

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.....	9
3.1 Алгоритм функции main.....	9
3.2 Алгоритм конструктора класса Class1.....	9
3.3 Алгоритм конструктора класса Class2.....	10
3.4 Алгоритм конструктора класса Class3.....	10
3.5 Алгоритм конструктора класса Class4.....	11
3.6 Алгоритм метода print класса Class1.....	11
3.7 Алгоритм метода print класса Class2.....	11
3.8 Алгоритм метода print класса Class3.....	12
3.9 Алгоритм метода print класса class4.....	12
4 БЛОК-СХЕМЫ АЛГОРИТМОВ.....	13
5 КОД ПРОГРАММЫ.....	16
5.1 Файл class1.cpp.....	16
5.2 Файл class1.h.....	16
5.3 Файл class2.cpp.....	17
5.4 Файл class2.h.....	17
5.5 Файл class3.cpp.....	18
5.6 Файл class3.h.....	18
5.7 Файл class4.cpp.....	18
5.8 Файл class4.h.....	19
5.9 Файл main.cpp.....	19
6 ТЕСТИРОВАНИЕ.....	21

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ.....	22
---------------------------------------	----

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 класса `class1` предназначен для ;
- объект класса `class2` предназначен для ;
- объект класса `class3` предназначен для ;
- объект класса `class4` предназначен для ;
- стандартная библиотека объектов ввода и вывода `iostream`;
- оператор присвоения `=`;
- оператор выделения памяти под новый объект `new`;
- оператор удаления динамически созданного объекта `delete`.

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: точка входа в программу.

Параметры: нет.

Возвращаемое значение: int.

Алгоритм функции представлен в таблице 1.

Таблица 1 – Алгоритм функции main

№	Предикат	Действия	№ перехода
1		объявление строки name, числа n	2
2		ввод name и n	3
3		инициализация указателя ptr новым объектом class4, присвоенным к class1	4
4		вызов print у ptr	5
5		вызов print у ptr, приведенный к class2	6
6		вызов print у ptr, приведенный к class3	7
7		вызов print у ptr, приведенный к class4	8
8		удаление ptr	9
9		возврат 0	Ø

3.2 Алгоритм конструктора класса Class1

Функционал: создание объекта.

Параметры: нет.

Алгоритм конструктора представлен в таблице 2.

Таблица 2 – Алгоритм конструктора класса Class1

№	Предикат	Действия	№ перехода
1		полю name присвоить n полю n присвоить n	Ø

3.3 Алгоритм конструктора класса Class2

Функционал: создание объекта, заполнение полей.

Параметры: строка name, целое n - имя и возводимое число.

Алгоритм конструктора представлен в таблице 3.

Таблица 3 – Алгоритм конструктора класса Class2

№	Предикат	Действия	№ перехода
1		полю name присвоить n полю n присвоить n	Ø

3.4 Алгоритм конструктора класса Class3

Функционал: создание объекта, заполнение полей.

Параметры: строка name, целое n - имя и возводимое число.

Алгоритм конструктора представлен в таблице 4.

Таблица 4 – Алгоритм конструктора класса Class3

№	Предикат	Действия	№ перехода
1		полю name присвоить n полю n присвоить n	Ø

3.5 Алгоритм конструктора класса Class4

Функционал: создание объекта, заполнение полей.

Параметры: строка name, целое n - имя и возводимое число.

Алгоритм конструктора представлен в таблице 5.

Таблица 5 – Алгоритм конструктора класса Class4

№	Предикат	Действия	№ перехода
1		полю name присвоить n полю n присвоить n	Ø

3.6 Алгоритм метода print класса Class1

Функционал: вывод значений полей.

Параметры: нет.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 6.

Таблица 6 – Алгоритм метода print класса Class1

№	Предикат	Действия	№ перехода
1		вывод name, " ", n, переход на новую строку	Ø

3.7 Алгоритм метода print класса Class2

Функционал: вывод значений полей.

Параметры: нет.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 7.

Таблица 7 – Алгоритм метода print класса Class2

№	Предикат	Действия	№ перехода
1		вывод name, " ", n, переход на новую строку	Ø

3.8 Алгоритм метода print класса Class3

Функционал: вывод значений полей.

Параметры: нет.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 8.

Таблица 8 – Алгоритм метода print класса Class3

№	Предикат	Действия	№ перехода
1		вывод name, " ", n, переход на новую строку	Ø

3.9 Алгоритм метода print класса class4

Функционал: вывод значений полей.

Параметры: нет.

Возвращаемое значение: нет.

Алгоритм метода представлен в таблице 9.

Таблица 9 – Алгоритм метода print класса class4

№	Предикат	Действия	№ перехода
1		вывод name, " ", n	Ø

4 БЛОК-СХЕМЫ АЛГОРИТМОВ

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

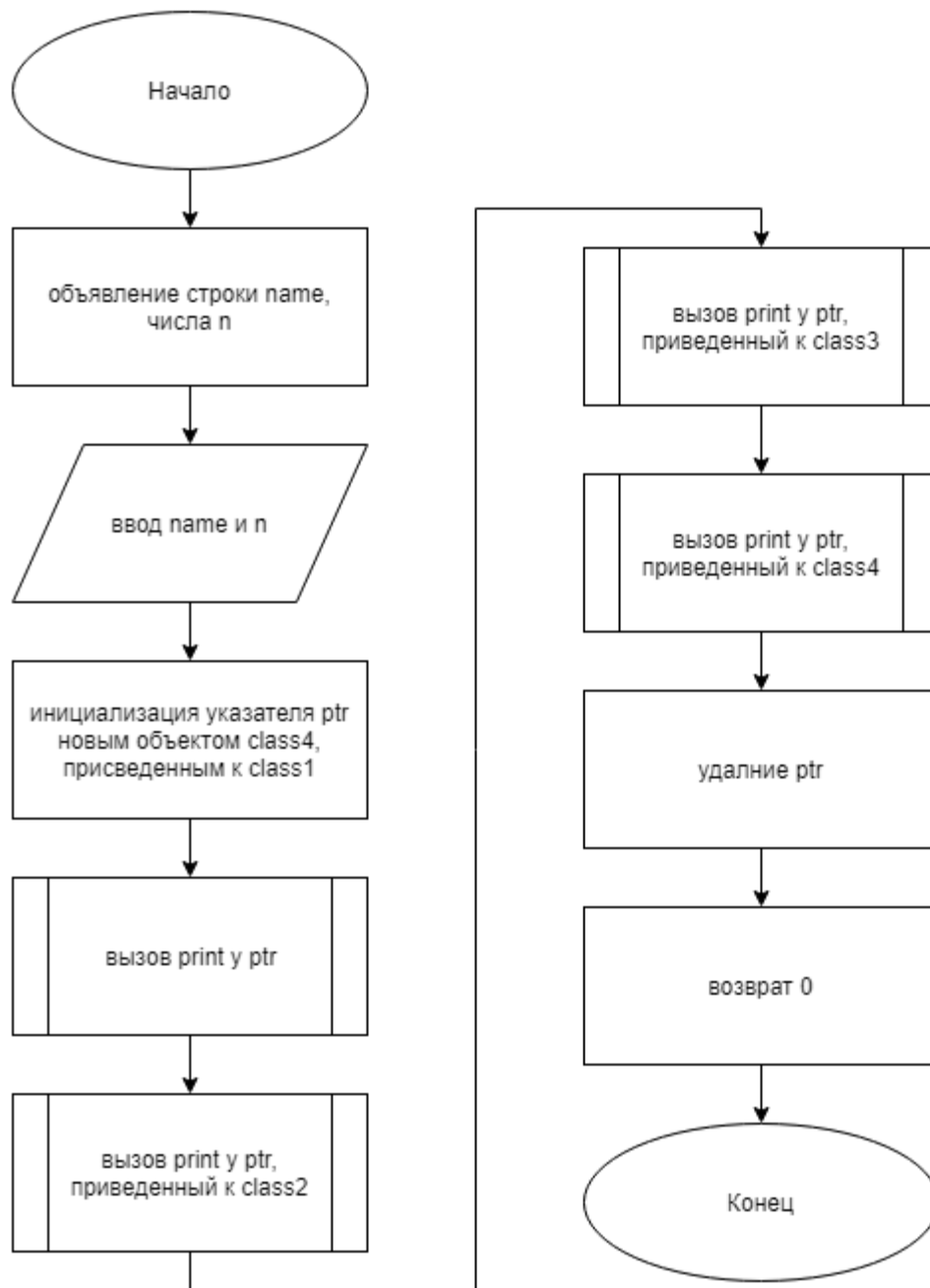


Рисунок 1 – Блок-схема алгоритма

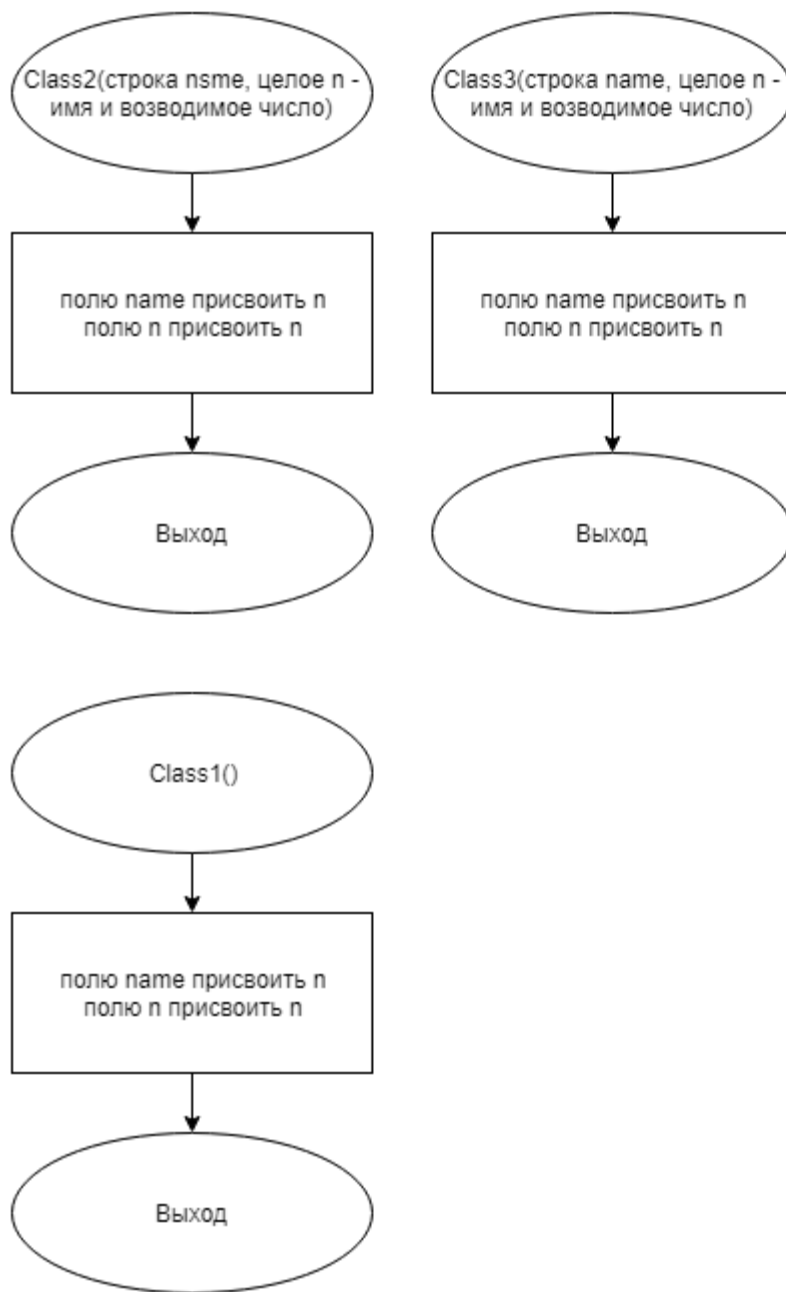


Рисунок 2 – Блок-схема алгоритма

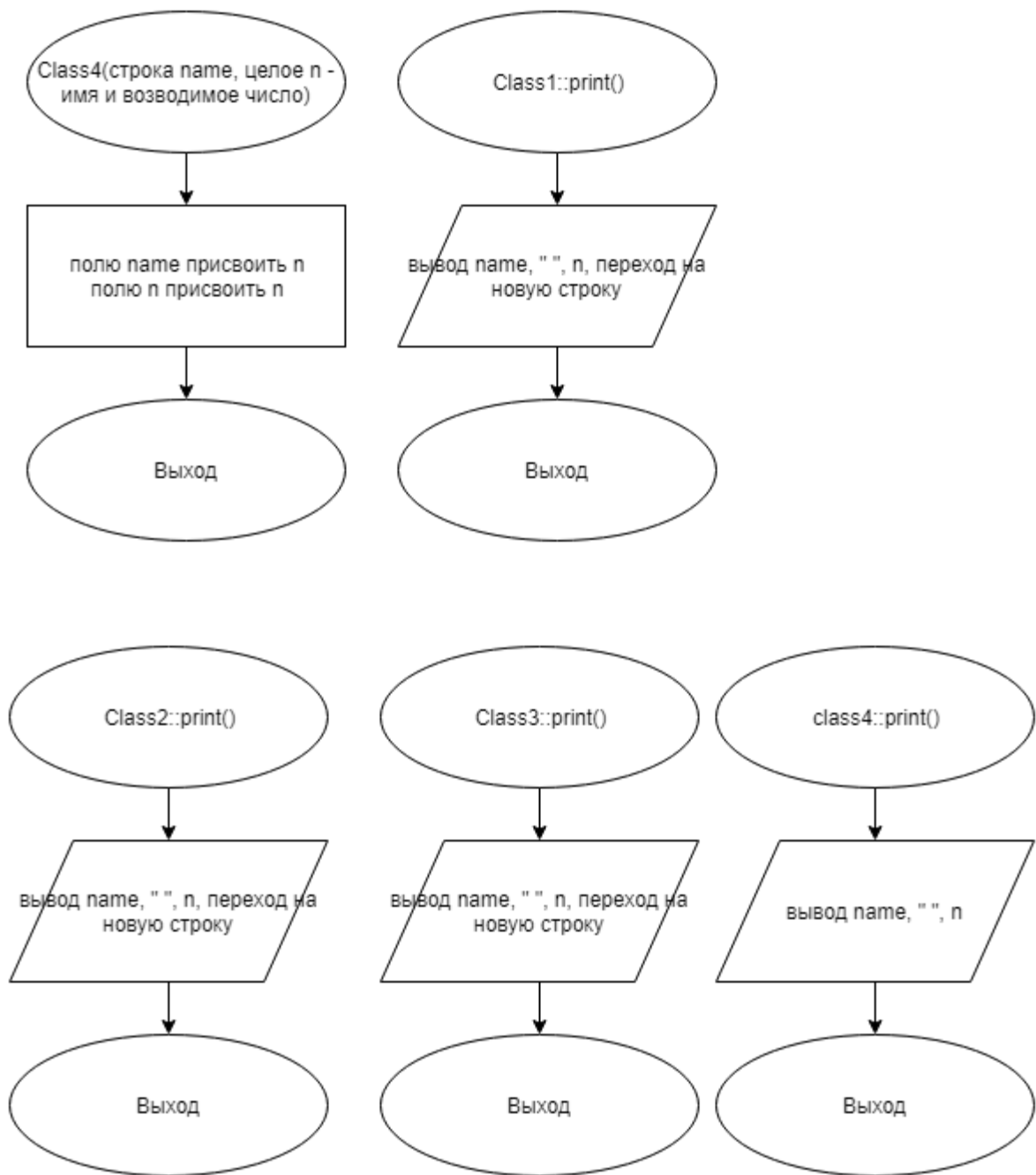


Рисунок 3 – Блок-схема алгоритма

5 КОД ПРОГРАММЫ

Программная реализация алгоритмов для решения задачи представлена ниже.

5.1 Файл class1.cpp

Листинг 1 – class1.cpp

```
#include "class1.h"

class1::class1(string name, int n)
{
    this->name = name + "_1";
    this->n = pow(n, 1);
}

void class1::print()
{
    cout << name << " " << n << "\n";
}
```

5.2 Файл class1.h

Листинг 2 – class1.h

```
#ifndef __CLASS__H
#define __CLASS__H

#include <string>
#include <math.h>
#include <iostream>
using namespace std;
class class1{
public:
    class1(string, int);
    void print();
private:
    string name;
    int n;
};
```

```
#endif
```

5.3 Файл class2.cpp

Листинг 3 – class2.cpp

```
#include "class2.h"

class2::class2(string name, int n):class1(name, n)
{
    this->name = name + "_2";
    this->n = pow(n, 2);
}

void class2::print(){
    cout << name << " " << n << "\n";
}
```

5.4 Файл class2.h

Листинг 4 – class2.h

```
#ifndef __CLASS2__H
#define __CLASS2__H
#include "class1.h"

class class2: private class1
{
public:
    class2(string, int);
    void print();
private:
    string name;
    int n;
};

#endif
```

5.5 Файл class3.cpp

Листинг 5 – class3.cpp

```
#include "class3.h"
class3::class3(string name, int n):class2(name, n)
{
    this->name = name + "_3";
    this->n = pow(n, 3);
}

void class3::print(){
    cout << name << " " << n << "\n";
}
```

5.6 Файл class3.h

Листинг 6 – class3.h

```
#ifndef __CLASS3__H
#define __CLASS3__H
#include "class2.h"

class class3: private class2
{
    string name;
    int n;
public:
    class3(string, int);
    void print();
};

#endif
```

5.7 Файл class4.cpp

Листинг 7 – class4.cpp

```
#include "class4.h"
class4::class4(string name, int n):class3(name, n)
{
    this->name = name + "_4";
    this->n = pow(n, 4);
}
```



```

}

void class4::print(){
    cout << name << " " << n;
}

```

5.8 Файл class4.h

Листинг 8 – class4.h

```

#ifndef __CLASS4__H
#define __CLASS4__H
#include "class3.h"

class class4: private class3
{
    string name;
    int n;
public:
    class4(string, int);
    void print();
};

#endif

```

5.9 Файл main.cpp

Листинг 9 – main.cpp

```

#include <stdlib.h>
#include <stdio.h>
#include "class4.h"
int main()
{
    string name;
    int n;
    cin >> name >> n;
    class1 *ptr = (class1*)new class4(name, n);
    ptr-> print();

    ((class2*)ptr)->print();
    ((class3*)ptr)->print();
    ((class4*)ptr)->print();
    delete ptr;
}

```

```
    return(0);  
}
```

6 ТЕСТИРОВАНИЕ

Результат тестирования программы представлен в таблице 10.

Таблица 10 – Результат тестирования программы

Входные данные	Ожидаемые выходные данные	Фактические выходные данные
оор 2	оор_1 2 оор_2 4 оор_3 8 оор_4 16	оор_1 2 оор_2 4 оор_3 8 оор_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).