# PROVA ONLINE COITOS


*O Retorno da União Anal Perfeita*

**LARANJA:** INCONSISTENTE ENTRE DUAS OU MAIS FONTES
**AMARELO:** DÚVIDA
**VERDE:** CORRETA
**AZUL:** GRÊMIO 🟦🟦🟦🟦🟦🟦🟦🟦🟦🟦🟦🟦🟦

**QUESTÃO 1: Considere o método implementado no código Questao1.java. Assinale a alternativa que corretamente descreve o que o método faz:**

> Devolve uma nova lista contendo os elementos **únicos** da lista original (isto é, sem a repetição dos valores considerados iguais) em ordem **crescente** de valor (segundo o critério de ordem implementado para tipo T).

**QUESTÃO 2: Considere o método implementado no código Questao2.java. Assinale a alternativa que corretamente descreve o que o método faz:**

> Devolve uma nova lista contendo os elementos **únicos** da lista original (isto é, sem a repetição dos valores considerados iguais) em ordem **indefinida** (ou seja, em uma ordem aparentemente aleatória que não leva em conta nem o valor dos elementos, e nem a ordem das primeiras aparições de cada elemento distinto).

**QUESTÃO 3: Considere o método implementado no código Questao3.java. Assinale a alternativa que corretamente descreve o que o método faz:**

> Nenhuma das demais alternativas descreve corretamente o que o método faz.

**QUESTÃO 4: Considere o método implementado no código Questao4.java. Assinale a alternativa que corretamente descreve o que o método faz:**

> Devolve um novo conjunto contendo os elementos exclusivos de A e os exclusivos de B (ou seja, apenas aqueles que não fazem parte simultaneamente de ambos os conjuntos).

**QUESTÃO 5: Considere o código Questao5.java. Ao compilar e executar tal código, vemos que o programa imprime a saída "Chave inexistente!", mesmo que exista no mapa uma chave que possui o campo id com o valor 5020. O que precisa ser feito para que o objeto Pessoa, armazenado no mapa**

🟥SPORT CLUB INTERNACIONAL 🟥 🟥🟥🟥🟥🟥🟥🟥🟥🟥🟥🟥🟥

**associado à chave que tem campo id igual a 5020, seja adequadamente encontrado e devolvido?**

> É preciso sobrescrever o método equals da classe Chave, de modo a considerar duas instâncias de Chave iguais quando ambas possuem o mesmo valor no atributo id. Além disso, o método hashcode desta classe também precisa ser redefinido de modo que dois objetos do tipo Chave considerados iguais também devolvam hashcodes iguais.

**QUESTÃO 6: Considere o método mediana, codificado no arquivo Questao6.java, que recebe uma lista valores do tipo double previamente ordenada, e determina a mediana destes valores. Em relação a este método, assinale a afirmativa incorreta:**

> O método pode receber listas tanto do tipo ArrayList quanto LinkedList, e em ambos os casos irá cumprir seu papel de devolver o valor da mediana. Além disso, a complexidade assintótica do método será sempre a mesma, qualquer que seja o tipo concreto da lista recebida em uma chamada.

**QUESTÃO 7: Considere a operação que remove um elemento de uma coleção do tipo List, a partir de seu índice. Podemos afirmar que:**

> A remoção do primeiro elemento é mais eficiente em uma lista do tipo LinkedList (em comparação à mesma operação feita sobre uma ArrayList).

**QUESTÃO 8: O método contains, declarado na interface Collection, recebe como parâmetro uma referência para um objeto e verifica se existe na coleção um objeto considerado igual ao recebido como parâmetro. Obviamente, é esperado que cada tipo de coleção implemente o contains de forma diferente, de modo a tirar proveito das características específicas de cada estrutura de dados. Além disso, para que as coleções sejam capazes de armazenar elementos de qualquer tipo e funcionar de forma adequada, as comparações feitas entre o objeto recebido como parâmetro e os objetos armazenados pela coleção são "terceirizadas" para fora da coleção (em geral, ficam a cargo da própria classe dos objetos guardados). Ou seja, o critério que determina quando dois objetos são considerados iguais é implementado fora da coleção em si. Em relação a esse funcionamento do método contains, marque a alternativa incorreta:**

> Todas as coleções Java estudadas (ArrayList, LinkedList, HashSet e TreeSet) dependem do critério de comparação definido no método equals (que todo objeto possui) em suas respectivas implementações do contains.

**QUESTÃO 9: Considere a classe FilterList, codificada no arquivo Questao13.java. Esta classe implementa uma lista linear, que usa como estrutura interna de armazenamento um array, e só aceita a adição de valores que satisfazem uma certa propriedade (propriedade esta que é escolhida na instanciação da lista). Selecione a alternativa que adequadamente relaciona os princípios SOLID que são violados na implementação desta classe:**

Responsabilidade única, aberto/fechado.

**QUESTÃO 10: Considere o código das classes Retangulo e Quadrado, codificadas no arquivo Questao14.java. Podemos afirmar que a declaração da classe Quadrado, por derivação da classe Retangulo, viola qual princípio SOLID?**

Substituição de Liskov.

**QUESTÃO 11: Ainda considerando a classe Quadrado da questão anterior, são listadas abaixo algumas sugestões de alterações que visam melhorar a declaração da mesma, com o objetivo de não mais violar um dos princípios SOLID:**

**I - Desfazer a relação hierárquica existente entre as classes Retângulo e Quadrado.**

**II - Usar composição como forma de reaproveitar a implementação já existente na classe Retangulo, ao mesmo tempo que se evita o recebimento por herança de comportamentos inapropriados para a classe Quadrado (mas que fazem sentido para a classe Retangulo).**

**III - Manter a relação de herança entre Quadrado e Retângulo, mas sobrescrever o método set na classe Quadrado de modo a ignorar um dos parâmetros recebidos, e atribuir o valor do outro em ambos os atributos w e h.**

**As alterações sugeridas que efetivamente evitam a violação de um dos princípios SOLID são:**

I e II

**QUESTÃO 12: Em relação ao princípio da inversão da dependência, são feitas as seguintes colocações:**

**I - A aderência ao princípio favorece um menor acoplamento entre as classes de um projeto.**

**II - O processo de desenvolvimento é voltado para as abstrações e não para as implementações concretas.**

**III - A existência de camadas de abstração entre as classes do projeto também favorece o respeito ao princípio aberto/fechado.**

**São colocações válidas:**

I, II e III