

Extra - 03

---

## Orientação a Objetos - DCC025

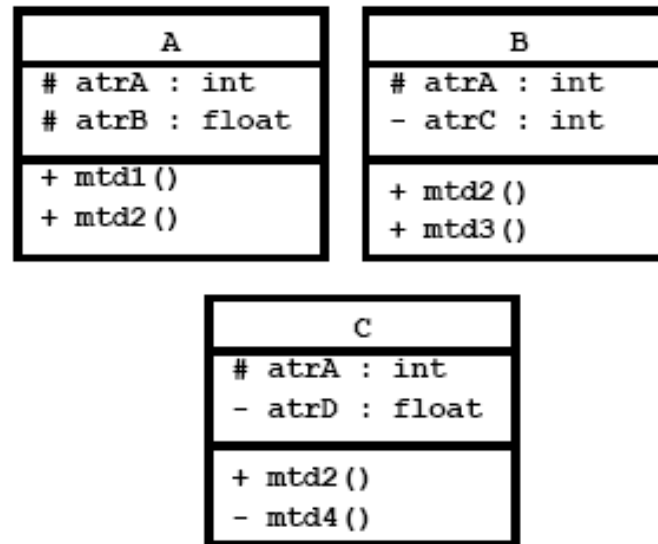
Prof. Edmar Welington Oliveira  
[oliveira.edmar@ufjf.edu.br](mailto:oliveira.edmar@ufjf.edu.br)

Universidade Federal de Juiz de Fora - UFJF  
Departamento de Ciência da Computação - DCC

# Exercício 01

Com base nas classes abaixo, criar um ou mais relacionamentos de herança entre elas.

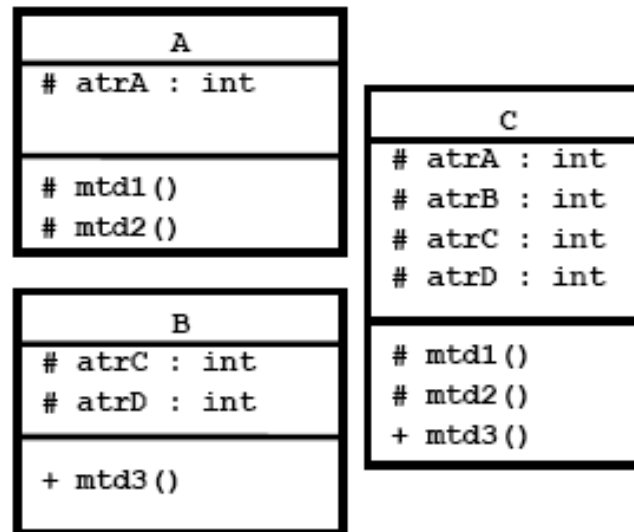
a)



## Exercício 02

Com base nas classes abaixo, criar um ou mais relacionamentos de herança entre elas.

b)



## Exercício 03

Elabore uma classe ContaBancaria, com os seguintes membros:

- ✓ atributo String cliente
- ✓ atributo int num\_conta
- ✓ atributo float saldo
- ✓ método sacar (o saldo não pode ficar negativo)
- ✓ método depositar

Agora acrescente ao projeto duas classes herdadas de ContaBancaria: ContaPoupança e ContaEspecial, com as seguintes características a mais:

⇒ Classe ContaPoupança:

- ✓ atributo int dia de rendimento
- ✓ método calcularNovoSaldo, recebe a taxa de rendimento da poupança e atualiza o saldo.

⇒ Classe ContaEspecial

- ✓ atributo float limite
- ✓ redefinição do método sacar, permitindo saldo negativo até o valor do limite.

## Exercício 03 - Continuação

Após a implementação das classes acima, você deverá implementar uma classe Contas.Java, contendo o método main. Nesta classe, você deverá implementar:

- a) Incluir dados relativos a(s) conta(s) de um cliente;
- b) Sacar um determinado valor da(s) sua(s) conta(s);
- c) Depositar um determinado valor na(s) sua(s) conta(s);
- d) Mostrar o novo saldo do cliente, a partir da taxa de rendimento, daqueles que possuem conta poupança;
- e) Mostrar os dados da(s) conta(s) de um cliente;

Extra - 03

---

## Orientação a Objetos - DCC025

Prof. Edmar Welington Oliveira  
[oliveira.edmar@ufjf.edu.br](mailto:oliveira.edmar@ufjf.edu.br)

Universidade Federal de Juiz de Fora - UFJF  
Departamento de Ciência da Computação - DCC