

Orientações iniciais:

- A prova é individual e sem consulta;
- A prova contém duas questões práticas valendo 12 pontos;
- A prova tem duração de mínima de 30 minutos e duração máxima de 3 horas;
- Utilize a IDE Eclipse ou um editor de textos comum para implementar a prova;
- Crie uma pasta com o seu nome na pasta Documentos e armazene o seu projeto dentro dela.

BOA PROVA!



Prova Optativa

Linguagem de Programação Orientada a Objetos — 2014

Turma P02

1. **(60 pontos)** Nesse exercício você deve implementar um conjunto de classes que modela os super-heróis descritos a seguir.

Todo super-herói possui as seguintes características: nome original, descrição, data de criação, país natal, nível de resistência, estado de saúde e um conjunto de 10 habilidades. O nível de resistência de um super-herói é um valor entre 0 (morto) e 100 que indica quão forte e bem de saúde o super-herói está. Portanto, o estado de saúde de um super-herói é associado ao seu nível de resistência: nível de resistência entre 0 e 30 indica um estado de saúde ruim, nível de resistência entre 31 e 70 indica um estado de saúde bom e nível de resistência entre 71 e 100 indica um estado de saúde excelente.

Além disso, todo super-herói tem dois comportamentos clássicos: machucar outro super-herói e ajudar outro super-herói, mas a intensidade do dano e da ajuda depende do super-herói. O homem de ferro é muito forte e egoísta, portanto, quando machuca outro super-herói, reduz em 20 unidades o nível de resistência do super-herói machucado, enquanto que, quando ajuda outro super-herói, aumenta em apenas 5 unidades o nível de resistência do super-herói ajudado. Ainda em relação ao homem de ferro, é importante armazenar a versão atual da sua armadura (número entre 1 e 10) e a quantidade de namoradas atuais.

Já o Wolverine, apesar de ser durão, tem um coração mole. Quando machuca outro super-herói, reduz em apenas uma unidade o nível de resistência do super-herói machucado, enquanto que, quando ajuda outro super-herói, aumenta em 20 unidades o nível de resistência do super-herói ajudado. Em relação ao Wolverine, também é interessante armazenar a quantidade de adamantium no corpo (em miligramas e com duas casas decimais) e tempo médio de cicatrização (em minutos). Um pequeno detalhe é que, quando o Wolverine machuca o Homem de Ferro, ele também rouba uma de suas namoradas.

Há também o Ciclope, que se considera um super-herói justo. Quando machuca outro super-herói, reduz em 10 unidades o nível de resistência do super-herói machucado, e quando tenta ajudar outro super-herói, aumenta em 10 unidades o nível de resistência do super-herói ajudado. Em relação ao Ciclope, sua richa pessoal é com o Wolverine: toda vez que ele o machuca, ele consegue fazer com que o tempo médio de cicatrização do Wolverine aumente em 5 minutos. Em relação ao Ciclope, é interessante armazenar qual a marca do seu óculos atual, que pode ser Oakley, HB ou Rayban.

Por mais que não seja um super-herói, o Coringa gosta muito de machucar e, de vez em nunca, ajudar super-heróis. Quando machuca outro super-herói, ele leva o nível de resistência do super-herói machucado a 1, e quando tenta ajudar outro super-herói, aumenta em apenas 1 unidade o nível de resistência do super-herói ajudado. Em relação ao Coringa, deve ser possível armazenar em qual país ele mora atualmente.



Considere os seguintes detalhes técnicos na implementação das classes acima:

- (a) Modele as suas classes antes de implementá-las. Isso facilitará muito a implementação;
- (b) Encapsule corretamente e de maneira coerente os atributos de todas as classes que você criar. Lembre-se de dar acesso a um atributo apenas quando isso não romper as regras de uso da classe. Lembre-se também de sempre manter os valores dos atributos consistentes;
- (c) Os construtores das classes criadas devem inicializar a maior quantidade possível de atributos;
- (d) Uma data deve ser representada por uma outra classe, com atributos encapsulados *dia, mês* e *ano,* que devem ser mantidos consistentes;
- (e) Há um conjunto muito específico de países, que são representados por uma das seguintes siglas: *BRA* (Brasil), *USA* (Estados Unidos) e *CAN* (Canadá). Portanto, esse conjunto deve ser representado através de um Enum com um atributo *nome completo*;
- (f) O estado de saúde e a marca do óculos também devem ser representados utilizando Enums;
- (g) As habilidades de um super-herói são adicionadas através de um método *adicionaHabilidade*. Uma habilidade é representada por uma cadeia de caracteres;
- (h) Você deve usar herança e seus recursos;
- (i) Sua implementação deve ter uma e apenas uma interface;
- (j) Os métodos *machucar* e *ajudar* devem ser implementados utilizando polimorfismo e devem possuir apenas um argumento: uma instância de um super-herói a ser machucado ou ajudado.
- (k) Deve ser possível imprimir qualquer objeto instanciado;
- (l) Só deve ser possível instanciar um super-herói se ele for ou o Homem de Ferro, ou o Wolverine ou o Ciclope;
- (m) Crie uma classe *TesteSuperHeroi*, que teste de maneira satisfatória as classes criadas neste exercício.



2. **(6.0 pontos)** *Tonio Raumus Peludus* é dono de um importante frigorífico no estado de Mato Grosso do Sul e, após as fraudes reveladas pela operação "Carne Fraca" da Polícia Federal, decidiu estabelecer um controle de qualidade mais rigoroso das carnes do seu frigorífico. Para isso, Tonio precisará de sua ajuda.

Diariamente, o frigorífico de Tonio (que trabalha apenas com carne bovina) recebe várias peças/pedaços de carnes, e para cada peça as seguintes informações são armazenadas: região do boi (picanha, maminha, alcatra, coxão mole, etc.), peso (em kg), origem (cidade e estado) e nome da fazenda de origem.

As informações referentes às peças recebidas estão armazenadas em um arquivo, onde cada linha corresponde a informações de uma peça recebida. Cada linha do arquivo obedece o seguinte formato: região,peso,cidade_origem,estado_origem,fazenda.

O arquivo abaixo é um exemplo de arquivo de entrada.

```
Picanha, 2, Campo Grande, MS, Fazenda do seu Zé do Papelão Coxão Mole, 3, Curitiba, PR, Recanto do Acido Ascórbico Maminha, 3, Brasília, DF, Rancho não há o que temer
```

Dada uma entrada com este formato, Tonio deseja que você forneça a ele os seguintes relatórios:

- Relatório 1: listagem ordenada de todas as peças contendo todas as informações. O nome da cidade deve ser usado como critério de ordenação;
- Relatório 2: listar a quantidade de peças de cada região do boi e o peso total das peças. Por exemplo: maminha -> 2 peças, total 15 kg; picanha -> 4 peças, total 22kg; etc.
- Relatório 3: listar, para cada fazenda, quais as peças que foram fornecidas. Por exemplo: Fazenda do seu Zé do Papelão -> Picanha, Maminha; Recanto do Acido Ascórbico -> Alcatra, Bananinha; etc.

Neste exercício você **DEVE** utilizar **PELO MENOS** as seguintes classes e interfaces Java: Collections, List, Map e Set.

Para a leitura de arquivo, você pode utilizar as seguintes classes e métodos:

```
BufferedReader bf =
new BufferedReader(new FileReader(new File(nomeDoArquivo)));
bf.ready(); // pergunta se há linhas no arquivo
bf.readLine(); // lê uma linha
```