Здесь будет титульник, листай ниже

СОДЕРЖАНИЕ

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 вызвать 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 параметризированный 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 параметризированный 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 cl4 класса cl_obj4 предназначен для решения с единственным указателем;
- cin/cout - объекты стандартного потока ввода/вывола;
- new- оператор выделения динамической памяти.

Класс cl_obj1:

- свойства/поля:
    - поле Имя объекта:
        - наименование — name;
        - тип — string;
        - модификатор доступа — private;
    - поле Числовое значение:
        - наименование — num;
        - тип — int;
        - модификатор доступа — private;
- функционал:
    - метод cl_obj1 — Параметризированный конструктор;
    - метод print — Вывод закрытых парамтеров.

Класс cl_obj2:

- свойства/поля:
    - поле Имя объекта:
        - наименование — name;
        - тип — string;
        - модификатор доступа — private;
    - поле Числовое значение:

        - наименование — num;
        - тип — int;
        - модификатор доступа — private;
- функционал:
    - метод cl_obj2 — Параметризированный конструктор;
    - метод print — Вывод закрытых парамтеров.

Класс cl_obj3:

- свойства/поля:
    - поле Имя объекта:
        - наименование — name;
        - тип — string;
        - модификатор доступа — private;
    - поле Числовое значение:
        - наименование — num;
        - тип — int;
        - модификатор доступа — private;
- функционал:
    - метод cl_obj3 — Параметризированный конструктор;
    - метод print — Вывод закрытых парамтеров.

Класс cl_obj4:

- свойства/поля:
    - поле Имя объекта:
        - наименование — name;
        - тип — string;
        - модификатор доступа — private;
    - поле Числовое значение:
        - наименование — num;

        - тип — int;
        - модификатор доступа — private;
- функционал:
    - метод cl_obj4 — Параметризированный конструктор;
    - метод print — Вывод закрытых парамтеров.

Таблица 1 – Иерархия наследования классов

№	Имя класса	Классы-наследники	Модификатор доступа при наследовании	Описание	Номер
1	cl_obj1			Базовый класс	
		cl_obj2	private		2
2	cl_obj2			Базовый класс	
		cl_obj3	private		3
3	cl_obj3			Базовый класс	
		cl_obj4	private		4
4	cl_obj4			Базовый класс	

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 конструктора класса cl_obj1

Функционал: Параметризированный конструктор.

Параметры: string name - имя, int num - числовое значение.

Алгоритм конструктора представлен в таблице 2.

Таблица 2 – Алгоритм конструктора класса cl_obj1

№	Предикат	Действия	№ перехода
1		Установка поля name равному значению параметра name + _1	2
2		Установка поля num равному значению параметра num	∅

3.2 Алгоритм метода print класса cl_obj1

Функционал: Вывод закрытых парамтеров.

Параметры: нет.

Возвращаемое значение: void.

Алгоритм метода представлен в таблице 3.

Таблица 3 – Алгоритм метода print класса cl_obj1

№	Предикат	Действия	№ перехода
1		Вывод значения полей name и int	∅

3.3 Алгоритм конструктора класса cl_obj2

Функционал: Параметризированный конструктор.

Параметры: string name - имя, int num - числовое значение.

Алгоритм конструктора представлен в таблице 4.

Таблица 4 – Алгоритм конструктора класса cl_obj2

№	Предикат	Действия	№ перехода
1		Передача параметров name и int родительскому конструктору cl_obj1	2
2		Установка поля name равному значению параметра name + _2	3
3		Установка поля num равному значению параметра num * num	∅

3.4 Алгоритм метода print класса cl_obj2

Функционал: Вывод закрытых парамтеров.

Параметры: нет.

Возвращаемое значение: void.

Алгоритм метода представлен в таблице 5.

Таблица 5 – Алгоритм метода print класса cl_obj2

№	Предикат	Действия	№ перехода
1		Вывод значения полей name и int	∅

3.5 Алгоритм конструктора класса cl_obj3

Функционал: Параметризированный конструктор.

Параметры: string name - имя, int num - числовое значение.

Алгоритм конструктора представлен в таблице 6.

Таблица 6 – Алгоритм конструктора класса cl_obj3

№	Предикат	Действия	№ перехода
1		Передача параметров name и int родительскому конструктору cl_obj2	2
2		Установка поля name равному значению параметра name + _3	3
3		Установка поля num равному значению параметра num * num * num	∅

3.6 Алгоритм метода print класса cl_obj3

Функционал: Вывод закрытых парамтеров.

Параметры: нет.

Возвращаемое значение: void.

Алгоритм метода представлен в таблице 7.

Таблица 7 – Алгоритм метода print класса cl_obj3

№	Предикат	Действия	№ перехода
1		Вывод значения полей name и int	∅

3.7 Алгоритм конструктора класса cl_obj4

Функционал: Параметризированный конструктор.

Параметры: string name - имя, int num - числовое значение.

Алгоритм конструктора представлен в таблице 8.

Таблица 8 – Алгоритм конструктора класса cl_obj4

№	Предикат	Действия	№ перехода
1		Передача параметров name и int родительскому конструктору cl_obj3	2
2		Установка поля name равному значению параметра name + _4	3
3		Установка поля num равному значению параметра num * num * num * num	∅

3.8 Алгоритм метода print класса cl_obj4

Функционал: Вывод закрытых парамтеров.

Параметры: нет.

Возвращаемое значение: void.

Алгоритм метода представлен в таблице 9.

Таблица 9 – Алгоритм метода print класса cl_obj4

№	Предикат	Действия	№ перехода
1		Вывод значения полей name и int	∅

3.9 Алгоритм функции main

Функционал: главаня функция программы.

Параметры: нет.

Возвращаемое значение: int - код ошибки.

Алгоритм функции представлен в таблице 10.

Таблица 10 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		Объявление переменных name типа string и num типа int	2
2		Ввод значений переменных name и num	3
3		Создание указателя cl4 типа cl_obj4 с передачей параметров name и num конструктору	4
4		Вызов метода print у объекта по указателю cl4 с приведением типа на cl_obj1	5
5		Вызов метода print у объекта по указателю cl4 с приведением типа на cl_obj2	6
6		Вызов метода print у объекта по указателю cl4 с приведением типа на cl_obj3	7

№	Предикат	Действия	№ перехода
7		Вызов метода print у объекта по указателю cl4	8
8		Возврат значения 0	∅

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

Представим описание алгоритмов в графическом виде на рисунках 1-5.

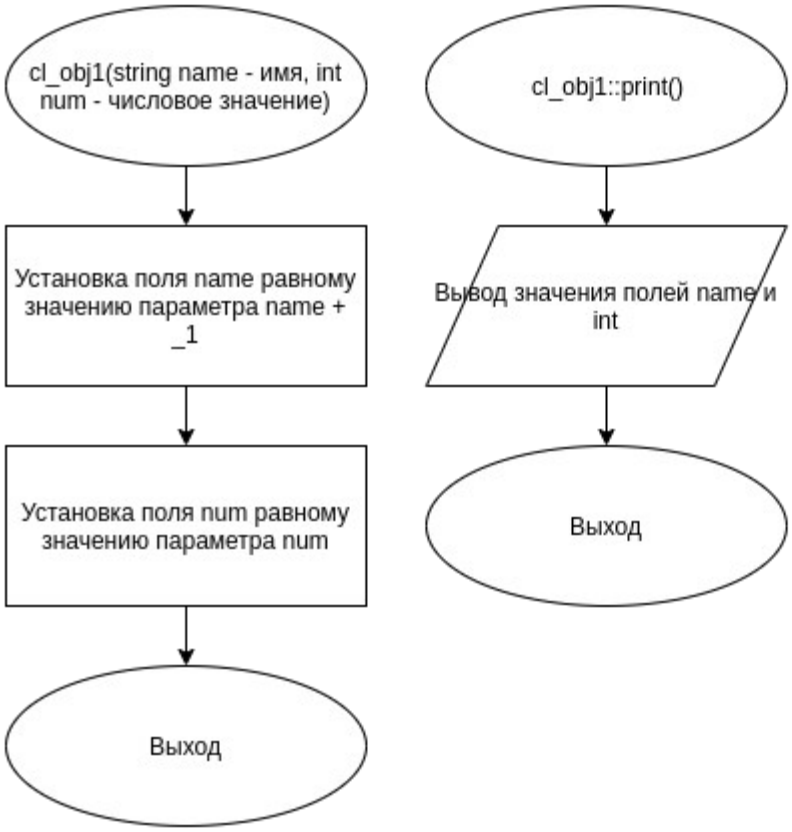

Рисунок 1 – Блок-схема алгоритма

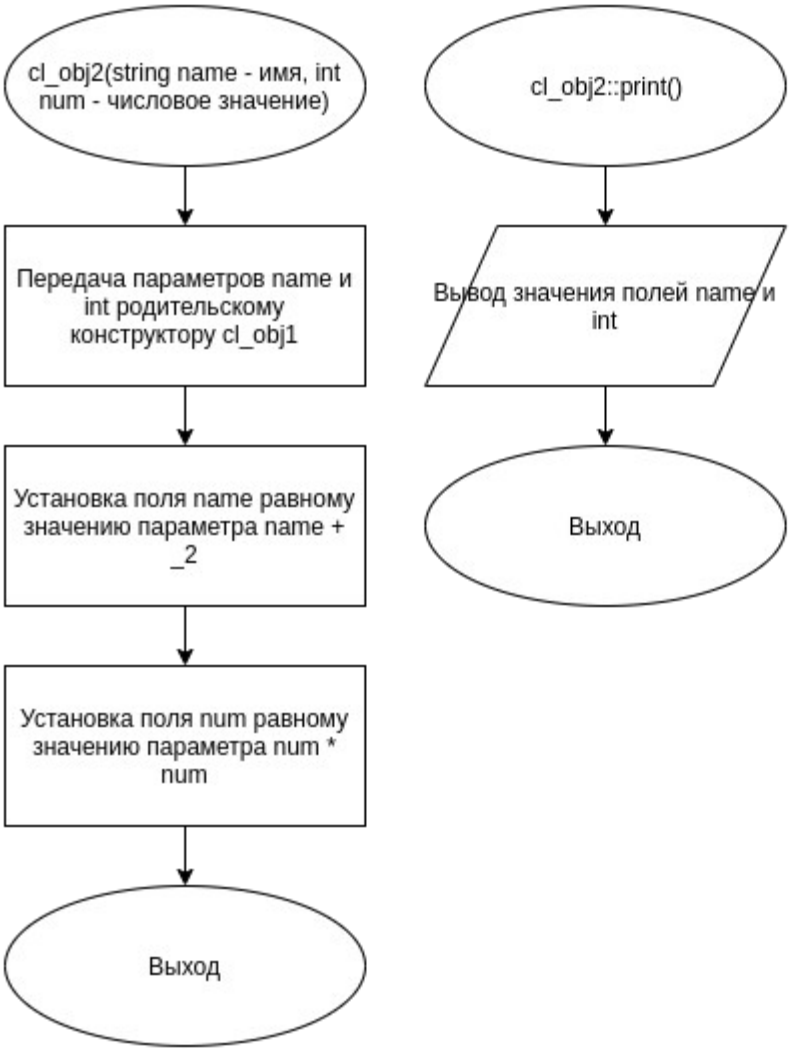

Рисунок 2 – Блок-схема алгоритма

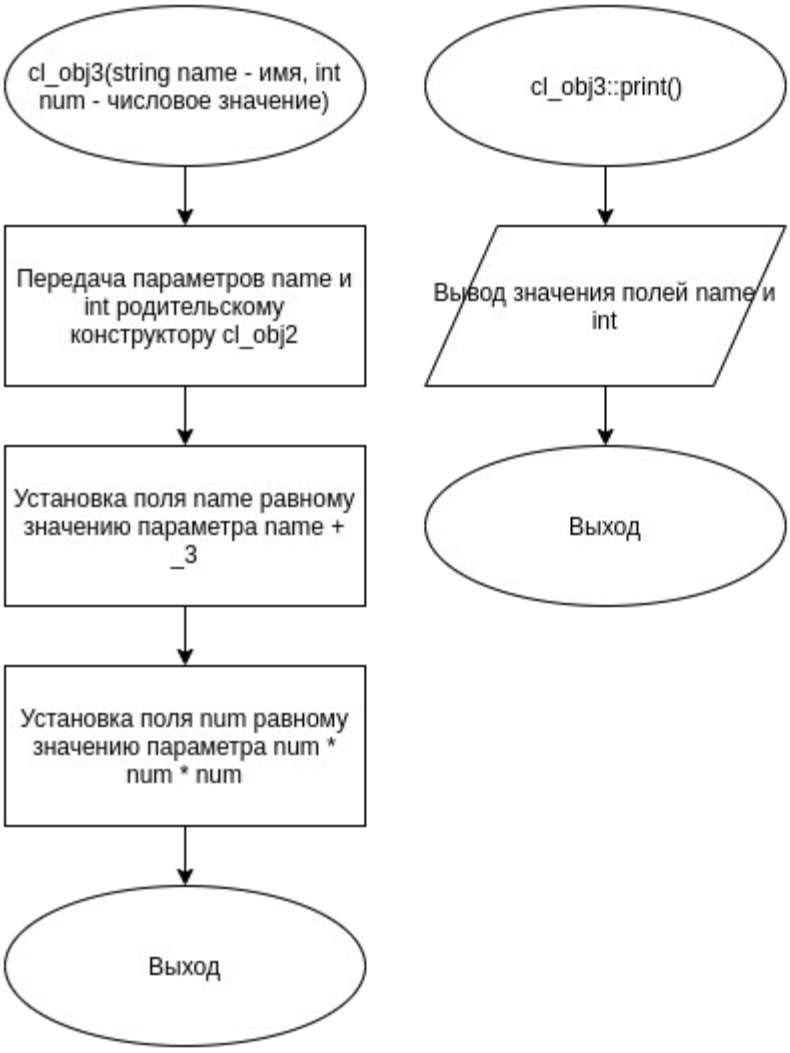

Рисунок 3 – Блок-схема алгоритма

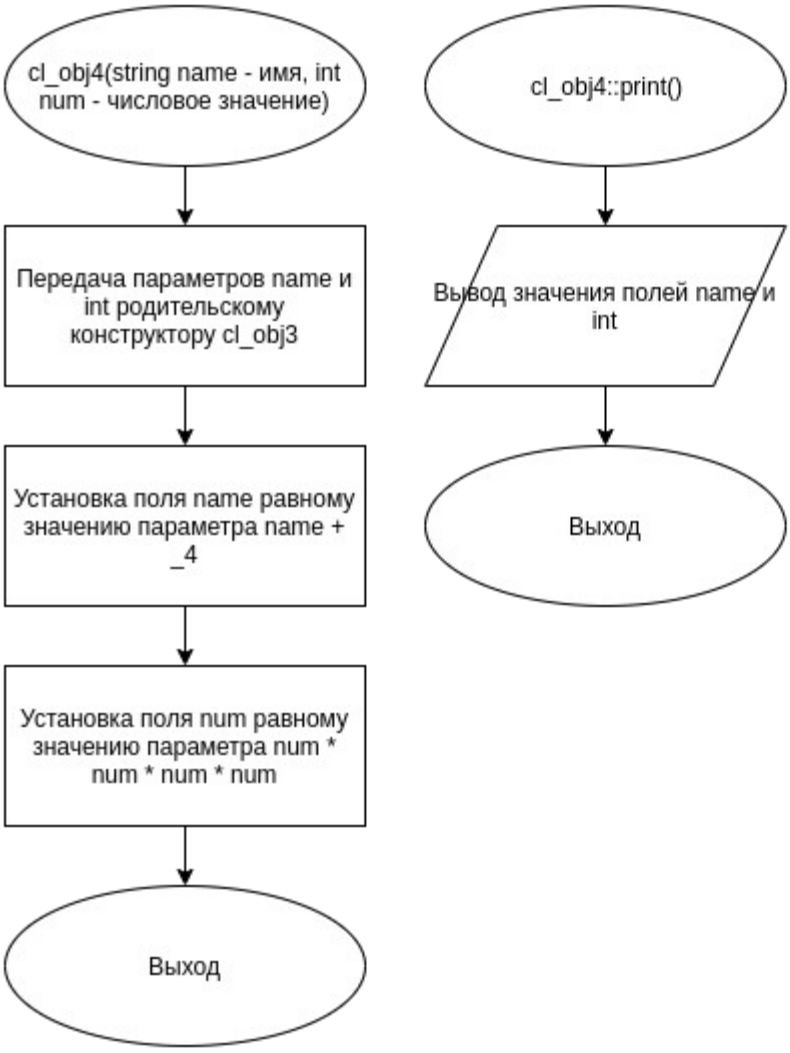

Рисунок 4 – Блок-схема алгоритма

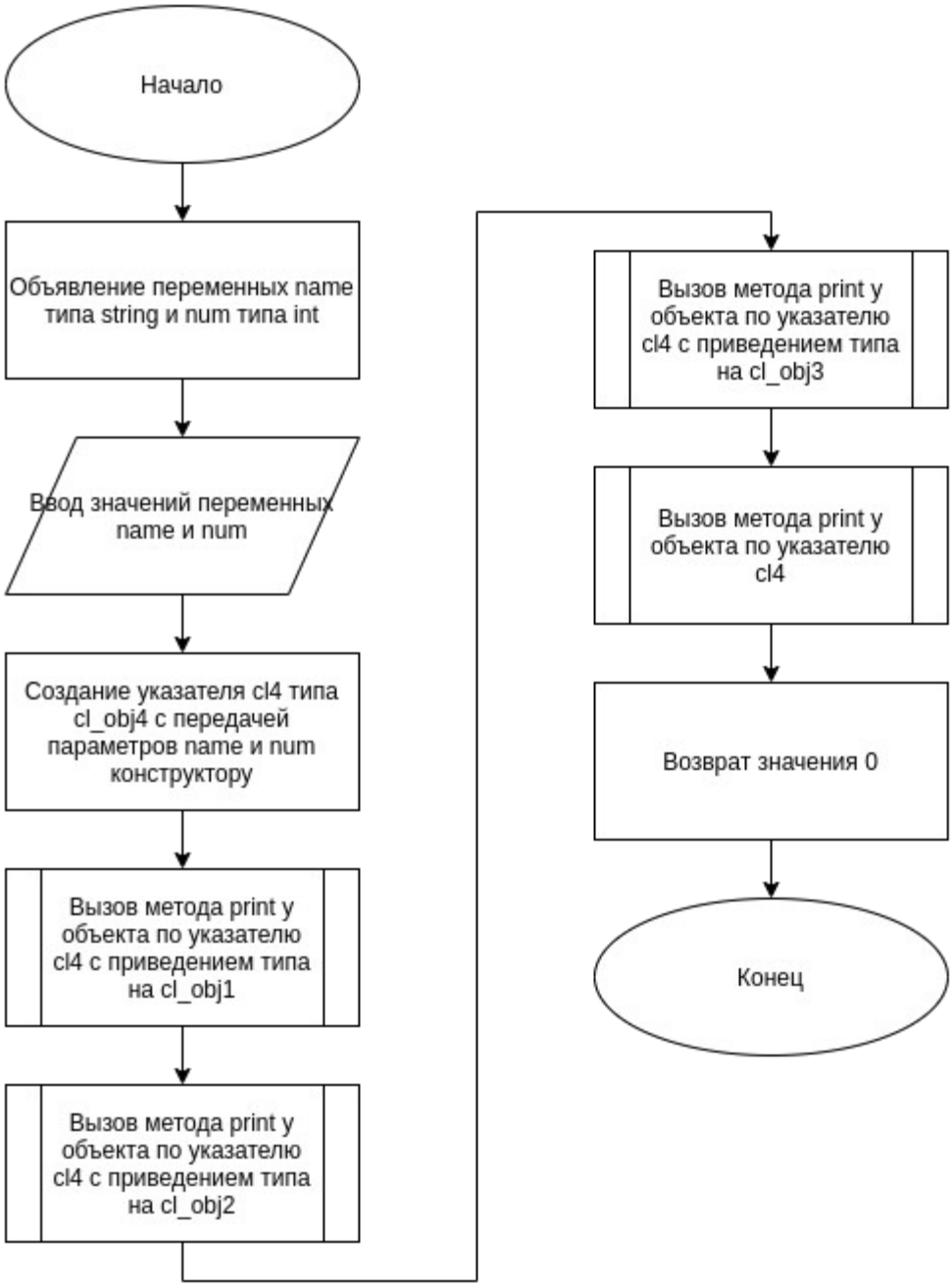

Рисунок 5 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл cl_obj1.cpp

Листинг 1 – cl_obj1.cpp

```
#include "cl_obj1.h"



cl_obj1::cl_obj1(string name, int num){
   this->name = name + "_1";
   this->num = num;
}

void cl_obj1::print() {
   cout << name << " " << num;
}
```

5.2 Файл cl_obj1.h

Листинг 2 – cl_obj1.h

```
#ifndef __CL_OBJ1__H
#define __CL_OBJ1__H
#include <string>
#include <iostream>

using namespace std;

class cl_obj1
{
private:
   string name;
   int num;
public:
   cl_obj1(string name, int num);
   void print();
};
```

```
#endif
```

5.3 Файл cl_obj2.cpp

Листинг 3 – cl_obj2.cpp

```
#include "cl_obj2.h"

cl_obj2::cl_obj2(string name, int num) : cl_obj1(name, num){
   this->name = name + "_2";
   this->num = num * num;
}

void cl_obj2::print() {
   cout << name << " " << num;
}
```

5.4 Файл cl_obj2.h

Листинг 4 – cl_obj2.h

```
#ifndef __CL_OBJ2__H
#define __CL_OBJ2__H
#include "cl_obj1.h"

using namespace std;

class cl_obj2 : private cl_obj1
{
private:
   string name;
   int num;
public:
   cl_obj2(string name, int num);
   void print();
};

#endif
```

5.5 Файл cl_obj3.cpp

Листинг 5 – cl_obj3.cpp

```
#include "cl_obj3.h"


cl_obj3::cl_obj3(string name, int num) : cl_obj2(name, num){
   this->name = name + "_3";
   this->num = num * num * num;
}

void cl_obj3::print() {
   cout << name << " " << num;
}
```

5.6 Файл cl_obj3.h

Листинг 6 – cl_obj3.h

```
#ifndef __CL_OBJ3__H
#define __CL_OBJ3__H

#include "cl_obj2.h"

using namespace std;

class cl_obj3 : private cl_obj2
{
private:
   string name;
   int num;
public:
   cl_obj3(string name, int num);
   void print();
};

#endif
```

5.7 Файл cl_obj4.cpp

Листинг 7 – cl_obj4.cpp

```
#include "cl_obj4.h"
```

```
cl_obj4::cl_obj4(string name, int num) : cl_obj3(name, num){
   this->name = name + "_4";
   this->num = num * num * num * num;
}

void cl_obj4::print() {
   cout << name << " " << num;
}
```

5.8 Файл cl_obj4.h

Листинг 8 – cl_obj4.h

```
#ifndef __CL_OBJ4__H
#define __CL_OBJ4__H
#include "cl_obj3.h"

using namespace std;

class cl_obj4 : private cl_obj3
{
private:
   string name;
   int num;
public:
   cl_obj4(string name, int num);
   void print();
};


#endif
```

5.9 Файл main.cpp

Листинг 9 – main.cpp

```
#include <stdlib.h>
#include <stdio.h>
#include "cl_obj4.h"

int main()
{
   string name;
```

```
        int num;

        cin >> name >> num;
        cl_obj4 *cl4 = new cl_obj4(name, num);
        ((cl_obj1*)cl4)->print();
        cout << endl;
        ((cl_obj2*)cl4)->print();
        cout << endl;

        ((cl_obj3*)cl4)->print();
        cout << endl;

        cl4->print();
        return(0);
     }
```

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-avrora.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-avrora.ru/student/files/Prilozheniye_k_methodichke.pdf (дата обращения 05.05.2021).

4. Шилдт Г. C++: 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).