Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы» направление подготовки: 09.03.04 – «Программная инженерия»

**Лабораторная работа №5.**

**«Наследование. Виртуальные функции. Полиморфизм»**

Выполнил студент гр. РИС-24-2б

Молочко Артём Анатольевич

Проверил:

Доц. Каф. ИТАС

Ольга Андреевна Полякова

(оценка) (подпись)

(дата)

г. Пермь, 2025

**Постановка задачи**

1. Создать абстрактный базовый класс
2. Разработать иерархию классов, наследующих от абстрактного класса (согласно лабораторной работе №4)
3. Реализовать класс Vector, содержащий указатели на объекты иерархии классов
4. Перегрузить оператор вывода (<<) для класса Vector для работы с потоками
5. В функции main продемонстрировать:
   * Работу перегруженных операторов
   * Полиморфное поведение объектов через Vector

Базовый класс:  
ЧЕЛОВЕК (PERSON)  
Имя (name) – string  
Возраст (age) – int  
Определить методы изменения полей.  
Создать производный класс STUDENT, имеющий поля Предмет – string и Оценка – int. Определить методы изменения полей и метод, выдающий сообщение о неудовлетворительной оценке.

**Анализ задачи**

**1. Архитектура программы**  
Программа реализует иерархию классов с использованием:

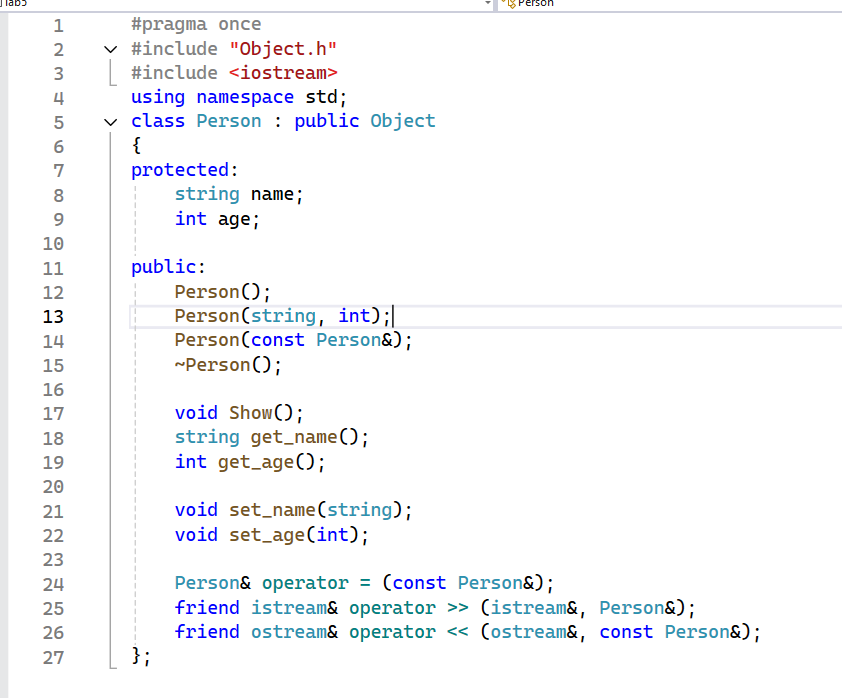
* Абстрактного базового класса Object
* Производного класса Person
* Производного класса Student (наследуется от Person)
* Класса Vector для хранения объектов

**2. Реализация полиморфизма**

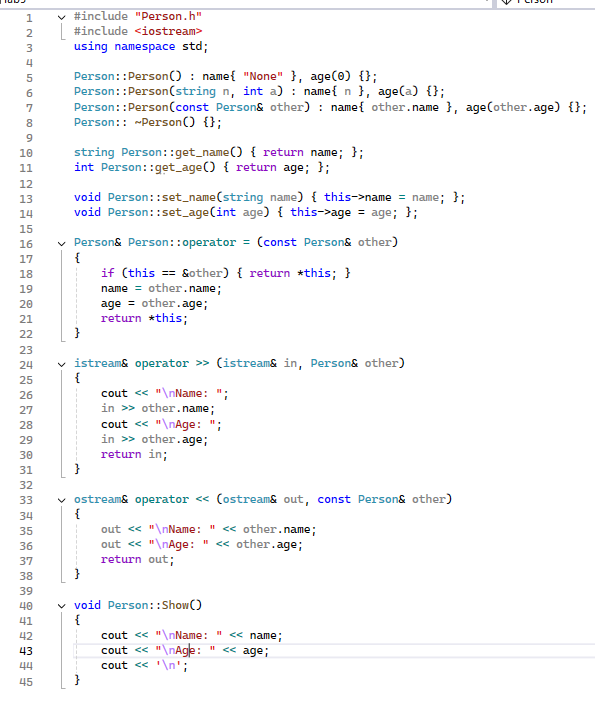
* Используется абстрактный класс Object с чисто виртуальной функцией Show()
* Все производные классы переопределяют метод Show()
* Вектор работает с указателями на Object, что позволяет хранить разные типы объектов
* Функция print демонстрирует полиморфное поведение через указатель на базовый класс

**Код программы**

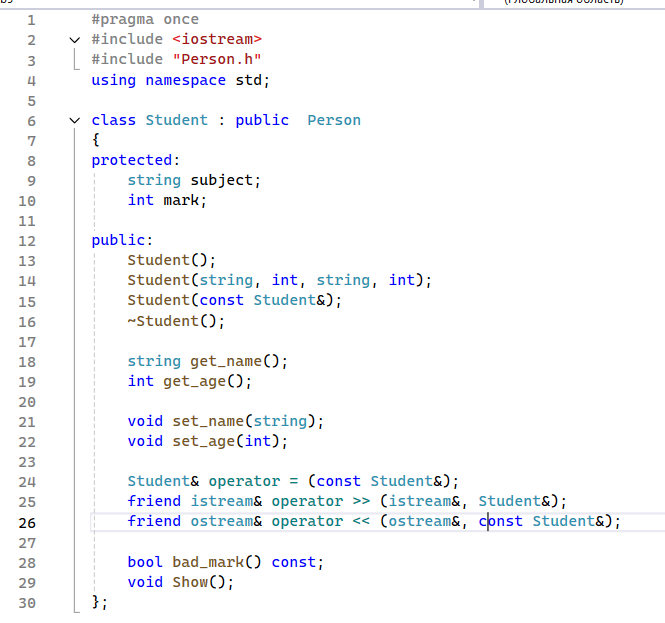
**Person.h**



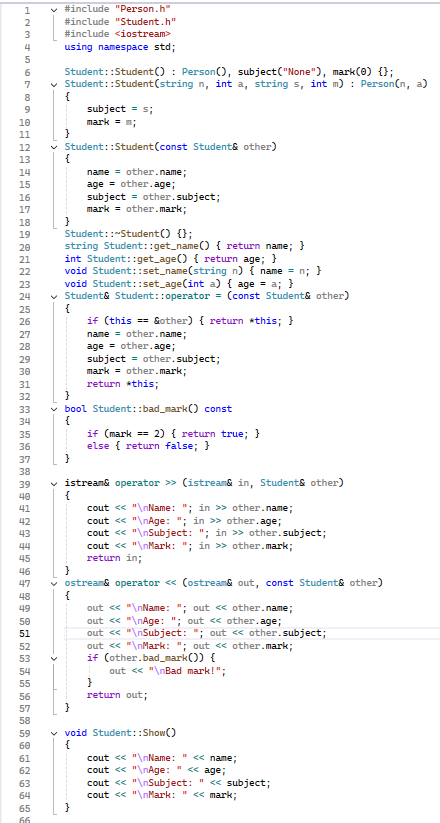
**Person.cpp**



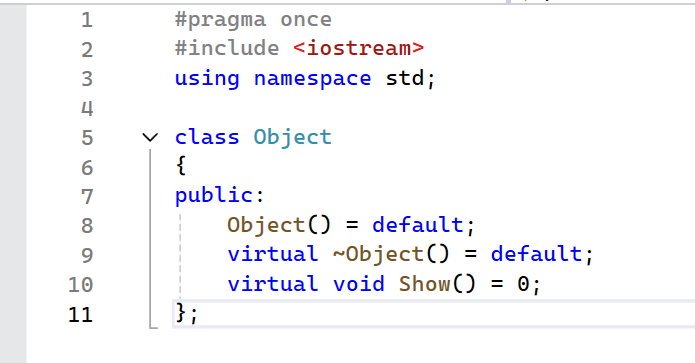
**Student.h**



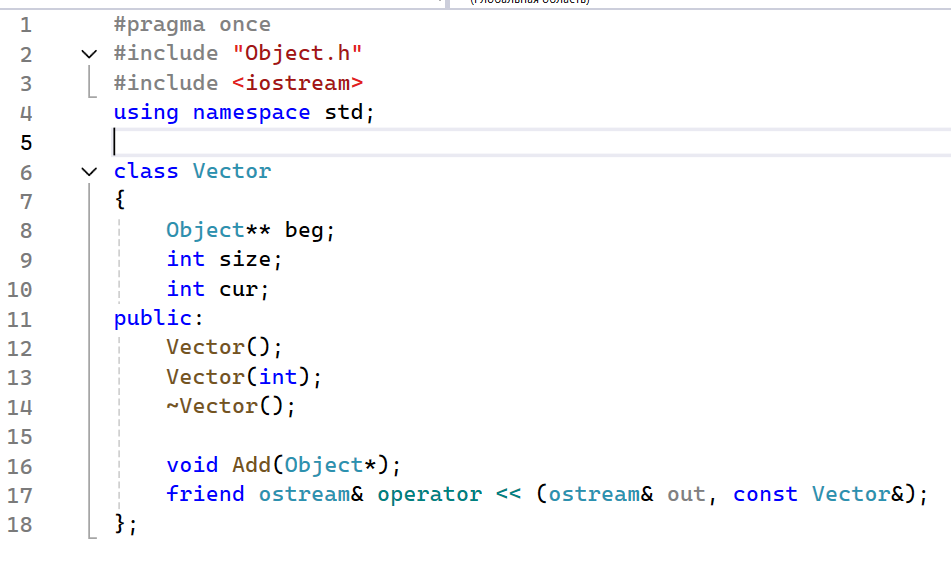
**Student.cpp**



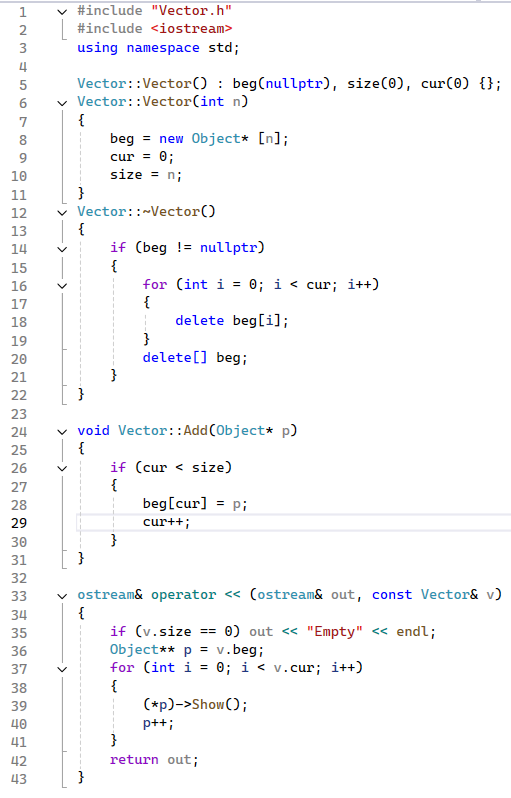
**Object.h**



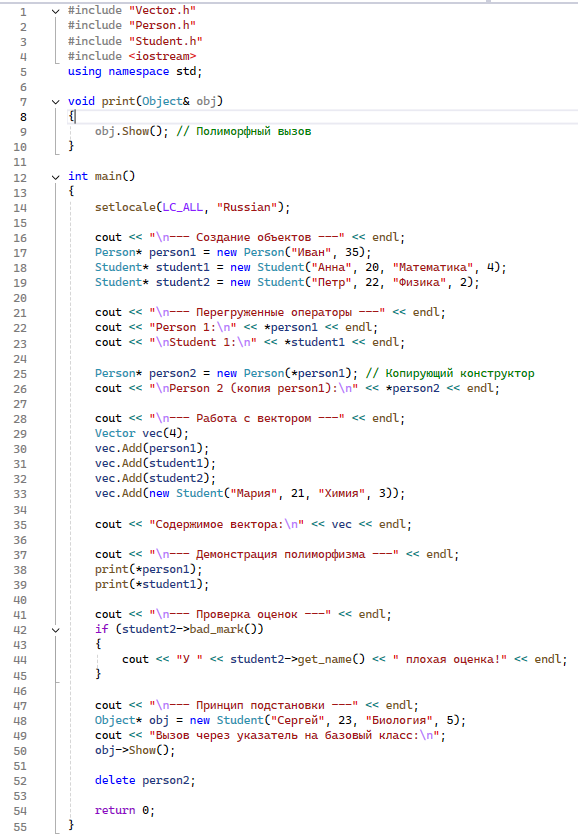
**Vector. h**



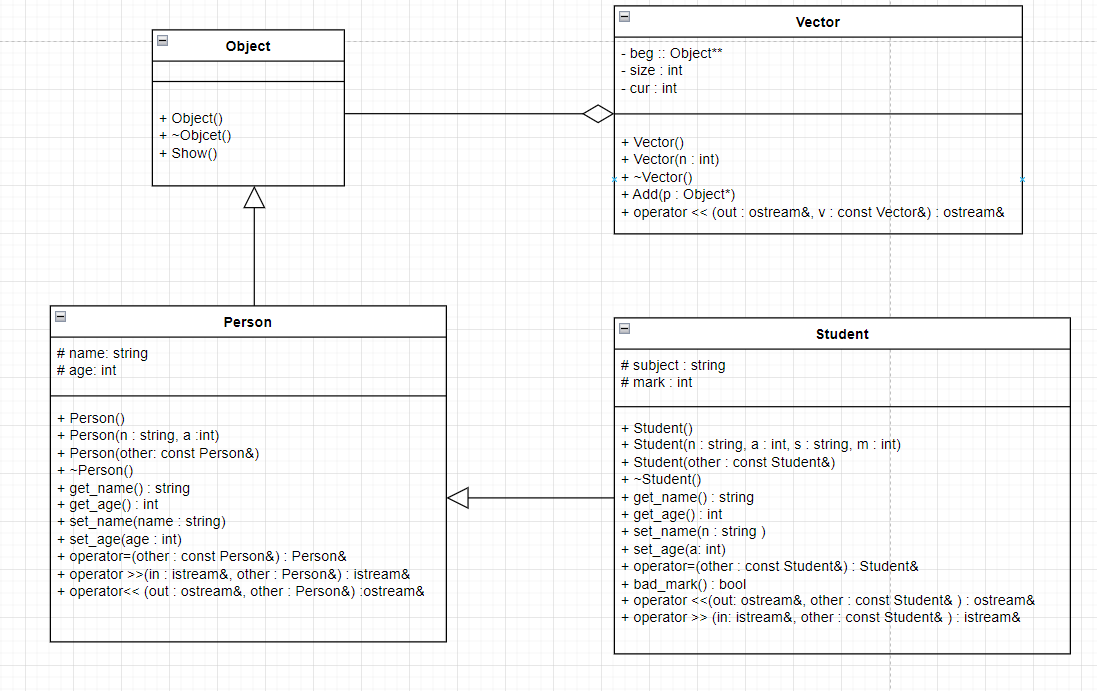
**Vector.cpp**



**main.cpp**



**Диаграмма классов**



**Контрольные вопросы**

**1. Какой метод называется чисто виртуальным? Чем он отличается от виртуального метода?**  
Ответ: Чисто виртуальный метод объявляется с "= 0" и не имеет реализации в базовом классе. Обычный виртуальный метод может иметь реализацию.

**2. Какой класс называется абстрактным?**  
Ответ: Класс, содержащий хотя бы один чисто виртуальный метод. Нельзя создать объект такого класса.

**3. Для чего предназначены абстрактные классы?**  
Ответ: Для создания интерфейсов и базовой функциональности, которую должны реализовать производные классы.

**4. Что такое полиморфные функции?**  
Ответ: Функции, которые могут работать с разными типами объектов через единый интерфейс.

**5. Чем полиморфизм отличается от принципа подстановки?**  
Ответ: Полиморфизм - общая концепция, принцип подстановки - конкретное правило, что объекты производного класса могут заменять объекты базового.

**6. Привести примеры иерархий с использованием абстрактных классов.**  
Ответ:

1. Фигуры: Shape → Circle, Square
2. Транспорт: Vehicle → Car, Plane

**7. Привести примеры полиморфных функций.**  
Ответ:

1. Функция вывода площади фигуры через виртуальный метод area()
2. Функция сохранения документа через виртуальный метод save()
3. Функция отрисовки графических объектов через draw()

**8. В каких случаях используется механизм позднего связывания?**  
Ответ: При вызове виртуальных методов, когда конкретная реализация выбирается во время выполнения программы.