Computação 2 - 2020/1 - Prova Única - Prof. Vinícius
Todas as questões referem-se ao código disponibilizado no projeto GuiaTuristico, no GitHub.

1) Se tentarmos executar o teste abaixo, o que acontece, e por quê?

```
EstabelecimentoAvaliado hotel = new Hotel("Palace");
Endereco endereco = new Endereco("Rua A", 100, "Rio");
hotel.setEndereco(endereco);
assertEquals(endereco, hotel.getEndereco());
```

2) Se executarmos dentro do main() o código abaixo, o que acontece, e por quê?

```
Estabelecimento posto = new PostoDeGasolina("Teste");
System.out.println(posto.getNome());
```

3) Se executarmos dentro do main() o código abaixo, o que acontece, e por quê?

```
GuiaTuristico guia1, guia2, guia3;
guia1 = new GuiaTuristico("Rio");
guia2 = new GuiaTuristico("Curitiba");
System.out.println(guia2.getQuantidadeGuias());
```

4) Assinale **todas** as modificações que **precisariam** ser feitas em Estabelecimento para que pudéssemos compilar corretamente o código abaixo:

```
Estabelecimento loja = new Estabelecimento();
loja.setNome("Minha Loja");
```

a) Remover o final do atributo nome.
b) Criar o setter público setNome().
c) Criar o getter público getEndereco().
d) Acrescentar um construtor default.
e) Remover abstract de getTipoEstabelecimento() e implementá-lo.
f) Remover abstract da classe Estabelecimento.

5) Por que o typecast do método sugerirEstabelecimento() em GuiaTuristico é seguro, no sentido de não corrermos o risco de tomar uma RuntimeException de cast inválido?

a) Porque todo Estabelecimento estende o tipo genérico T.
b) Porque o array estabelecimentos só permite escrever objetos que aceitam esse typecast.
c) Porque todo typecast é intrinsicamente seguro se o código compila corretamente.
d) Porque a classe GuiaTuristico escreve apenas quem é do tipo genérico T no array estabelecimentos.

6) Supondo que apenas as ações relacionadas abaixo possam ser feitas, qual ou quais delas precisaria de fato ser feita, necessariamente, para que um GuiaTuristico pudesse incluir também postos de gasolina?

a) Fazer com que a classe PostoDeGasolina extends EstabelecimentoAvaliado.
b) Acrescentar um atributo público String avaliacao em PostoDeGasolina.
c) Declarar seu GuiaTuristico como GuiaTuristico<PostoDeGasolina>.
d) Acrescentar um método público getAvaliacao() em PostoDeGasolina.
e) Fazer com que a classe PostoDeGasolina implements Avaliado.

7) Qual ou quais das ações abaixo **não** eliminariam o *risco* de uma NullPointerException ao executarmos o terceiro if do método incluirEstabelecimento() em GuiaTuristico?

a) Acrescentarmos o código abaixo antes da comparação que já existe naquele terceiro if:

```
estabelecimento.getEndereco().getCidade() = null ||
```

b) Acrescentarmos um outro if, entre o primeiro e o segundo if que lá estão:

```
if (estabelecimento.getEndereco().getCidade() = null) {
      throw new EnderecoInvalidoException("Cidade nula");
}
```

c) Acrescentarmos NullPointerException à cláusula throws na declaração do método.
d) Envolvermos o segundo if com um try...catch.
e) Verificarmos, no construtor de Endereco, que a cidade passada como parâmetro não é nula, lançando lá uma exceção caso seja.

8) Escreva como você faria para declarar e instanciar, *em uma única linha de código*, um GuiaTuristico que lide apenas com hotéis. (Escreva a linha de código.)

9) Se você quisesse categorizar restaurantes como Churrascaria, Pizzaria, Japonês, Vegano ou Variado, o que você faria? (Não precisa escrever código, mas diga **tudo** que você teria que fazer para que a categoria pudesse ser lida e escrita em um restaurante.)

10) Se você quisesse que a classe GuiaTuristico oferecesse um método para buscar um estabelecimento a partir de seu nom exato, o que você faria para não ter que percorrer possivelmente uma lista inteira de estabelecimentos até encontrar o que deseja?