

Exame de Paradigmas de Programação

Exame da Época Normal - 26/06/2019
1º Ano da Licenciatura em Engenharia Informática do ISEP

Parte prática (14 valores); Cotações: 1. – 20%; 2. – 50%; 3. – 30%

Prova sem consulta; Duração: 1h45

Responda a cada pergunta numa folha separada, identificada com número e nome.

Recomenda-se a leitura integral do enunciado antes de começar a responder.

Pretendem-se classes Java para uma aplicação de apoio à gestão de imóveis numa imobiliária. Todos os imóveis têm um proprietário e podem ser de três tipos: apartamento, moradia ou escritório.

Responda às alíneas seguintes tendo sempre em consideração os principais princípios da programação orientada por objetos: abstração, encapsulamento, herança e polimorfismo.

1. Defina:

- A classe `Proprietario` para representar proprietários, caracterizados pelo nome, NIF (int), email e número de telefone (int). Considere que já existe: o construtor completo, o método reescrito `equals`, os métodos `get` para todas as características e os métodos `set` para todas as características, exceto o número de telefone.

Especifique:

- Os atributos, o construtor sem parâmetros (declare constantes para os valores por omissão) e o construtor de cópia.
- O método `toString` usando o método `format` da classe nativa do Java `String`.
- Uma classe de exceção própria `unchecked`, chamada `ArgumentoInvalidoException`. Considere que esta exceção é lançada no método `set` do atributo `telefone` quando este não tem 9 dígitos. A mensagem enviada deve ser personalizada.

2. Elabore as classes representativas de cada tipo de imóvel referidos e que satisfaçam os seguintes requisitos:

- Considere que já existe para cada uma das classes de imóveis: os construtores, os métodos reescritos `equals` (exceto para a classe das moradias) e `toString` e os métodos `get` e `set` para todas as características.
- Cada imóvel é caracterizado pela morada, localidade, área útil e preço.
- As moradias e os apartamentos (imóveis de habitação) são ainda caracterizados pelo número de quartos.
- As moradias são ainda caracterizadas pela sua área exterior.
- As comissões praticadas pela imobiliária são de 4% para os apartamentos e de 7% para escritórios. Se o valor das moradias for inferior a 150 000 euros a comissão praticada é de 5%, caso contrário é de 6%. Estes valores são passíveis de ser alterados no futuro.
- Deve ser fornecida uma funcionalidade que permita obter a quantidade de apartamentos existentes.
- As classes devem estar preparadas para, usando código nativo do Java, permitir a ordenação de uma lista das suas instâncias, por localidade (ordem alfabética) e em cada localidade por ordem crescente de áreas.
- Deve ser disponibilizado o método `calcularValorImovel` que devolverá o valor total a pagar pela compra de um imóvel. O valor total a pagar será o preço do imóvel acrescido da comissão cobrada pela imobiliária.
- Os apartamentos e os escritórios terão que pagar condomínio. Nos apartamentos, o valor definido é de 50 centimos por metro quadrado do imóvel enquanto que nos escritórios é de 40 centimos por metro quadrado. Estes valores são passíveis de serem alterados no futuro.
- Deve ser declarada e implementada uma interface designada `Condominio`, que deve definir o método `calcularCondominio` que calcula e devolve o valor que os apartamentos e os escritórios terão de pagar mensalmente pelo condomínio.
- Método `equals` reescrito da classe das moradias. Para tal, faça uso de outro método `equals` já implementado da hierarquia de classes. Considere iguais as moradias que possuem todos os atributos com valor igual.

3. Defina a classe `ListaImoveis` para representar um conjunto de imóveis. Os imóveis devem ser armazenados num contentor do tipo `ArrayList`. Especifique os atributos da classe e apenas os seguintes métodos:

- O método `eliminarImovel` para eliminar do contentor o imóvel com a menor área de uma localidade recebida por parâmetro. Para tal, faça uso da ordenação do contentor. Deve ser possível identificar o sucesso ou insucesso da operação com o valor do retorno do método.

- Método `calcularImoveis` que a partir dos imóveis armazenados no contentor, calcule e apresente por tipo de imóvel, o valor de venda desses imóveis.
- Método `exportarParaFicheiro` para guardar uma `ListaImoveis` para num ficheiro binário, cujo nome é passado por parâmetro. Deve retornar um valor booleano para indicar o sucesso/insucesso da ação executada. Considere a possibilidade de ser gerada uma exceção do tipo `IOException`. No caso de considerar necessário alterar as classes desenvolvidas anteriormente, indique estas modificações.