

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
3.0 Алгоритм конструктора класса MyClass.....	8
3.1 Алгоритм метода Change класса MyClass.....	8
3.2 Алгоритм метода HydenActive класса MyClass.....	9
3.3 Алгоритм метода Print класса MyClass.....	9
3.4 Алгоритм метода HydenChange класса MyClass.....	9
3.5 Алгоритм функции main.....	10
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	11
5 КОД ПРОГРАММЫ.....	14
5.0 Файл main.cpp.....	14
5.1 Файл MyClass.cpp.....	14
5.2 Файл MyClass.h.....	15
6 ТЕСТИРОВАНИЕ.....	16
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	17

# 1 ПОСТАНОВКА ЗАДАЧИ

Спроектировать объект, в котором есть доступные и скрытые элементы (свойства и методы). Есть одно доступное свойство целого типа и одно скрытое свойство целого типа.

У объекта есть параметризованный конструктор с параметром целого типа. В конструкторе доступному свойству присваивается значение параметра, а скрытому свойству удвоенное значение параметра.

У объекта есть доступные методы со следующим функционалом:

Метод изменения значения доступного и скрытого свойства. Доступному свойству добавляется 1, скрытому свойству добавляется 4;

Метод вызова скрытого метода;

Метод вывода состояния, выводит значение доступного и скрытого свойства.

У объекта есть один скрытый метод со следующим функционалом:

1. метод доступному свойству добавляет 5, скрытому свойству добавляет 7.

Написать программу, которая состоит из описания класса выше представленного объекта и основной функции, в которой реализован следующий алгоритм:

1. Ввод целочисленного значения переменной `i_data`.
2. Создание объекта, параметризованному конструктору в качестве аргумента передается переменная `i_data`.
3. Вывод исходного состояния объекта.
4. Вызов метода изменения значений свойств объекта.
5. Вывод текущего состояния объекта.
6. Ввод целочисленного значения переменной `i_data`.
7. Непосредственное изменение доступного свойства объекта посредством

умножения его значения на i\_data.

8. Вывод текущего состояния объекта.
9. Вызов метода объекта, который вызывает скрытый метод объекта.
10. Вывод текущего состояния объекта.

## **1.1 Описание входных данных**

Первая строка

«Целочисленное значение»

Вторая строка

«Целочисленное значение»

## **1.2 Описание выходных данных**

Метод вывода состояния, первый вывод делает в первой строке, а далее всегда с новой строки. Шаблон вывода:

Value of the available property «значение доступного свойства»; Value of a hidden property «значение закрытого свойства»

## 2 МЕТОД РЕШЕНИЯ

Для решения задачи используется

Объект стандартного ввода cin;

Объект obj(i\_data) класса MyClass;

Объект стандартного вывода cout;

Класс MyClass:

Методы:

Конструктор MyClass(int i\_data)

Функционал - присвоение значений свойств(доступного/скрытого)

Наименование - i\_data

Тип - int

Метод Change()

Функционал - изменение значений свойств(доступного/скрытого)

Модификатор доступа - public

Метод HydenActive()

Функционал - вызов метода HydenChange()

Модификатор доступа - public

Метод Print()

Функционал - вывод значений свойств(доступного/скрытого)

Модификатор доступа - public

Метод HydenChange()

Функционал - изменение значений свойств(доступного/скрытого)

Модификатор доступа - private

## 3 ОПИСАНИЕ АЛГОРИТМОВ

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

### 3.0 Алгоритм конструктора класса MyClass

Функционал: Присваивание значений свойств(доступного/скрытого).

Параметры: int, i\_data, присваивание значений свойств.

Алгоритм конструктора представлен в таблице 1.

Таблица 1 – Алгоритм конструктора класса MyClass

№	Предикат	Действия	№ перехода
1		Значение доступного свойства = значению параметра	2
2		Значение скрытого свойства = значению параметра	∅

### 3.1 Алгоритм метода Change класса MyClass

Функционал: Изменение значений свойств(доступного/скрытого).

Параметры: отсутствуют.

Возвращаемое значение: Целый тип.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода Change класса MyClass

№	Предикат	Действия	№ перехода
1		Значение доступного свойства + 1	2
2		Значение скрытого свойства + 4	∅

### 3.2 Алгоритм метода HydenActive класса MyClass

Функционал: Вызывает скрытый метод.

Параметры: отсутствуют.

Возвращаемое значение: Вызываемый метод.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода HydenActive класса MyClass

№	Предикат	Действия	№ перехода
1		Вызов скрытого метода HydenChange()	Ø

### 3.3 Алгоритм метода Print класса MyClass

Функционал: Вывод значений свойств(доступного/скрытого).

Параметры: отсутствуют.

Возвращаемое значение: Текст, целый тип.

Алгоритм метода представлен в таблице 4.

Таблица 4 – Алгоритм метода Print класса MyClass

№	Предикат	Действия	№ перехода
1		Вывод значений свойств(доступного/скрытого)	Ø

### 3.4 Алгоритм метода HydenChange класса MyClass

Функционал: Изменение значений свойств(доступного/скрытого).

Параметры: отсутствуют.

Возвращаемое значение: Целый тип.

Алгоритм метода представлен в таблице 5.

Таблица 5 – Алгоритм метода HydenChange класса MyClass

№	Предикат	Действия	№ перехода
1		Значение доступного свойства + 5	2
2		Значение скрытого свойства + 7	∅

### 3.5 Алгоритм функции main

Функционал: Основная функция.

Параметры: отсутствуют.

Возвращаемое значение: Текст, целый тип.

Алгоритм функции представлен в таблице 6.

Таблица 6 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		Объявление переменной i_data	2
2		Идентификация переменной i_data	3
3		Объявление объекта obj класса MyClass	4
4		Вызов метода Print()	5
5		Вызов метода Change()	6
6		Вызов метода Print()	7
7		Идентификация переменной i_data	8
8		Присвоение доступному свойству, значение доступного свойства * i_data	9
9		Вызов метода Print()	10
10		Вызов метода HydenActive()	11
11		Вызов метода Print()	∅



## 4 БЛОК-СХЕМЫ АЛГОРИТМОВ

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

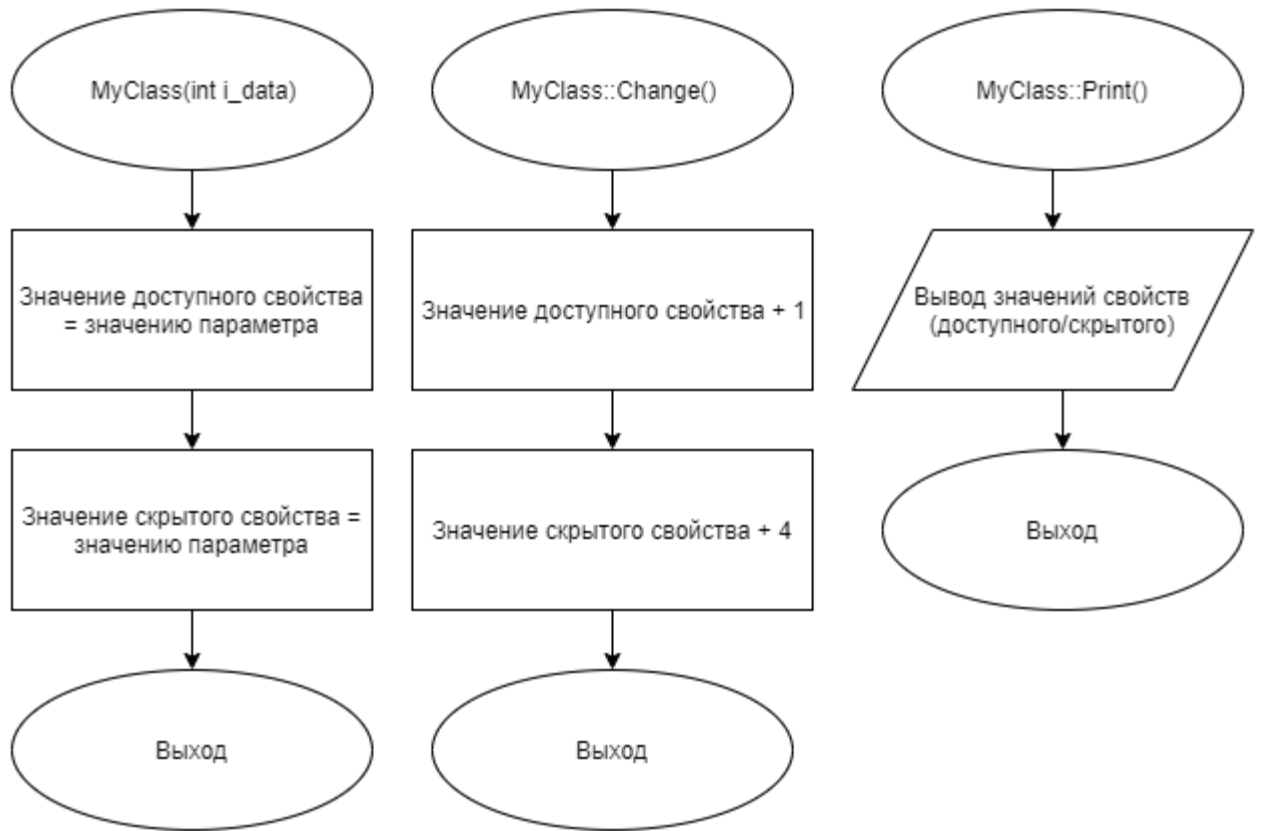
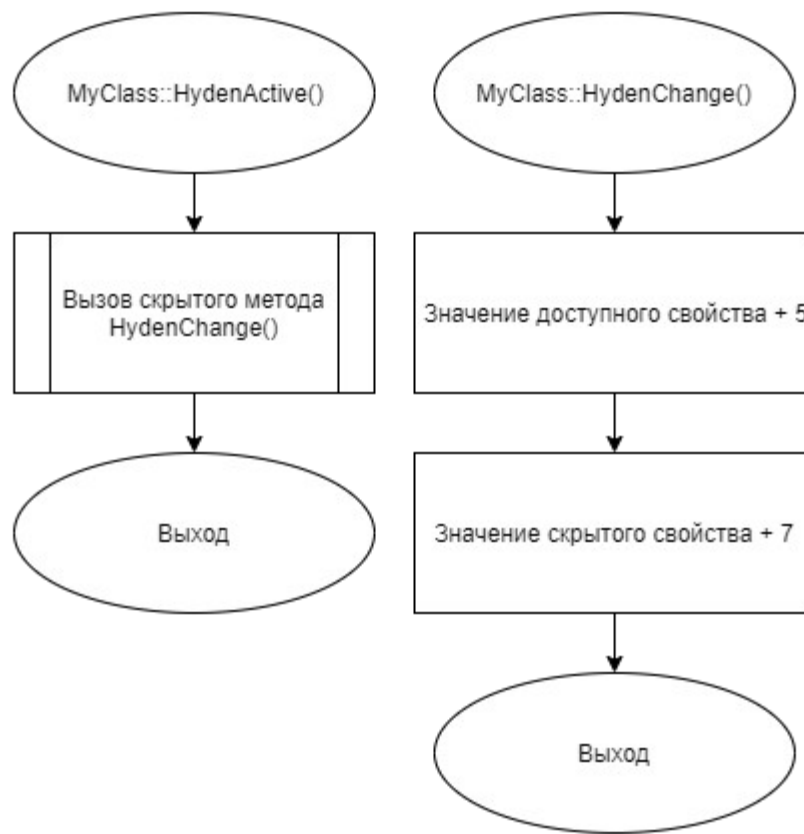


Рисунок 1 – Блок-схема алгоритма



**Рисунок 2 – Блок-схема алгоритма**

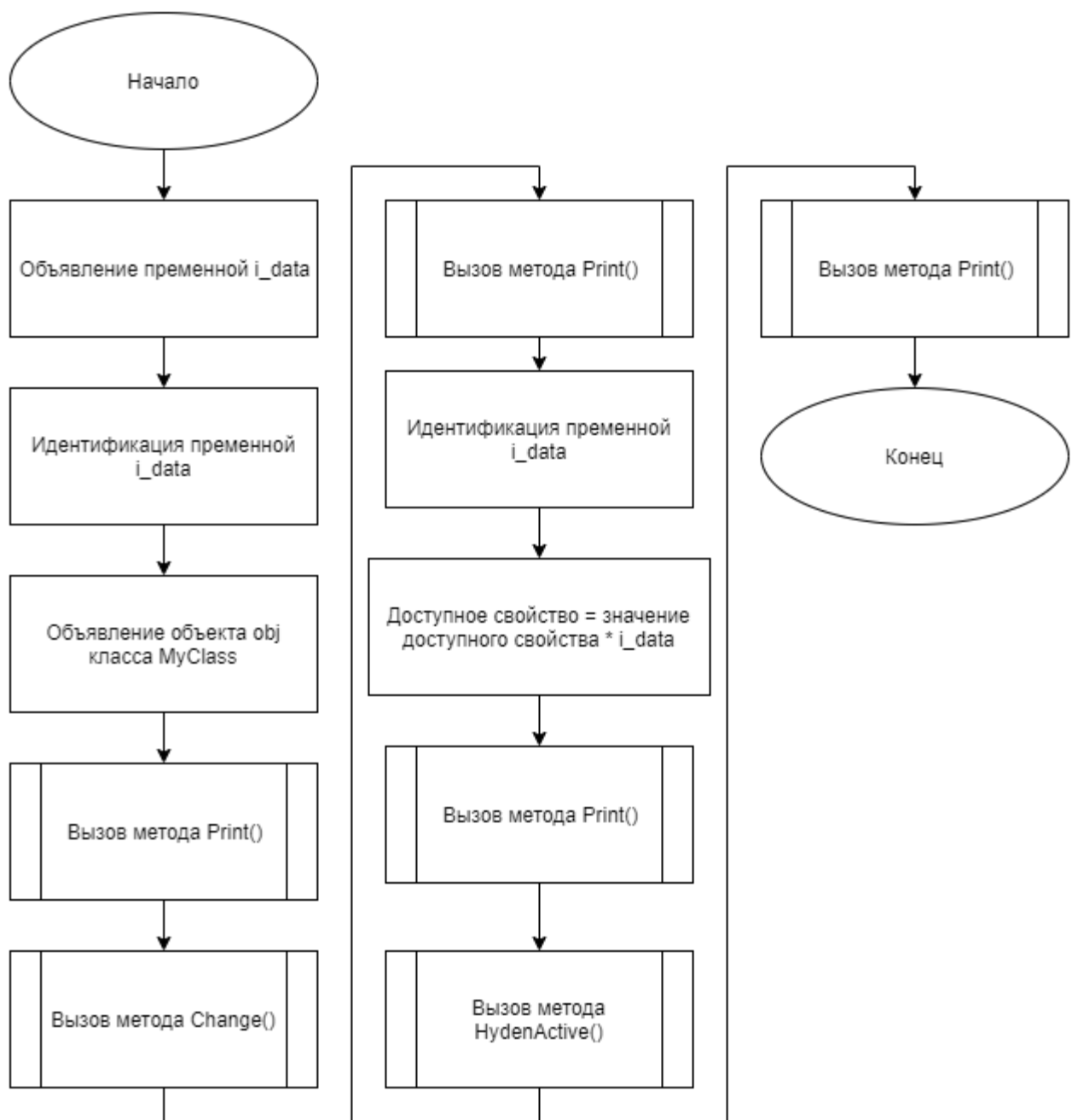


Рисунок 3 – Блок-схема алгоритма

## 5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

### 5.0 Файл main.cpp

*Листинг 1 – main.cpp*

```
#include "MyClass.h"

int main()
{
    int i_data;
    cin >> i_data;
    MyClass obj(i_data);
    obj.Print();
    cout << endl;
    obj.Change();
    obj.Print();
    cout << endl;
    cin >> i_data;
    obj.open_num = obj.open_num * i_data;
    obj.Print();
    cout << endl;
    obj.HydenActive();
    obj.Print();
}
```

### 5.1 Файл MyClass.cpp

*Листинг 2 – MyClass.cpp*

```
#include "MyClass.h"

MyClass::MyClass(int i_data)
{
    open_num = i_data;
    hyden_num = i_data * 2;
}

void MyClass::Change()
{
    open_num = open_num + 1;
    hyden_num = hyden_num + 4;
}
```

```

void MyClass::HydenActive()
{
    HydenChange();
}

void MyClass::Print()
{
    cout << "Value of the available property " << open_num << "; Value of a
hidden property " << hyden_num;
}

void MyClass::HydenChange()
{
    open_num = open_num + 5;
    hyden_num = hyden_num + 7;
}

```

## 5.2 Файл MyClass.h

*Листинг 3 – MyClass.h*

```

#ifndef __MYCLASS_H__
#define __MYCLASS_H__
#include <iostream>
using namespace std;
class MyClass
{
public:
    int open_num;
    MyClass(int i_data);
    void Change();
    void HydenActive();
    void Print();
private:
    int hyden_num;
    void HydenChange();
};
#endif

```

## 6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 7.

Таблица 7 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
2 7	Value of the available property 2; Value of a hidden property 4 Value of the available property 3; Value of a hidden property 8 Value of the available property 21; Value of a hidden property 8 Value of the available property 26; Value of a hidden property 15	Value of the available property 2; Value of a hidden property 4 Value of the available property 3; Value of a hidden property 8 Value of the available property 21; Value of a hidden property 8 Value of the available property 26; Value of a hidden property 15

## СПИСОК ИСПОЛЬЗОВАННЫХ 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 заданий, контрольных и курсовых работ по дисциплине «Объектно-ориентированное программирование» [Электронный ресурс] — URL: [https://mirea.aco-avrora.ru/student/files/methodicheskoe\\_posobie\\_dlya\\_laboratornyh\\_rabot\\_3.pdf](https://mirea.aco-avrora.ru/student/files/methodicheskoe_posobie_dlya_laboratornyh_rabot_3.pdf) (дата обращения 05.05.2021).
4. Приложение к методическому пособию студента по выполнению заданий в рамках курса «Объектно-ориентированное программирование» [Электронный ресурс]. URL: [https://mirea.aco-avrora.ru/student/files/Prilozheniye\\_k\\_methodichke.pdf](https://mirea.aco-avrora.ru/student/files/Prilozheniye_k_methodichke.pdf) (дата обращения 05.05.2021).
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).