

<p>Universidade Federal da Paraíba</p> <p>Centro de Informática</p> <p>Departamento de Informática</p>	<p>Laboratório de LP1</p> <p>Semestre: 2016.2</p> <p>Professor: Tiago Maritan</p>
--	---

ROTEIRO 5

ASSUNTO: POO – Introdução a Classes e Objetos

1) (a) Crie uma classe chamada `Data` para representar uma data. Essa classe deve conter três atributos: o dia, o mês, e o ano. Considere também que a classe `Data` contém:

- Um construtor que inicializa os três atributos e verifica a validade dos valores fornecidos;
- Um método `set()` um `get()` para cada atributo;
- Um método `avancarDia()` que para avançar uma data para o dia seguinte.

(b) Crie uma classe `DataTest`, com um método `main`, que cria alguns objetos da classe `Data` e utiliza as suas operações (métodos).

2) a) Crie uma classe chamada `Invoice` para que uma loja de suprimentos de informática possa utilizá-la para representar uma fatura de um item vendido na loja. Uma `Invoice` (fatura) deve incluir quatro partes de informações como variáveis de instância (atributos) – o número, a descrição, a quantidade comprada de um item e o preço do item. Sua classe deve conter um construtor que inicializa as quatro variáveis de instância. Forneça um método `get` e `set` para cada atributo da classe. Além disso, forneça um método `getInvoiceAmount` que calcula o valor da fatura (isto é, multiplica a quantidade pelo preço por item). Se a quantidade não for positiva, ela deve ser configurada como 0 (zero). Se o preço por item não for positivo, ele deve ser configurado como 0.0.

b) Crie uma classe `InvoiceTest`, contendo um método `main`, que cria alguns objetos da classe `Invoice` e testa as capacidades da classe `Invoice`.

3) a) Crie uma classe chamada `Empregado` que possui três variáveis de instância (atributos) – um primeiro nome, um sobrenome e um salário mensal. Forneça um construtor que inicializa as três variáveis de instância. Forneça também um método `get` e `set` para cada atributo da classe. Se o salário mensal não for positivo, não configura esse valor.

b) Crie uma classe `EmpregadoTest` com um método `main` que demonstra as capacidades da classe `Empregado`. Nessa classe, crie dois objetos `Empregado` e exiba o **salário anual** de cada (objeto) `Empregado`. Então dê para cada `Empregado` um aumento de 10% e exiba novamente os seus salários anuais.

4) Considere um sistema que possui uma classe `Pessoa` e uma classe `CadastroDePessoas`.

(a) Implemente uma classe `Pessoa` que possui os atributos `nome`, `idade` e `telefone`. Considere também que ela possui dois construtores (um que só recebe o nome e outro que recebe nome, idade e telefone) e métodos para obter e alterar esses atributos (métodos `get` e `set`).

(b) Em seguida, crie uma classe principal `CadastroDePessoas` com o método `main` que cria alguns objetos do tipo `Pessoa` e configura os valores de nome, idade e telefone desses objetos.

5) Considere um Sistema de Controle de Gastos de Mesada. Esse sistema apresenta uma classe chamada `Despesa`, com os atributos `valor`, do tipo `double` e `tipoDeGasto` e métodos para obter e alterar esses atributos (métodos `get` e `set`).

Esse sistema apresenta também uma classe `ControleDeGastos`, que apresenta um atributo chamado `despesas`, que é um conjunto de objetos do tipo `Despesa` e o método `setDespesas`. Essa classe apresenta também um método chamado `calculaTotalDeDespesas`, que não tem parâmetros e deve retornar o valor total das despesas do sistema. Nessa classe há ainda o método `existeDespesaDoTipo()` que verifica se dentre as despesas guardadas no `ControleDeGastos` há alguma delas que é do tipo passado retornando `true` neste caso e `false`, caso contrário.

a) Implemente a classe `Despesa`.

b) Implemente a classe `ControleDeGastos`.

c) Escreva um programa principal que crie um objeto do tipo `ControleDeGastos` e que tenha ao menos duas despesas. Faça com que o programa imprima o total das despesas utilizando o método correspondente da classe `ControleDeGastos`.

6) Considere um Sistema de Controle de Pagamentos de Funcionários de uma empresa. Esse sistema apresenta uma classe chamada `Pagamento`, com os atributos `valorPagamento`, do tipo `double` e `nomeDoFuncionario` e métodos para obter e alterar esses atributos (métodos `get` e `set`).

Esse sistema apresenta também uma classe `ControleDePagamentos`, que apresenta um atributo chamado `pagamentos`, que é um conjunto de objetos do tipo `Pagamento` e o método `setPagamentos`. Essa classe apresenta também um método chamado `calculaTotalDePagamentos`, que não tem parâmetros e deve retornar o valor total dos pagamentos do sistema. Nessa classe há ainda o método `existePagamentoParaFuncionario (String nomeFuncionario)` que verifica se dentre os pagamentos guardados no `ControleDePagamentos` há algum deles que se refere ao funcionário passado como parâmetro, retornando `true` neste caso e `false`, caso contrário.

a) Implemente a classe `Pagamento`.

b) Implemente a classe `ControleDePagamentos`.

c) Escreva um programa principal que crie um objeto do tipo `ControleDePagamentos` e que tenha ao menos dois pagamentos. Faça com que o programa imprima o total dos pagamentos utilizando o método correspondente da classe `ControleDePagamentos`.