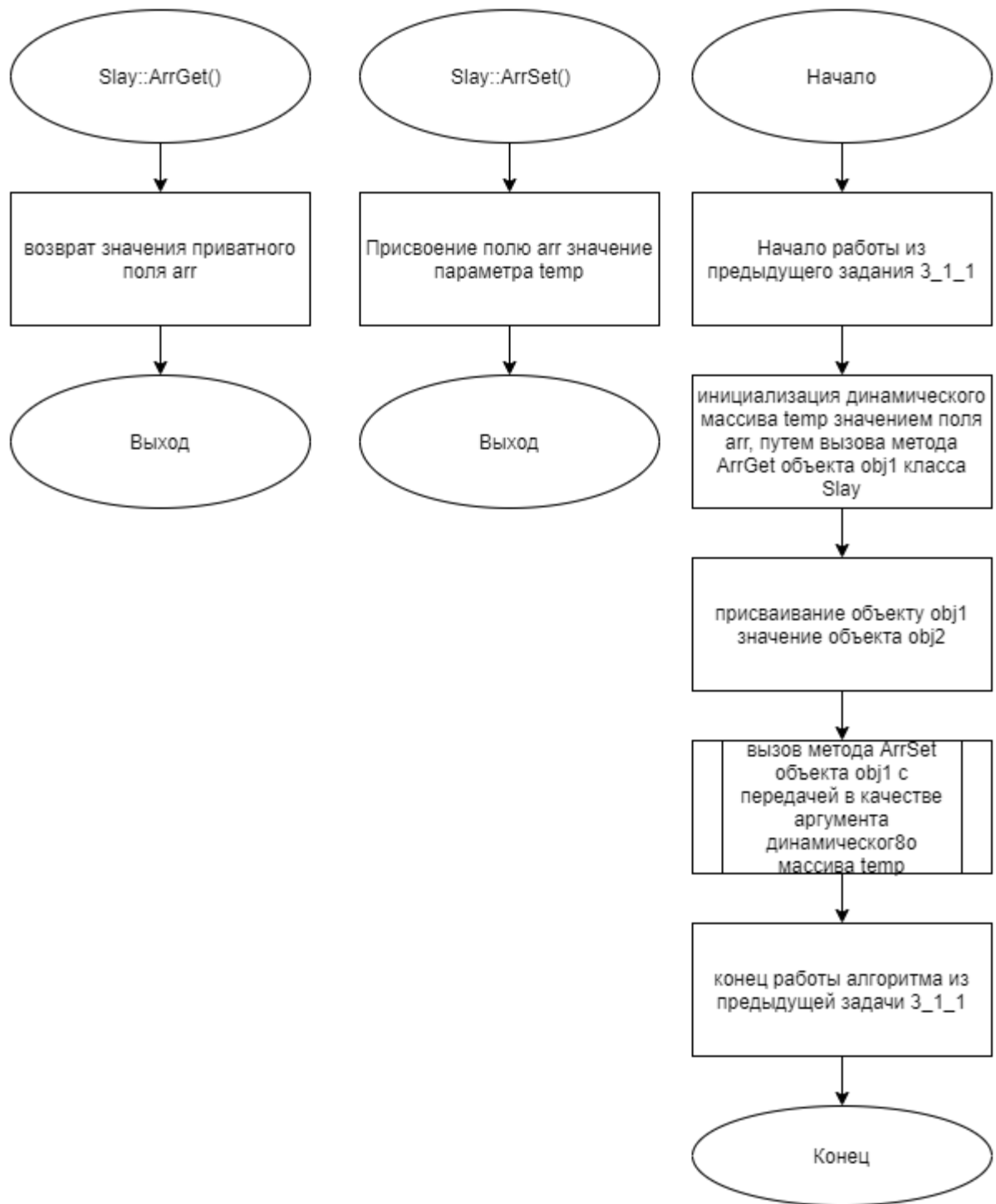


Рисунок 1 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл main.cpp

Листинг 1 – main.cpp

```
#include <iostream>
#include "Slay.h"
using namespace std;

int main(){
    int data;
    cin >> data; //ввод первого значения
    Slay obj1(data);

    cin >> data; //ввод второго значения
    Slay obj2(data);

    int* temp = obj1.ArrGet();
    obj1 = obj2;

    obj1.ArrSet(temp);
    obj1.ArrayOutput();
    cout <<endl;
    obj2.ArrayOutput();

    return (0);
}
```

5.2 Файл Slay.cpp

Листинг 2 – Slay.cpp

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

using namespace std;

Slay::Slay(int size)
{
```

```

        if (size <=4){
            arr=nullptr;
            cout << size << "?";
            exit(0);
        }
        arr = new int[size];
        for (int i =0; i < size; i++){
            arr[i] = size;
        }
    }

void Slay::ArrayOutput()
{
    cout << arr[0];
    for (int i =1; i < arr[0]; i++) cout << " " << arr[i];
}
int* Slay::ArrGet()
{
    return arr;
}
void Slay::ArrSet(int* temp)
{
    arr=temp;
}

Slay::~~Slay()
{
    if (arr != nullptr){
        delete[] arr;
    }
}

```

5.3 Файл Slay.h

Листинг 3 – Slay.h

```

#ifndef __SLAY__H
#define __SLAY__H

class Slay
{
public:

    Slay(int size);
    void ArrayOutput();
    ~Slay();
    int* ArrGet();
    void ArrSet(int* temp);
private:
    int* arr;
}

```

```
};  
#endif
```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 4.

Таблица 4 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
5 8	5 5 5 5 5 8 8 8 8 8 8 8 8	5 5 5 5 5 8 8 8 8 8 8 8 8
3 7	3?	3?
6 2	2?	2?
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).