

**ECOP13 – Laboratório de Programação Orientada a Objetos**

**Prof. João Paulo R. R. Leite ([joaopaulo@unifei.edu.br](mailto:joaopaulo@unifei.edu.br))**

**Atividade 09 – Herança e Polimorfismo**

As atividades devem ser feitas em arquivos “.CPP” e “.H”. Os nomes dos arquivos “.CPP” principais de cada exercício estão indicados (ex01.cpp, etc.). Ao final, coloque todos em um mesmo **arquivo com extensão “.ZIP”**, que você deverá enviar na tarefa “Laboratório 09” cadastrada no SIGAA até o dia 19/11/2021 às 10:10h. **Não serão aceitas entregas via e-mail ou atrasadas.** A entrega valerá nota e presença. Os exercícios da semana têm como objetivo principal fazer com que o aluno continue praticando os conceitos de **Herança e Polimorfismo**. Siga as instruções:

1. **(ex01.cpp)** Crie um programa em C++ para simular a operação de um banco simplificado. Seu sistema deverá ter 3 classes: **Cliente**, **Conta** e **ContaEspecial**.
  - A classe Cliente tem como atributos o **nome** (*string*), a **idade** (*int*) e o **endereço** (*string*). A classe deve ter um único **construtor**, que recebe os três valores, e contém valores *default* (caso não sejam especificados). Além disso, a classe possui uma função membro que **imprime** as características do cliente.
  - A classe Conta tem como atributos um **número** (*int*), uma **agência** (*int*), um **saldo** atual (*float*) e um **cliente** (*Cliente*). Ao ser inicializada, a conta precisa obrigatoriamente da existência de pelo menos um cliente que será responsável por ela. Ela tem funções membro de **saque** (recebe um valor *float* que é retirado da conta se houver saldo: caso não tenha saldo suficiente, imprima uma mensagem de erro) e de **depósito** (que recebe um valor *float* e acrescenta ao saldo). Além disso, como de costume, ela deve ter uma função que **imprime** todos os dados sobre a conta e o cliente associado a ela.
  - Por último, a classe ContaEspecial é uma classe derivada de Conta (herança). Além de todos os atributos de Conta, ela possui um atributo **limite** (*float*) que é um limite de cheque especial. As funções de **saque** e **impressão** devem ser atualizadas nesta classe, para se adequarem às novas características. No saque da ContaEspecial, é possível sacar mais do que o saldo (dá para sacar até saldo + limite). Já na função de impressão, é necessário incluir a impressão do limite.

Por fim, escreva um programa em que são criados alguns clientes e contas (tanto a simples quanto a especial). Depois disso, coloque os ponteiros para as contas criadas em um vetor e, através de um laço, imprima as informações sobre todas as contas, em sequência (utilize o polimorfismo). A seguir, faça várias operações de saque e depósito nas contas e, finalmente, imprima novamente as informações de todas as contas, percorrendo o vetor.

Fique à vontade para criar dados e preencher os atributos das classes como preferir.

Escreva bons códigos!