

# 1 - Sistema de Funcionários de uma Empresa

Discente: VINICIUS DE SOUZA MENDES

## Descrição Geral do Problema

Você deve desenvolver um sistema para calcular salários de diferentes tipos de funcionários. Todos os funcionários têm nome, matrícula e salário base. No entanto, **cada cargo calcula seu salário de forma diferente**, exigindo uso de polimorfismo.

## Classes obrigatórias

- Funcionário (superclasse)
  - Atributos: nome, matricula, salarioBase
  - Método abstrato: double calcularSalario()
- Gerente
  - Ganha: salário base + bônus fixo de R\$ 1.500,00
- Vendedor
  - Ganha: salário base + comissão de 10%
- Estagiario
  - Ganha: apenas o salário base (sem benefícios)

## O que o programa deve fazer

- Criar uma lista contendo vários tipos de funcionários (polimórfica).
- Exibir para cada funcionário:
  - Nome
  - Cargo
  - Salário final (chamando calcularSalario())

# 2 - Zoológico: Cadastro e Comportamento dos Animais

Discente: BRAZ AMORIM CAMPOS

## Descrição Geral

Crie um sistema para administrar animais de um zoológico. Todos os animais têm nome, espécie e idade, mas cada um **emite um som diferente** e possui alimentação específica.

## Classes obrigatórias

- Animal (classe abstrata)
  - Métodos abstratos: emitirSom(), alimentar()
- Leao
- Elefante
- Macaco

## O que o programa deve fazer

- Armazenar todos os animais em uma lista polimórfica.
- Executar os métodos de todos os animais:
  - emitirSom()
  - alimentar()

# **3- Sistema de Pagamentos**

Discente: JOAO PAULO ALVES CAMPOS

## **Descrição Geral**

Desenvolver um sistema que permita processar diferentes tipos de pagamentos. Cada forma de pagamento possui regras próprias.

## **Classes obrigatórias**

- Pagamento (superclasse)
  - atributo valor
  - método abstrato processar()
- PagamentoPix — confirma imediatamente
- PagamentoCartao — valida limite e “aprova”
- PagamentoBoleto — tem prazo de compensação

## **O que o programa deve fazer**

- Processar uma lista de pagamentos diversos (usando polimorfismo).
- Exibir mensagens apropriadas para cada tipo de pagamento.

# **4 - Loja de Veículos**

Discente: GABRIEL PIVETTA LOSS

## **Descrição Geral**

Criar um sistema para simular veículos com comportamentos diferentes ao acelerar e frear.

## **Classes obrigatórias**

- Veiculo (superclasse)
  - Métodos abstratos: acelerar(), frear(), descricao()
- Carro
- Moto
- Caminhao

## **O que o programa deve fazer**

- Colocar todos os veículos em uma lista.
- Mostrar a descrição e acelerar/frear cada veículo.

# 5 - Sistema de Biblioteca

Discente: VITORIA CALONGA DOS SANTOS

## Descrição Geral

Criar um sistema que gerencie itens emprestáveis numa biblioteca.

## Classes obrigatórias

- ItemBiblioteca (superclasse)
  - Métodos abstratos:
    - emprestar()
    - devolver()
    - Multar ()
- Livro
- Revista
- MidiaDigital
- Jornal
- Artigo

## O que o programa deve fazer

- Criar lista com vários tipos de itens.
- Simular empréstimos e devoluções e multas.

# 6 - Jogo de RPG

Discente: JOAO PEDRO ZAGONEL

## Descrição Geral

Criar personagens com habilidades diferentes. O método atacar(), defender() e fugir() deve ser polimórfico.

## Classes obrigatórias

- Personagem
  - atributos: nome, vida, atk, dfs
  - métodos abstratos: atacar(), defender() e fugir()
- Guerreiro
- Mago
- Arqueiro
- Sacerdote
- Druida
- Necromante
- Bardo
- Monge
- Paladino
- Ladino

## O que o programa deve fazer

- Simular uma rodada de combate:
  - cada personagem atacando
  - cada personagem defendendo
  - cada personagem fugindo
  - Se criar o boss ganha mais 2 pontos no trabalho.

# 7 - Formas Geométricas

Discente: PEDRO JOAO PAULINO FRANZ

## Descrição Geral

Criar formas que calculam área de maneira diferente.

## Classes obrigatórias

- Forma (abstrata)
  - Método abstrato: calcularArea()
- Circulo
- Quadrado
- Retangulo
- Triangulo
- Losangulo
- Hipercubo
- Paraboloides
- Trapézio

## O que o programa deve fazer

- Criar lista de TODAS as formas.
- Calcular e exibir a soma total das áreas de todas as formas.

# **8 - Sistema de Transporte Público**

Discente: GREGORIO TAVARES DE MATOS

## **Descrição Geral**

Você deve criar um simulador de transporte urbano onde cada modalidade possui tarifa diferente.

## **Classes obrigatórias**

- Transporte (superclasse)
  - Método abstrato: calcularTarifa()
- Onibus
- Metro
- Taxi
- Trem
- Bicicleta (estilo bradesco)
- Patinete
- Avião da FAB

## **O que o programa deve fazer**

- Criar lista com transportes diversos.
- Mostrar o valor da tarifa de cada um.

## **Modificações obrigatórias**

1. Criar a classe Horario e associar a cada transporte.

# **9 - Gerenciamento de Arquivos**

Discente: JOAO MARCELLO GALDINO PEREIRA

## **Descrição Geral**

Criar um sistema que **simula** abertura e fechamento de arquivos.

## **Classes obrigatórias**

- Arquivo (superclasse)
  - Métodos abstratos: abrir(), fechar(), estimativa()
- ArquivoTexto
- Arquivolmagem
- ArquivoAudio
- ArquivoBytecode
- ArquivoMorse
- ArquivoASCII
- ArquivoNotaçãoAlgebricaXadrez

## **O que o programa deve fazer**

- Criar lista polimórfica de arquivos.
- Abrir, fechar e estimativa para todos os arquivos.

# **10 - Loja Online – Produtos**

Discente: EDUARDO PRUDENCIO FONTENELE

## **Descrição Geral**

Criar um sistema de produtos, onde cada um calcula seu preço final de forma diferente.

## **Classes obrigatórias**

- Produto (superclasse)
  - atributos: nome, precoBase, freteBase
  - método abstrato: calcularPrecoFinal()
- ProdutoEletronico
- ProdutoVestuario
- ProdutoAlimento
- ProdutoConstrucao
- ProdutoAutomativo
- ProdutoHigienico

## **O que o programa deve fazer**

- Criar carrinho de compras (lista).
- Exibir preço final de cada produto.
- Calcular Frete para tipo de produto.

# **11 - Sistema de Mensagens**

Discente: PEDRO AUGUSTO NOBRE SALES

## **Descrição Geral**

Criar um sistema que envia diferentes tipos de mensagens através de métodos polimórficos.

## **Classes obrigatórias**

- Mensagem (superclasse)
  - atributos: destinatario, conteudo, remetente
  - método abstrato: enviar(), NaoRecebido()
- MensagemEmail
- MensagemSMS
- MensagemWhatsApp
- MensagemTelegrama
- MensagemCorreio
- MensagemFax
- MensagemMorse

## **O que o programa deve fazer**

- Criar lista polimórfica com vários tipos de mensagens.
- Enviar todas usando polimorfismo.
- Gerar casos de não recebimento ou não encontrado