KМинистерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учереждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

Направление подготовки: Разработка информационных систем (РИС)

**Лабораторная работа 4**

Выполнил студент гр. РИС-24-3б

Караваев Артем Андреевич

Проверил:

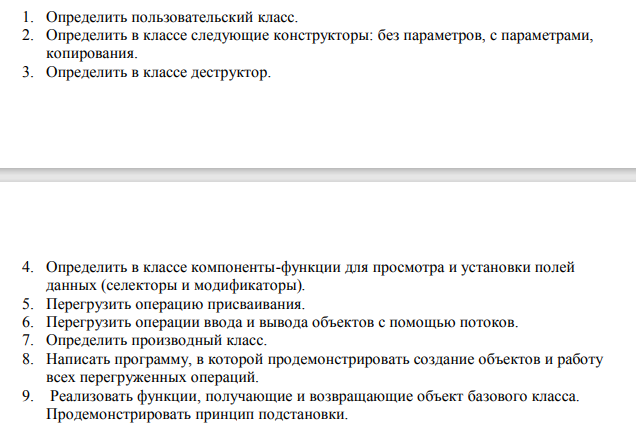
Доц. каф. ИТАС

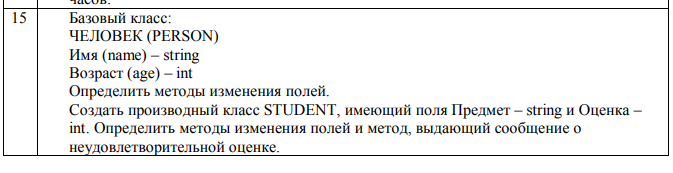
Ольга Андреевна Полякова

г. Пермь, 2024

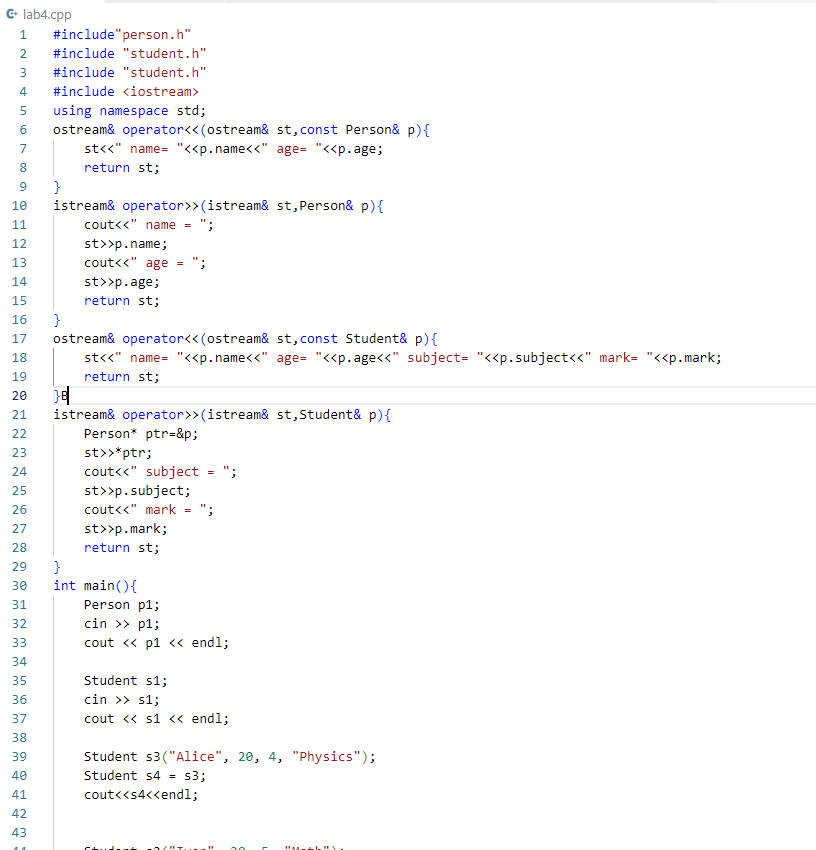
**Постановка задачи.**

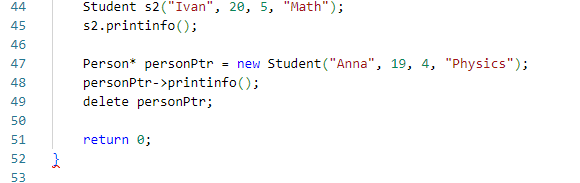
15 вариант:

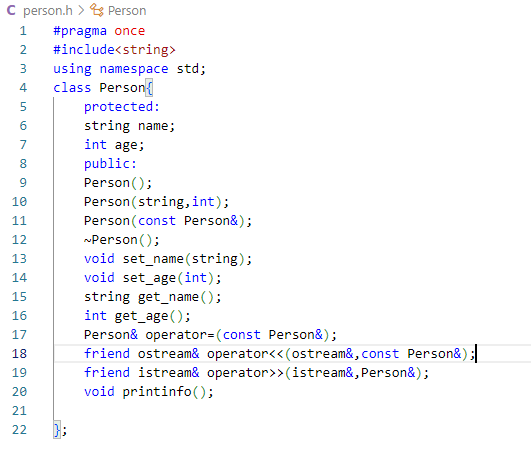


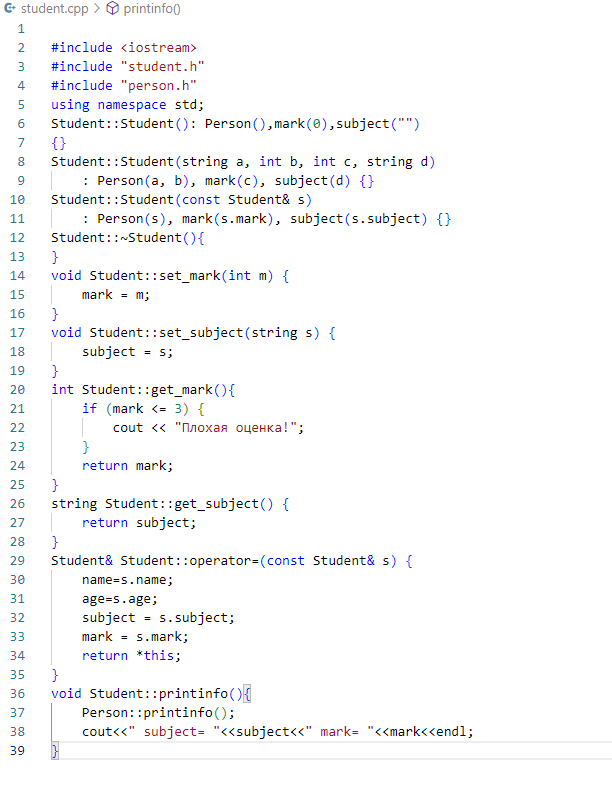
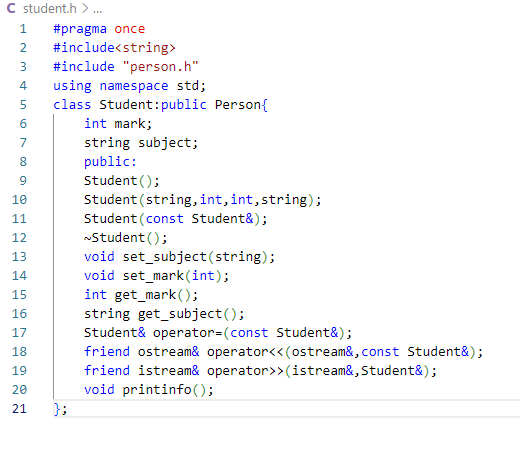
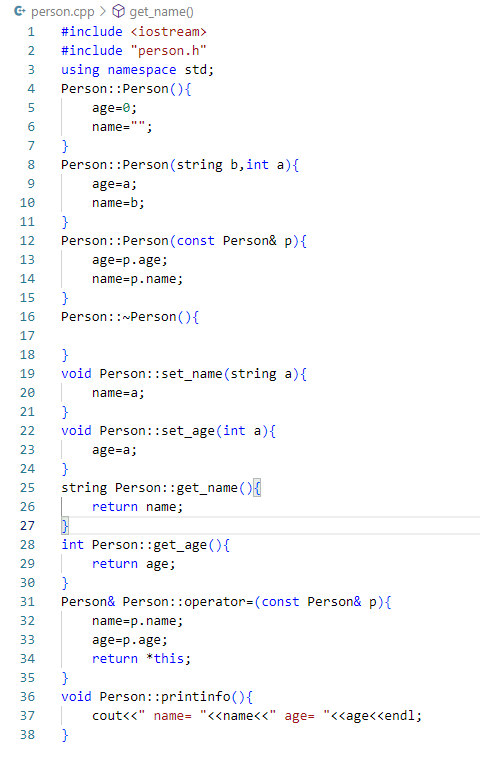


Программа на с++:

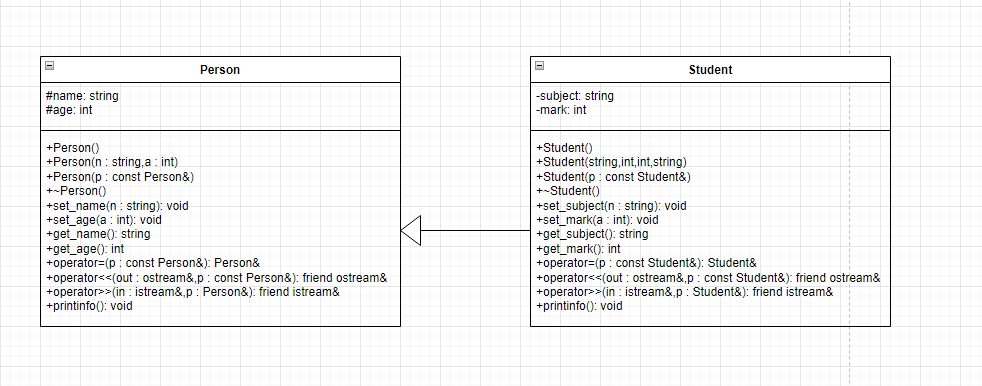


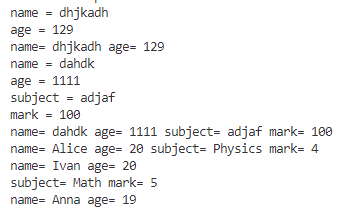


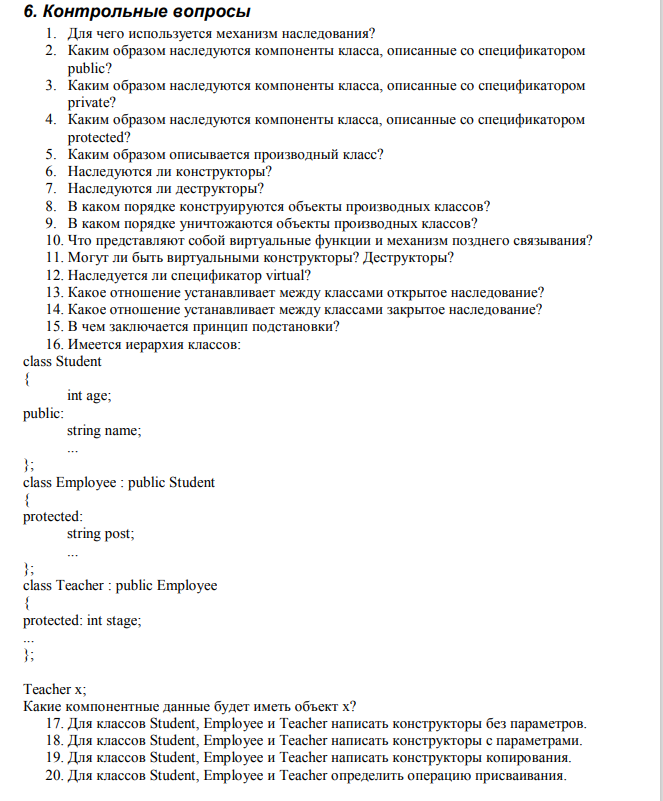




UML-диаграмма:

Входные и выходные данные:





1. Для создания иерархии классов, где производные классы получают свойства и методы базовых классов, для реализации полиморфизм.
2. При наследовании по умолчанию public-члены базового класса становятся private в производном классе.
3. При наследовании по умолчанию private-члены базового класса становятся недоступны
4. При наследовании по умолчанию protected-члены базового класса становятся private в производном классе.
5. class Derived : спецификатор\_доступа Base {}
6. Нет
7. Нет, но деструктор базового класса автоматически вызывается после деструктора производного.
8. **С**начала базовые классы, затем члены производного класса и его функции
9. В обратном порядке
10. Виртуальные функции позволяют переопределять поведение в производных классах. Позднее связывание - выбор реализации функции во время выполнения.
11. Конструкторы не могут быть виртуальными, деструкторы должны быть виртуальными в базовом классе
12. Если функция объявлена виртуальной в базовом классе, она остаётся виртуальной во всей иерархии.
13. Устанавливает отношение "является" (is-a). Интерфейс базового класса сохраняется.
14. Устанавливает отношение "реализовано посредством" (implemented-in-terms-of). Все члены базового становятся private.
15. Объекты производного класса должны быть заменяемы на объекты базового без изменения корректности программы.
16. string name (public из Student), string post (protected из Employee), int stage (protected из Teacher)

17-20.

class Student {

int age;

public:

string name

// Конструктор без параметров

Student() : age(0), name("") {}

// Конструктор с параметрами

Student(int a, const string& n) : age(a), name(n) {}

// Конструктор копирования

Student(const Student& other) : age(other.age), name(other.name) {

// Оператор присваивания

Student& operator=(const Student& other) {

age = other.age;

name = other.name

return \*this;

}

};

class Employee : public Student {

protected:

string post;

public:

// Конструктор без параметров

Employee() : Student(), post("") {}

// Конструктор с параметрами

Employee(int a, const string& n, const string& p)

: Student(a, n), post(p) {}

// Конструктор копирования

Employee(const Employee& other)

: Student(other), post(other.post) {}

// Оператор присваивания

Employee& operator=(const Employee& other) {

Student::operator=(other);

post = other.post;

return \*this;

}

};

class Teacher : public Employee {

protected:

int stage;

public:

// Конструктор без параметров

Teacher() : Employee(), stage(0) {

// Конструктор с параметрами

Teacher(int a, const string& n, const string& p, int s)

: Employee(a, n, p), stage(s) {}

// Конструктор копирования

Teacher(const Teacher& other)

: Employee(other), stage(other.stage) {}

// Оператор присваивания

Teacher& operator=(const Teacher& other) {

Employee::operator=(other);

stage = other.stage;

return \*this;

}

};

https://github.com/Prefix008/lab.git