

Здесь будет титульник, листай ниже

СОДЕРЖАНИЕ

1 ПОСТАНОВКА ЗАДАЧИ.....	6
1.1 Описание входных данных.....	7
1.2 Описание выходных данных.....	7
2 МЕТОД РЕШЕНИЯ.....	8
3 ОПИСАНИЕ АЛГОРИТМОВ.....	11
3.1 Алгоритм метода Vivod класса Object_1.....	11
3.2 Алгоритм конструктора класса Object_1.....	11
3.3 Алгоритм метода Vivod класса Object_2.....	11
3.4 Алгоритм конструктора класса Object_2.....	12
3.5 Алгоритм метода Vivod класса Object_3.....	12
3.6 Алгоритм конструктора класса Object_3.....	13
3.7 Алгоритм метода Vivod класса Object_4.....	13
3.8 Алгоритм конструктора класса Object_4.....	13
3.9 Алгоритм функции main.....	14
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	15
5 КОД ПРОГРАММЫ.....	18
5.1 Файл main.cpp.....	18
5.2 Файл Object_1.cpp.....	18
5.3 Файл Object_1.h.....	19
5.4 Файл Object_2.cpp.....	19
5.5 Файл Object_2.h.....	19
5.6 Файл Object_3.cpp.....	20
5.7 Файл Object_3.h.....	20
5.8 Файл Object_4.cpp.....	21
5.9 Файл Object_4.h.....	21
6 ТЕСТИРОВАНИЕ.....	22

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	23
---------------------------------------	----

1 ПОСТАНОВКА ЗАДАЧИ

Иерархия наследования

Описать четыре класса которые последовательно наследуют друг друга, последовательными номерами классов 1,2,3,4.

Реализовать программу, в которой использовать единственный указатель на объект базового класса (номер класса 1).

Наследственность реализовать так, что можно было вызывать методы, принадлежащие объекту конкретного класса, только через объект данного класса.

В закрытом разделе каждого класса определены два свойства: строкового типа для наименования объекта и целого типа для значения определенного целочисленного выражения.

Описание каждого класса содержит один параметризованный конструктор с строковым и целочисленным параметром.

В реализации каждого конструктора объекта определяются значения закрытых свойств:

- Наименование объекта по шаблону: «значение строкового параметра»_«номер класса»;
- Целочисленного свойства значением выражения возведения в степень номера класса целочисленного значения параметра конструктора.

Еще в описании каждого класса определен метод с одинаковым наименованием для всех классов, реализующий вывод значений закрытых свойств класса.

В основной функции реализовать алгоритм:

1. Вводится идентификатор и натуральное число от 2 до 10.
2. Создать объект класса 4, используя параметризованный конструктор,

которому в качестве аргументов передаются введенный идентификатор и натуральное число.

3. Построчно, для всех объектов согласно наследственности, от объекта базового (класс 1) до производного объекта (класса 4) вывести наименование объекта класса и значение целочисленного свойства.

1.1 Описание входных данных

Первая строка:

«идентификатор» «натуральное число»

Пример ввода:

Object 2

1.2 Описание выходных данных

Построчно (четыре строки):

«идентификатор»_«номер класса» «значение целочисленного свойства»

Разделитель - 1 пробел.

Пример вывода:

Object_1 2
Object_2 4
Object_3 8
Object_4 16

2 МЕТОД РЕШЕНИЯ

Для решения задачи используется:

- объект obj класса Object_1 предназначен для Выполняет свою функцию;
- объект obj класса Object_2 предназначен для Выполняет свою функцию;
- объект obj класса Object_3 предназначен для Выполняет свою функцию;
- объект obj класса Object_4 предназначен для Выполняет свою функцию;
- функция main для Основная функция;
- Объект стандартного потока ввода/ вывода cin/ cout.

Класс Object_1:

- свойства/поля:
 - поле Название:
 - наименование — name;
 - тип — Строковое;
 - модификатор доступа — private;
 - поле Номер:
 - наименование — num;
 - тип — Целочисленное;
 - модификатор доступа — private;
- функционал:
 - метод Vivod — Вывод результат;
 - метод Object_1 — Параметризированный конструктор.

Класс Object_2:

- свойства/поля:
 - поле Название:
 - наименование — name;
 - тип — Строковое;

- модификатор доступа — private;
- о поле Номер:
 - наименование — num;
 - тип — Целочисленное;
 - модификатор доступа — private;
- функционал:
 - о метод Vivod — Вывод результат;
 - о метод Object_2 — Параметризованный конструктор.

Класс Object_3:

- свойства/поля:
 - о поле Название:
 - наименование — name;
 - тип — Строковое;
 - модификатор доступа — private;
 - о поле Номер:
 - наименование — num;
 - тип — Целочисленное;
 - модификатор доступа — private;
- функционал:
 - о метод Vivod — Вывод результат;
 - о метод Object_3 — Параметризованный конструктор.

Класс Object_4:

- свойства/поля:
 - о поле Название:
 - наименование — name;
 - тип — Строковое;
 - модификатор доступа — private;

- о поле Номер:
 - наименование — num;
 - тип — Целочисленное;
 - модификатор доступа — private;
- функционал:
 - о метод Vivod — Вывод результат;
 - о метод Object_4 — Параметризованный конструктор.

Таблица 1 – Иерархия наследования классов

№	Имя класса	Классы-наследники	Модификатор доступа при наследовании	Описание	Номер
1	Object_1				
2	Object_2				
3	Object_3				
4	Object_4				

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 Vivod класса Object_1

Функционал: Вывод результат.

Параметры: нет.

Возвращаемое значение: Отсутствует.

Алгоритм метода представлен в таблице 2.

Таблица 2 – Алгоритм метода Vivod класса Object_1

№	Предикат	Действия	№ перехода
1		Выводит значения полей	Ø

3.2 Алгоритм конструктора класса Object_1

Функционал: Параметризованный конструктор.

Параметры: name, num.

Алгоритм конструктора представлен в таблице 3.

Таблица 3 – Алгоритм конструктора класса Object_1

№	Предикат	Действия	№ перехода
1		Присваивает значение полям	Ø

3.3 Алгоритм метода Vivod класса Object_2

Функционал: Вывод результат.

Параметры: нет.

Возвращаемое значение: Отсутствует.

Алгоритм метода представлен в таблице 4.

Таблица 4 – Алгоритм метода Vivod класса Object_2

№	Предикат	Действия	№ перехода
1		Выводит значение полей	Ø

3.4 Алгоритм конструктора класса Object_2

Функционал: Параметризированный конструктор.

Параметры: name, num.

Алгоритм конструктора представлен в таблице 5.

Таблица 5 – Алгоритм конструктора класса Object_2

№	Предикат	Действия	№ перехода
1		Присваивает значение полям	Ø

3.5 Алгоритм метода Vivod класса Object_3

Функционал: Вывод результат.

Параметры: нет.

Возвращаемое значение: Отсутствует.

Алгоритм метода представлен в таблице 6.

Таблица 6 – Алгоритм метода Vivod класса Object_3

№	Предикат	Действия	№ перехода
1		Выводит значение полей	Ø

3.6 Алгоритм конструктора класса Object_3

Функционал: Параметризированный конструктор.

Параметры: name, num.

Алгоритм конструктора представлен в таблице 7.

Таблица 7 – Алгоритм конструктора класса Object_3

№	Предикат	Действия	№ перехода
1		Присваивает значение полям	Ø

3.7 Алгоритм метода Vivod класса Object_4

Функционал: Вывод результат.

Параметры: нет.

Возвращаемое значение: Отсутствует.

Алгоритм метода представлен в таблице 8.

Таблица 8 – Алгоритм метода Vivod класса Object_4

№	Предикат	Действия	№ перехода
1		Выводит значение полей	Ø

3.8 Алгоритм конструктора класса Object_4

Функционал: Параметризированный конструктор.

Параметры: нет.

Алгоритм конструктора представлен в таблице 9.

Таблица 9 – Алгоритм конструктора класса Object_4

№	Предикат	Действия	№ перехода
1		присваивает значение полям	Ø

3.9 Алгоритм функции main

Функционал: Основная функция.

Параметры: нет.

Возвращаемое значение: Целое, строковое.

Алгоритм функции представлен в таблице 10.

Таблица 10 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		Вводится идентификатор и натуральное число от 2 до 10	2
2		Создать объект класса 4, используя параметризованный конструктор, которому в качестве аргументов передаются введенный идентификатор и натуральное число.	3
3		Построчно, для всех объектов согласно наследственности, от объекта базового (класс 1) до производного объекта (класса 4) вывести наименование объекта класса и значение целочисленного свойства.	Ø

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

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

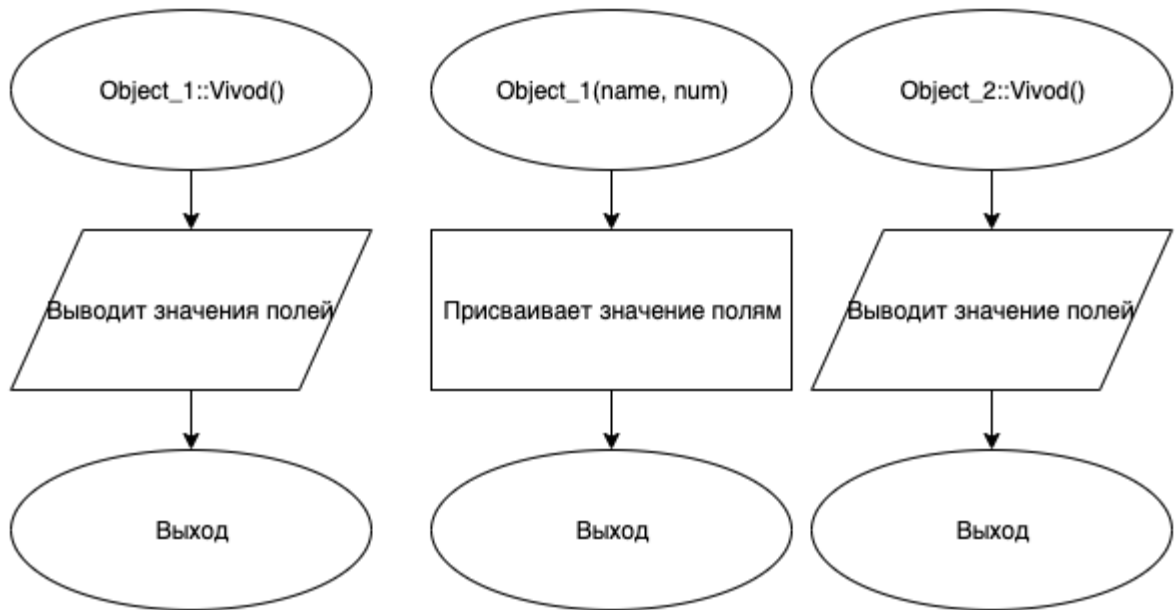


Рисунок 1 – Блок-схема алгоритма

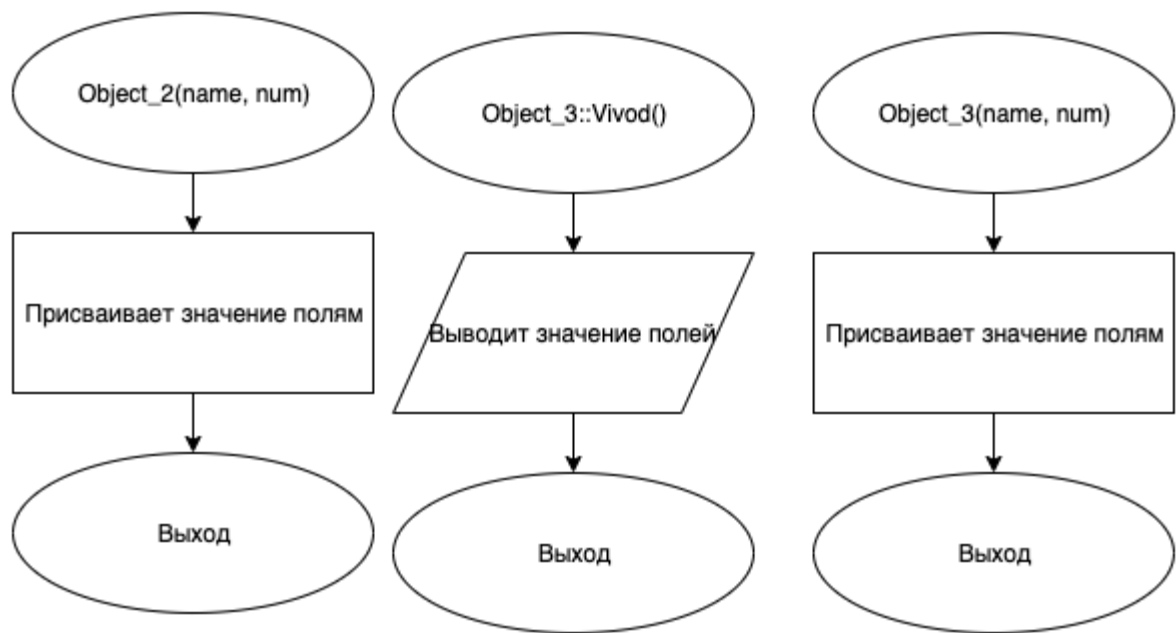


Рисунок 2 – Блок-схема алгоритма

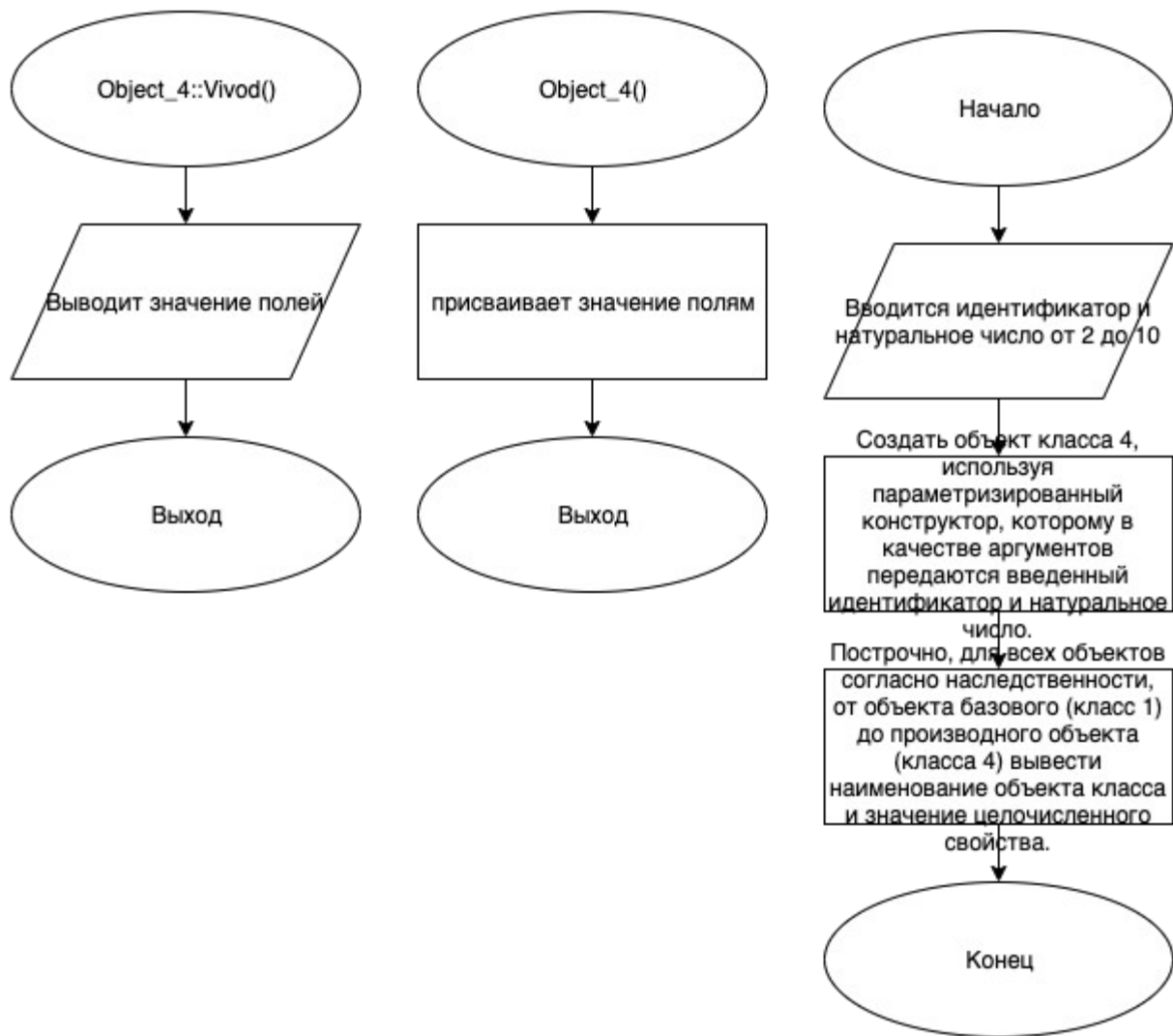


Рисунок 3 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл main.cpp

Листинг 1 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include "Object_4.h"
#include <iostream>
using namespace std;
int main()
{
    string id;
    int num;
    cin >> id >> num;
    Object_4* obj = new Object_4(id, num);
    ((Object_1*)obj)-> Vivod();
    ((Object_2*)obj)-> Vivod();
    ((Object_3*)obj)-> Vivod();
    obj -> Vivod();
    return(0);
}
```

5.2 Файл Object_1.cpp

Листинг 2 – Object_1.cpp

```
#include "Object_1.h"
#include <iostream>
Object_1::Object_1(string name, int num) : name(name + "_1"),
num(pow(num,1))
{
}
void Object_1::Vivod()
{
    cout << name << " " << num << endl;
}
```


5.3 Файл Object_1.h

Листинг 3 – Object_1.h

```
#ifndef __OBJECT_1__H
#define __OBJECT_1__H
#include <string>
#include <cmath>
using namespace std;
class Object_1
{
private:
    string name;
    int num;
public:
    Object_1(string name, int num);
    void Vivod();
};

#endif
```

5.4 Файл Object_2.cpp

Листинг 4 – Object_2.cpp

```
#include "Object_2.h"
#include <iostream>
Object_2::Object_2(string name, int num):Object_1(name, num)
{
    this->name = name + "_2";
    this->num = pow(num, 2);
}
void Object_2::Vivod()
{
    cout << name << " " << num << endl;
}
```

5.5 Файл Object_2.h

Листинг 5 – Object_2.h

```
#ifndef __OBJECT_2__H
#define __OBJECT_2__H
```

```

#include <string>
#include <cmath>
#include "Object_1.h"
using namespace std;
class Object_2: private Object_1{
private:
    string name;
    int num;
public:
    Object_2(string, int);
    void Vivod();
};

#endif

```

5.6 Файл Object_3.cpp

Листинг 6 – Object_3.cpp

```

#include "Object_3.h"
#include <iostream>

Object_3::Object_3(string name, int num):Object_2(name, num)
{
    this->name = name + "_3";
    this->num = pow(num, 3);
}
void Object_3::Vivod()
{
    cout << name << " " << num << endl;
}

```

5.7 Файл Object_3.h

Листинг 7 – Object_3.h

```

#ifndef __OBJECT_3__H
#define __OBJECT_3__H
#include <string>
#include "Object_2.h"
class Object_3: private Object_2
{
private:
    string name;
    int num;
}

```

```

public:
    Object_3(string, int);
    void Vivod();
};
#endif

```

5.8 Файл Object_4.cpp

Листинг 8 – Object_4.cpp

```

#include "Object_4.h"
#include <iostream>

Object_4::Object_4(string name, int num):Object_3(name, num)
{
    this->name = name + "_4";
    this->num = pow(num, 4);
}
void Object_4::Vivod()
{
    cout << name << " " << num << endl;
}

```

5.9 Файл Object_4.h

Листинг 9 – Object_4.h

```

#ifndef __OBJECT_4__H
#define __OBJECT_4__H
#include <string>
#include "Object_3.h"
using namespace std;
class Object_4: private Object_3
{
private:
    string name;
    int num;
public:
    Object_4(string, int);
    void Vivod();
};
#endif

```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 11.

Таблица 11 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
Object 2	Object_1 2 Object_2 4 Object_3 8 Object_4 16	Object_1 2 Object_2 4 Object_3 8 Object_4 16

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).