

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 first класса MyClass.....	8
3.3 Алгоритм метода hidden класса MyClass.....	9
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	10
5 КОД ПРОГРАММЫ.....	12
5.1 Файл main.cpp.....	12
5.2 Файл MyClass.cpp.....	12
5.3 Файл MyClass.h.....	13
6 ТЕСТИРОВАНИЕ.....	14
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	15

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 параметризированный конструктор с параметром целого типа. В конструкторе доступному свойству присваивается значение параметра, а скрытому свойству утроенное значение параметра.

У объекта есть доступные методы со следующим функционалом:

- Метод изменения значения доступного и скрытого свойства. Доступному свойству добавляется 4, скрытому свойству добавляется 1;
- Метод вызова закрытого метода;
- Метод вывода состояния, выводит значение доступного и скрытого свойства.

У объекта есть один скрытый метод со следующим функционалом:

- Метод доступному свойству добавляется 7, скрытому свойству добавляется 5.

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

1. Ввод целочисленного значения переменной `i_data`.
2. Создание объекта посредством оператора функции `new` и использованием указателя на объект, параметризированному 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. Если значение `i_data` больше, чем значение доступного свойства объекта, то
 - 7.1. Присвоение доступному свойству объекта значение выражения `i_data * 8` и переход к пункту 9.
8. Иначе
 - 8.1. Переход к пункту 9.
9. Вывод текущего состояния объекта.
10. Вызов метода объекта, который вызывает скрытый метод объекта.
11. 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 с новой. 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 потока ввода cin;
- объект стандартного потока вывода cout;
- основная функция main;
- объект obj класса MyClass.

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: `int`.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции `main`

№	Предикат	Действия	№ перехода
1		создание объекта с помощью оператора функции <code>new</code> и использование указателя на объект, конструктору в качестве аргумента в качестве аргумента передается <code>i_data</code>	2
2	значение <code>i_data</code> > значение доступного свойства	присвоение доступного свойству значения <code>i_data</code>	Ø
			Ø

3.2 Алгоритм метода `first` класса `MyClass`

Функционал: метод публичных и приватных свойств объекта.

Параметры: нет.

Возвращаемое значение: `int`.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода *first* класса *MyClass*

№	Предикат	Действия	№ перехода
1		увеличение parametr_public на 1	2
2		увеличение parametr_private на 4	∅

3.3 Алгоритм метода *hidden* класса *MyClass*

Функционал: метод публичных и приватных свойств объекта.

Параметры: нет.

Возвращаемое значение: int.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода *hidden* класса *MyClass*

№	Предикат	Действия	№ перехода
1		увеличение parametr_public на 5	2
2		увеличение parametr_private на 7	∅

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-2.

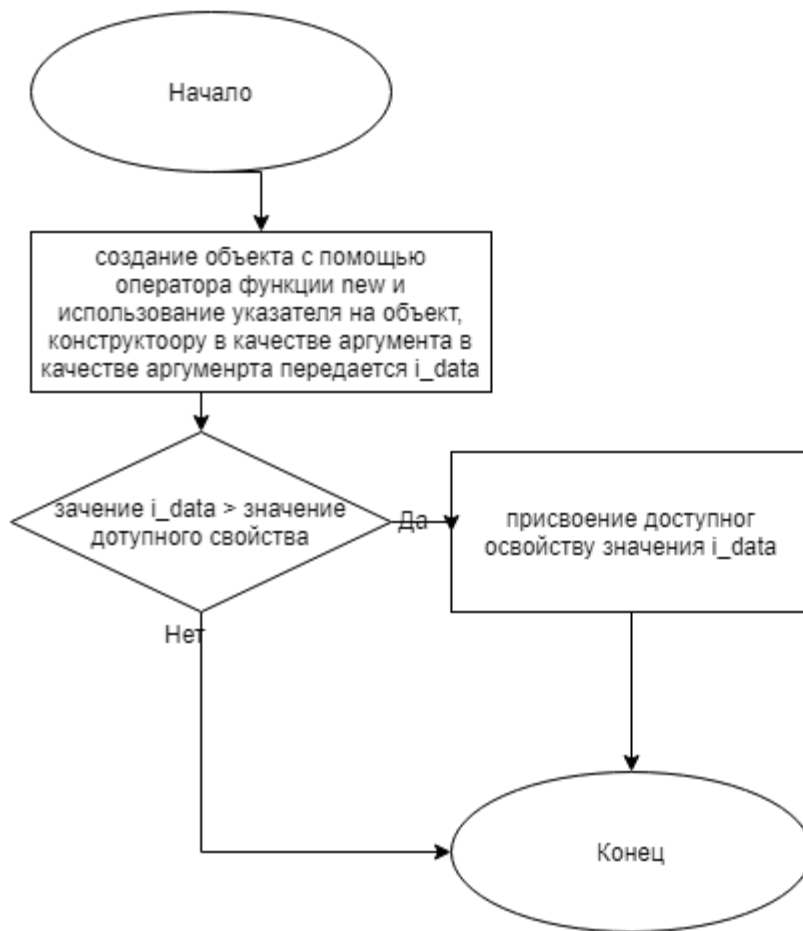


Рисунок 1 – Блок-схема алгоритма

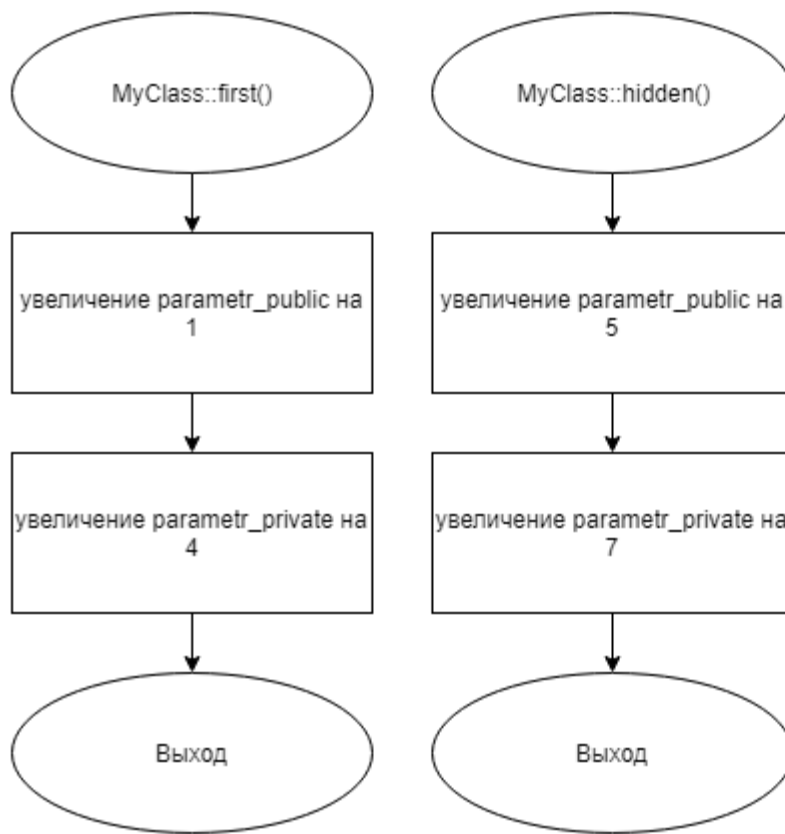


Рисунок 2 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл main.cpp

Листинг 1 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include <iostream>
#include "MyClass.h"
using namespace std;
int main()
{
    int i_data;
    cin >> i_data;
    MyClass *obj = new MyClass(i_data);
    obj->res();
    obj->first();
    obj->res();
    cin >> i_data;
    if (i_data > obj->parametr_public)
    {
        obj->parametr_public = i_data * 8;
    }
    obj->res();
    obj->second();
    obj->res();
    return(0);
}
```

5.2 Файл MyClass.cpp

Листинг 2 – MyClass.cpp

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

MyClass::MyClass(int i_data)
{
```

```

    parametr_public = i_data;
    parametr_private = i_data*3;
}

void MyClass::first()
{
    parametr_public += 4;
    parametr_private += 1;
}
void MyClass::second()
{
    hidden();
}
void MyClass::res()
{
    cout << "Value of the available property " << parametr_public << "; " <<
    "Value of a hidden property " << parametr_private << endl;
}
void MyClass::hidden()
{
    parametr_public += 7;
    parametr_private += 5;
}

```

5.3 Файл MyClass.h

Листинг 3 – MyClass.h

```

#ifndef __MYCLASS__H
#define __MYCLASS__H
class MyClass
{
private:
    int parametr_private;
    void hidden();
public:
    MyClass(int i_data);
    void first();
    void second();
    void res();
    void res2();
    int parametr_public;
};

#endif

```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 4.

Таблица 4 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
2 3	Value of the available property 2; Value of a hidden property 6 Value of the available property 6; Value of a hidden property 7 Value of the available property 6; Value of a hidden property 7 Value of the available property 13; Value of a hidden property 12	Value of the available property 2; Value of a hidden property 6 Value of the available property 6; Value of a hidden property 7 Value of the available property 6; Value of a hidden property 7 Value of the available property 13; Value of a hidden property 12

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).