

Lista de Exercícios – Programação Orientada a Objetos em C++

Instruções Gerais

- Leia cada situação descrita e identifique as possíveis classes, atributos e métodos.
- Determine os relacionamentos entre as classes, diferenciando quando deve ser **composição** e quando deve ser **agregação**.
- Implemente em C++ com **construtores**, **destrutores** e um **método main** de teste.
- Onde houver agregação, utilize ponteiros ou referências. Onde houver composição, utilize objetos por valor ou `unique_ptr`.

Exercícios

- Exercício 1:** Uma pequena biblioteca deseja informatizar seu acervo. Ela guarda vários livros, cada um com informações básicas como título, autor e ano de publicação. O sistema deve permitir registrar os livros e listá-los. Identifique as classes e implemente.
- Exercício 2:** Um usuário de música online pode criar playlists, onde cada playlist contém várias músicas. Um mesmo usuário pode ter diversas playlists associadas ao seu perfil. Modele as classes necessárias e implemente.
- Exercício 3:** Uma loja virtual registra pedidos de clientes. Cada pedido possui itens (com produto, quantidade e preço unitário) e está vinculado a um cliente. O sistema deve calcular o valor total do pedido. Descubra as classes e implemente.
- Exercício 4:** Em uma clínica médica, cada paciente tem um prontuário contendo diversas anotações médicas feitas por um médico responsável. O prontuário deve estar vinculado a um paciente e a um médico. Identifique e implemente as classes envolvidas.
- Exercício 5:** Uma instituição de ensino organiza cursos, que possuem várias turmas. Cada turma é conduzida por um professor e contém vários alunos matriculados. Represente esse cenário com classes em C++.
- Exercício 6:** Uma empresa de transporte possui uma frota de ônibus, cada um com sua rota. As rotas possuem várias paradas. A empresa também mantém motoristas associados à frota. Elabore as classes e implemente.
- Exercício 7:** Em um jogo, cada jogador possui um inventário que guarda itens e armas. Além disso, há inimigos que podem estar equipados com armas. Estruture as classes necessárias e implemente.
- Exercício 8:** Em um sistema de e-commerce, o cliente adiciona produtos em um carrinho de compras. Esse carrinho é usado para gerar um pedido, que inclui informações de pagamento. Monte as classes necessárias e implemente.

Template de Código em C++

```
#include <iostream>
#include <vector>
#include <memory>
using namespace std;

class Livro {
    string titulo;
public:
    Livro(const string& t) : titulo(t) {
        cout << "Livro(' " << titulo << "') criado\n";
    }
    ~Livro() { cout << "Livro(' " << titulo << "')
        destruido\n"; }
    const string& getTitulo() const { return titulo; }
};

class Biblioteca {
    string nome;
    vector<Livro> acervo; // COMPOSIÇÃO
public:
    Biblioteca(const string& n) : nome(n) {
        cout << "Biblioteca(' " << nome << "') criada\n";
    }
    ~Biblioteca() { cout << "Biblioteca(' " << nome << "')
        destruida\n"; }

    void adicionarLivro(const Livro& l) { acervo.
        push_back(l); }

    void listar() const {
        cout << "Acervo de " << nome << ":\n";
        for (const auto& l : acervo) cout << " - " << l.
            getTitulo() << "\n";
    }
};

int main() {
    {
```

```
        Biblioteca bib("Central");
        Livro l1("POO em C++");
        Livro l2("Estruturas de Dados");
        bib.adicionarLivro(l1);
        bib.adicionarLivro(l2);
        bib.listar();
    } // observe a ordem de destrui o no console
    return 0;
}
```