

1) Em um sistema para gestão de um restaurante, existem classes para gerenciar funcionários, cozinheiros e garçons. A classe funcionario possui atributos como nome, CPF e endereço. A classe cozinheiro herda de funcionario e possui atributos como especialidade e número\_de\_receitas. A classe garcom também herda de funcionario e possui atributos como horário\_de\_trabalho e área\_designada. A classe pedido gerencia os pedidos feitos pelos clientes e contém atributos como data, garcom e cliente.

a) Qual tipo de herança você usaria para as classes funcionario, cozinheiro e garcom? Explique como a relação aconteceria e justifique sua resposta.

b) Sabendo que o sistema precisa calcular a idade dos funcionários, qual modificador de acesso você usaria na classe funcionario para armazenar a data de nascimento, permitindo acesso direto nas classes derivadas (cozinheiro e garcom), mas sem expor esse dado publicamente? Justifique.

2. Considere uma hierarquia de classes para modelar diferentes formas geométricas e animais em C++. A hierarquia inclui as seguintes classes:

- Shape: Uma classe base que representa uma forma geométrica. Ela possui um método virtual chamado `area()` que pode ser sobrescrito pelas classes derivadas.
- Circle: Uma classe derivada de `Shape` que representa um círculo. Ela tem um atributo `radius` e sobrescreve o método `area()` para calcular a área do círculo.
- Rectangle: Outra classe derivada de `Shape` que representa um retângulo. Ela possui atributos `width` e `height` e também sobrescreve o método `area()` para calcular a área do retângulo.
- Animal: Uma classe base que representa um animal. Ela possui um método virtual puro chamado `speak()`, que deve ser implementado pelas classes derivadas.
- Dog: Uma classe derivada de `Animal` que representa um cachorro. Ela implementa o método `speak()` para fazer o cachorro latir.
- Cat: Outra classe derivada de `Animal` que representa um gato. Ela implementa o método `speak()` para fazer o gato miar.

a) Explique os conceitos de **função virtual** e **função virtual pura** em C++ e forneça exemplos de uso desses conceitos na hierarquia de classes acima.

b) Explique brevemente o que significa ter um atributo ou método como **estático** (*static*) em C++ e dê um exemplo de utilização de um atributo e um método estático (na classe Shape ou na classe Animal).

3. Complete o código abaixo:

```
class midia {
    __ int total_midias; // a)
    Complete para contar mídias
public:
    midia() { total_midias++; }
    virtual ~midia() {}

    __ void reproduzir() __ //
    b) Complete para criar um método
    virtual puro

    __ int get_total() { __ } //
    c) Complete para obter o número
    total de mídias
};

int midia::total_midias = 0;
```

```
class audio : __ midia { // d)
    Complete para herdar corretamente
public:
    void reproduzir() override {
        // e) Implemente para exibir
        mensagem
    }
};

class video : __ midia { // f)
    Complete para herdar corretamente
public:
    void reproduzir() override {
        // g) Implemente para exibir
    }
};

class imagem : __ midia { // h)
    Complete para herdar corretamente
public:
    // i) Implemente o método virtual
    puro reproduzir()
};
```