Пермский национальный исследовательский политехнический университет ПНИПУ

Отчёт по лабораторной работе на тему “Классы и объекты. Инкапсуляция”

Выполнил студент группы РИС-23-3Б:

Смирнов Андрей Сергеевич

Проверила:

Доцент кафедры ИТАС

О.А. Полякова

2024

1. Постановка задачи:

1. Реализовать определение нового класса. Для демонстрации работы с объектами написать главную функцию. Продемонстрировать разные способы создания объектов и массивов объектов.

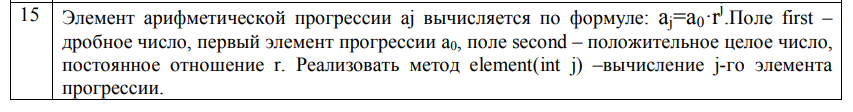
2. Структура-пара – структура с двумя полями, которые обычно имеют имена first и second. Требуется реализовать тип данных с помощью такой структуры. Во всех заданиях должны присутствовать:

a. метод инициализации Init (метод должен контролировать значения аргументов на корректность);

b. ввод с клавиатуры Read;

c. вывод на экран Show.

3. Реализовать внешнюю функцию make\_тип(), где тип – тип реализуемой структуры. Функция должна получать значения для полей структуры как параметры функции и возвращать структуру как результат. При передаче ошибочных параметров следует выводить сообщение и заканчивать работу.

2. Класс – структура с двумя переменными first и second, функцией pow для нахождения j-го элемента прогрессии. 

В классе созданы конструктор, деструктор, сеттеры и функция по запуску метода element.

3.

Реализовать внешнюю функцию element(int first, int second), где first и second – числа для нахождения поставленной задачи. Функция принимает числа как параметры и возвращает число как результат. При передаче ошибочных параметров следует выводить сообщение и заканчивать работу.

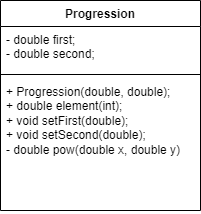
4. Класс:

Анализ:

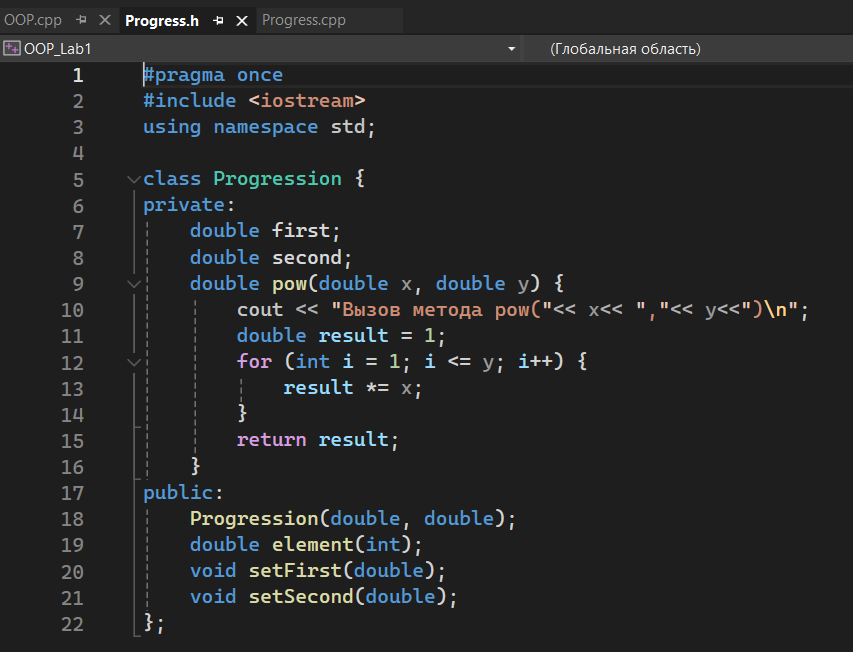
Создаем файл Progress.h в нем определяем класс его конструкторы и методы. Создаем файл Progress.cpp в нем описываем конструкторы и методы.

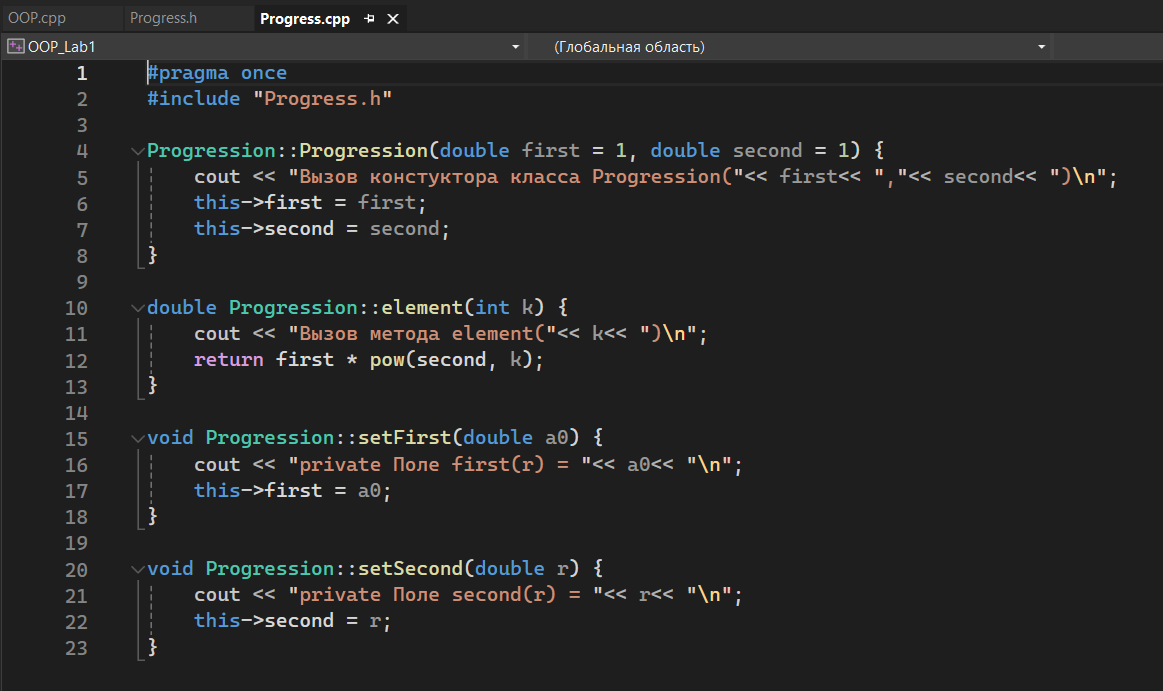
Создаем файл main.cpp и используем класс.

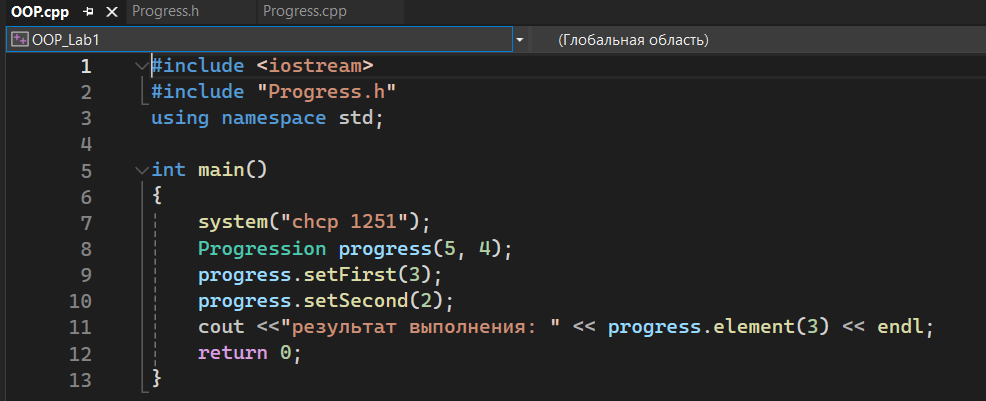
Диаграмма класса:



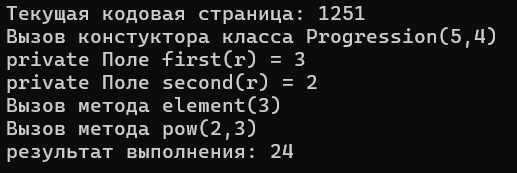
5. Код:







6. Результат работы:



7. Вывод:

Класс – структура, в которой можно хранить как функции, так и операторы.

Его можно использовать в других файлах, обратившись через оператор доступа.

8.Ответы на вопросы:

1. Что такое класс?

1. Это шаблон, который определяет структуру и поведение объектов в программе. Класс содержит данные (атрибуты) и функции (методы), которые определяют, какие операции могут выполняться над этими данными. Каждый объект в программе создается на основе класса и имеет свои собственные копии данных, но все объекты одного класса используют одни и те же методы.

2. Что такое объект (экземпляр) класса?

2. Сущность в цифровом пространстве, которая обладает состоянием и поведением, имеет поля и методы. Он создается на основе описания, заложенного в классе, и представляет собой конкретный объект в памяти. Объекты принадлежат одному или нескольким классам, которые определяют их поведение. Термины "экземпляр класса" и "объект" часто используются как взаимозаменяемые. Объекты обладают свойствами наследования, инкапсуляции и полиморфизма

‍‍

3. Как называются поля класса?

3. Поля класса называются атрибутами

4. Как называются функции класса?

4. Функции класса в программировании называются методами. Методы определяют поведение объектов класса и могут быть вызваны для выполнения определенных действий или операций.

5. Для чего используются спецификаторы доступа?

5. Спецификаторы доступа в программировании используются для управления видимостью и доступностью элементов класса (полей и методов) из других частей программы. Они позволяют контролировать, какие части кода могут взаимодействовать с определенными членами класса.

6. Для чего используется спецификатор public?

6. Элементы класса с таким уровнем доступа могут быть использованы любым кодом вне класса. Это означает, что они видимы и доступны для всех других классов и функций.

7. Для чего используется спецификатор private?

7. Элементы класса с этим уровнем доступа видны и доступны только внутри самого класса. Это обеспечивает максимальную инкапсуляцию, предотвращая внешнее вмешательство в работу класса.

8. Если описание класса начинается со спецификатора class, то какой спецификатор доступа будет использоваться по умолчанию?

По умолчанию будет использоваться private.

9. Если описание класса начинается со спецификатора struct, то какой спецификатор доступа будет использоваться по умолчанию?

По умолчанию будет использоваться public.

10. Какой спецификатор доступа должен использоваться при описании интерфейса класса? Почему?

Public, т.к. к ним нужно будет обращаться в любой части программы

11. Каким образом можно изменить значения атрибутов экземпляра класса?

1. Напрямую

2. Через ссылку

3. Через указатель

12. Каким образом можно получить значения атрибутов экземпляра класса?

1. Напрямую

2. Через public методы класса

3. Через указатель.

13. Класс описан следующим образом

struct Student {

string name;

int group;

………

};

Объект класса определен следующим образом Student \*s=new Student;

Как можно обратиться к полю name объекта s?

Ответ: s->name;

14. Класс описан следующим образом struct Student  
{  
string name;  
int group;  
};  
Объект класса определен следующим образом Student s;  
Как можно обратиться к полю name объекта s?

Ответ: s.name;

15. Класс описан следующим образом  
class Student  
{  
string name;  
int group;  
};  
Объект класса определен следующим образом Student \*s=new Student;  
Как можно обратиться к полю name объекта s?

Ответ: через public методы:

Void showName () {

Cout<<name;

}

Main(){

Student\*s = new Student;

s->showName();

}

16. Класс описан следующим образом  
class Student  
{  
string name;  
int group;

public:

};  
Объект класса определен следующим образом Student s;  
Как можно обратиться к полю name объекта s?

Struct Student{

String name;

Int group;

}

Void showName () {

Cout<<name;

}

Main () {

Student s;

s.showName ();

}

17. Класс описан следующим образом  
class Student  
{  
public:  
char\* name;  
int group;  
};  
Объект класса определен следующим образом Student \*s=new Student;  
Как можно обратиться к полю name объекта s?

Student s;

s.name;

s.group;