



**Orientações iniciais:**

- A prova é individual e sem consulta;
- A prova vale **100 pontos**, que equivalem à nota **10**;
- Distribua bem o tempo gasto em cada questão. A quantidade de pontos de cada questão corresponde ao tempo que você deveria gastar em cada uma delas;
- A prova tem duração de mínima de **30 minutos** e duração máxima de **3 horas**;
- Caso você tenha alguma dúvida referente ao enunciado das questões durante a prova, pergunte em voz alta. O professor não irá atendê-lo individualmente.
- Crie uma pasta com o seu nome na pasta Documentos e armazene todas as implementações da prova dentro dessa pasta. Você pode utilizar um editor de texto comum ou o Eclipse. Crie dentro dessa pasta um arquivo chamado *README.txt*, que contém um descritivo de como você organizou os exercícios (1 projeto no eclipse, vários projetos no eclipse, utilizou editor de texto comum, etc.).

**BOA PROVA!**



## Prova 1

### Linguagem de Programação Orientada a Objetos — 2016

1. (5.0 pontos) Em um posto de saúde, chegam pessoas com diferentes de doenças, e um dos papéis de um enfermeiro é registrar qual a doença que foi diagnosticada pelos médicos em cada paciente. O seu objetivo neste exercício será criar um pequeno sistema orientado a objetos que permite com que o enfermeiro de um posto de saúde cadastre pacientes e as doenças associadas a cada paciente.

Um paciente é identificado pelo seu nome, cpf (que deve ter o formato `YYY.YYY.YYY-YY`), rg, data de nascimento (que deve ter o formato `YY/YY/YYYY`) e endereço (rua, número e cep no formato `YY.YYY-YYY`). Um paciente pode visitar um posto de saúde quantas vezes ele considerar necessário e, a cada visita, o médico diagnostica qual a doença do paciente e cadastra no sistema. Uma doença é identificada por um CID (identificação numérica única de uma doença), nome e sintomas do paciente. Crie um sistema orientado a objetos que permita:

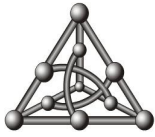
- (a) Cadastrar um paciente com apenas suas informações básicas;
- (b) Listar todos os pacientes, exibindo para cada paciente suas informações básicas e seu histórico de doenças;
- (c) Informar que um determinado paciente foi diagnosticado com uma determinada doença.

Para implementar uma solução para o problema acima, considere as seguintes observações adicionais:

- (a) O sistema deve admitir o cadastro de no máximo 1000 pacientes;
- (b) Um paciente possuirá no máximo 50 diagnósticos de doenças associadas a ele;
- (c) Um paciente é identificado unicamente por seu cpf;
- (d) Uma doença é identificada unicamente por seu CID;
- (e) O endereço de um paciente é a única informação de um paciente que pode ser alterada após o seu cadastro;
- (f) Uma vez cadastrada uma doença de um paciente, ela não pode ser alterada nem removida, e permanece no histórico do paciente da maneira como foi cadastrada. Portanto, só é possível adicionar novas doenças a um paciente.

Seu sistema deve conter pelo menos as seguintes classes: Paciente, Endereco e Doenca. O objetivo principal deste exercício é testar a capacidade do aluno de implementar um problema utilizando orientação a objetos com os princípios do encapsulamento.

Dica: o método `charAt` da classe `String` permite acessar o caractere de uma determinada posição em uma `String`.



**ATENÇÃO: o cenário descrito no exercício a seguir é hipotético e não retrata de maneira fidedigna o sistema político Brasileiro.**

2. (5.0 pontos) Neste exercício você criará um sistema capaz de gerenciar as informações básicas de um político no Brasil e quais são os seus gastos para os cofres públicos.

Todos os políticos possuem as seguintes informações básicas: nome, cpf (que deve ter o formato `YYY.YYY.YYY-YY`), partido, salário base (que não pode ultrapassar R\$ 15.000,00) e bonificação. O salário total de um político é inicialmente composto pelo seu salário base mais o valor da bonificação. O nosso sistema gerenciará as informações básicas dos seguintes cargos políticos: vereador, deputado estadual, deputado federal.

Um vereador possui auxílio moradia e auxílio para compra de celulares novos. O auxílio moradia deve ser incluído no cálculo do salário total de um vereador.

Todos os deputados possuem um auxílio combustível (que não entra na conta do salário total) e uma bonificação salarial extra de 10% do seu salário base. Entretanto, deputados estaduais possuem um vale alimentação, e seu salário total deve incluir o vale alimentação. Já um deputado federal possui auxílio terno, mas esse valor não entra no cálculo do seu salário total.

Crie um sistema orientado a objetos que permita:

- (a) Cadastrar um vereador;
- (b) Cadastrar um deputado estadual;
- (c) Cadastrar um deputado federal;
- (d) Listar, para cada político, suas informações básicas, seu salário total e o valor recebido em vales e auxílios.

O objetivo principal deste exercício é testar a capacidade do aluno de implementar um problema utilizando orientação a objetos com os princípios do encapsulamento, herança e polimorfismo.

O item d é o item no qual vocês mais serão avaliados pelo uso do polimorfismo. Para isso, não se esqueçam de incluir um método salarioTotal na sua classe base e, sempre que possível, utilizar variáveis da classe base.