

Revisar envio do teste: Semana 5 - Atividade Avaliativa

Programação Orientada a Objetos - COM230 - Turma 012

Página Inicial

Avisos

Cronograma

Atividades

Fóruns

Calendário Lives

Colaborate

Notas

Menu das Semanas

Semana 1

Semana 2

Semana 3

Semana 4

Semana 5

Semana 6

Semana 7

Semana 8

Orientações para realização da prova

Orientações para realização do exame

Documentos e informações gerais

Gabaritos

Referências da disciplina

Repositório de REA's

Usuário

LIZIS BIANCA DA SILVA SANTOS

Curso

Programação Orientada a Objetos - COM230 - Turma 012

Teste

Semana 5 - Atividade Avaliativa

Iniciado

09/11/23 19:50

Enviado

09/11/23 19:58

Data de vencimento

10/11/23 05:00

Status

Completada

Resultado da tentativa

10 em 10 pontos

Tempo decorrido

7 minutos

Instruções

Olá, estudante!

1. Para responder a esta atividade, selecione a(s) alternativa(s) que você considerar correta(s);

2. Após selecionar a resposta correta em todas as questões, vá até o fim da página e pressione “Enviar teste”.

3. A cada tentativa, você receberá um conjunto diferente de questões.

Pronto! Sua atividade já está registrada no AVA.

Resultados exibidos Todas as respostas, Respostas enviadas, Respostas corretas, Comentários, Perguntas respondidas incorretamente

Pergunta 1

1,42 em 1,42 pontos

Set e Map são duas interfaces importantes do Java, e ambas são encontradas na estrutura da Coleção. Além disso, as interfaces Set e Map podem ser usadas para armazenar uma Coleção de objetos na forma de uma única unidade. Nesse sentido, “Maps diferem de Sets pelo fato de que Maps contêm chaves e valores, enquanto Sets contêm somente valores. Três das várias classes que implementam a interface Map são Hashtable, HashMap e TreeMap. HashTables e HashMaps armazenam elementos em Tabelas Hash, e TreeMaps armazenam elementos em árvores” (DEITEL; DEITEL, 2016, p. 561).

DEITEL, P.; DEITEL, H. Java: como programar. 10. ed. São Paulo: Pearson Education do Brasil, 2016.

De acordo com o apresentado no texto, e com base nas características de Mapas, identifique se são verdadeiras (V) ou falsas (F) as afirmativas a seguir.

I. ( ) A interface SortedMap estende Map e mantém as suas chaves em ordem classificada – na ordem natural dos elementos ou em uma ordem especificada por uma implementação Comparator. A classe TreeMap implementa SortedMap.

II. ( ) As classes Hashtable e HashMap permitem usar *hashes* sem ter de implementar mecanismos de Tabela Hash – um exemplo clássico da reutilização, paradigma da programação orientada a objetos.

III. ( ) A interface Java Map tem um método chamado putAll(), que pode copiar todos os pares chave + valor (entradas) de outra instância do Map para si mesmo. Na teoria dos Conjuntos, isso também é referido como a interseção de duas instâncias de Map.

Assinale a alternativa que apresenta a sequência correta.

Resposta Selecionada: 

a. V, V, F.

Respostas: 

a. V, V, F.

b. F, V, V.

c. V, V, V.

d. F, V, F.

e. V, F, V.

Comentário da resposta: JUSTIFICATIVA

A afirmativa I é verdadeira, pois essa interface é usada com o intuito de estabelecer a ordenação natural dos elementos ou com base em alguma especificação estabelecida em uma implementação do tipo Comparator.

A afirmativa II é verdadeira, pois HashMap e Hashtable armazenam pares de chaves e valores em uma Tabela Hash. Ao usar um Hashtable ou um HashMap, especificamos um objeto usado como uma chave e o valor que se deseja vincular a essa chave. A chave recebe o Hash, e o código Hash resultante é usado como o índice no qual o valor é armazenado na tabela.

A afirmativa III é falsa, pois, na verdade, na teoria dos conjuntos, isso também é referido como a união de duas instâncias de Map, pois não obterão apenas elementos repetidos, mas na totalidade, nas duas instâncias referenciadas.

Pergunta 2

1,42 em 1,42 pontos

“Abordagens de solução de problemas de recursão têm um número de elementos em comum. Quando um método recursivo é chamado para resolver um problema, na verdade, ele é capaz de atuar somente no(s) caso(s) mais simples(s), ou caso(s) básico(s). Se o método é requisitado para um caso básico, ele retorna um resultado. Se é para um problema mais complexo, ele o divide em duas partes conceituais – uma que o método sabe como solucionar e outra que ele não sabe. Para tornar a recursão realizável, a última parte deve assemelhar-se ao problema original, mas ser uma versão ligeiramente mais simples ou menor dele. Como esse novo problema é parecido com o original, o método destina uma cópia nova dele próprio para trabalhar no problema menor – isso é referido como chamada recursiva e também é denominado passo de recursão” (DEITEL; DEITEL, 2016, p. 609).

DEITEL, P.; DEITEL, H. Java: como programar. 10. ed. São Paulo: Pearson Education do Brasil, 2016.

Sobre o que foi apresentado, observe as asserções a seguir e as relações propostas entre elas.

I. A execução de uma recursão é feita durante a chamada do método original, que ainda se encontra em aberto, ou seja, a execução não foi finalizada.

PORQUE

II. É possível que o passo de execução resulte em outras chamadas recursivas à medida que o método divide cada novo subproblema em partes conceituais.

Analizando as asserções anteriores, assinale a alternativa correta.

Resposta Selecionada: 

b. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa para I.

Respostas: 

a. As asserções I e II são proposições verdadeiras, e a II é uma justificativa para I.

b. As asserções I e II são proposições verdadeiras, mas a II não é uma justificativa para I.

c. As asserções I e II são proposições falsas.

d. A asserção II é uma proposição verdadeira, e a I é uma proposição falsa.

e. A asserção I é uma proposição verdadeira, e a II é uma proposição falsa.

Comentário da resposta: JUSTIFICATIVA

A asserção I é verdadeira, pois é realizada a chamada do método em execução antes que o próprio realize a finalização de sua execução atual. Envolve o uso de um procedimento, sub-rotina, função ou algoritmo que chama a si mesmo em uma etapa com uma condição de término, de modo que repetições sucessivas sejam processadas até a etapa crítica em que a condição é atendida.

A asserção II é verdadeira, mas não justifica a I, pois, caso outras chamadas sejam necessárias, as subpartes ou os subproblemas conceituais são “criados” para que a execução possa ser feita.

Pergunta 3

1,42 em 1,42 pontos

Disponível no pacote java.util, a API Collections é um conjunto de classes e interfaces que implementam Collection. Assinale a alternativa correta:

I. Uma coleção Set é uma coleção que não admite elementos duplicados. A forma como consegue isso é baseada no método equals dos elementos que tentas inserir.

II. A interface-raiz na hierarquia e coleções a partir da qual as interfaces Set e List são derivadas chama-se Collection

III. Set é uma coleção que possui elementos duplicados

IV. A interface List é uma coleção que associa chaves a valores e que não pode conter chaves duplicadas.

Resposta Selecionada: 

a. Apenas I e II estão corretas.

Respostas: 

a. Apenas I e II estão corretas.

Apenas I e IV estão corretas.

Apenas II e III estão corretas.

Apenas I, II e III estão corretas.

Apenas I, II e IV estão corretas.

Comentário da resposta: RESOLUÇÃO

A resposta correta é: “Apenas I e II estão corretas.”

Justificativa

Set é uma coleção que não possui elementos duplicados. E a coleção que associa chaves a valores e que não deve conter chaves duplicadas chama-se Map.

Pergunta 4

1,42 em 1,42 pontos

Considere a sequência do código a seguir:

```
import java.util.LinkedHashSet;
import java.util.Set;

public class TesteHashSet {

    public static void main(String[] args) {
        Set<String> nomes = new LinkedHashSet<String>();
        nomes.add("Joao");
        nomes.add("Jose");
        nomes.add("Maria");
        nomes.add("Julio");
        nomes.add("Fernanda");
        nomes.add("Heloisa");
        nomes.add("Ana");
        for (String nome : nomes) {
            System.out.println(nome);
        }
    }
}
```

Acerca da utilização da Classe LinkedHashSet, assinale a alternativa correta:

Resposta Selecionada: 

a. Ordem em que os dados serão retornados é a mesma em que foi inserida.

Respostas: 

A ordem correta é Joao, Jose, Maria, Julio, Fernanda, Heloisa, Ana.

A ordem correta é Ana, Joao, Jose, Maria, Julio, Fernanda, Heloisa.

A ordem em que os dados serão retornados não é garantida.

A ordem correta é Ana, Joao, Jose, Julio, Maria, Fernanda, Heloisa.

A ordem em que os dados serão retornados é a mesma em que foi inserida.

Comentário da resposta: RESOLUÇÃO

A resposta correta é: “A ordem que os dados serão retornados é a mesma em que foi inserida.”

Justificativa

A classe LinkedHashSet que é uma extensão da classe HashSet, difere desta última naquela classe os dados são recuperados na ordem em que foram inseridos.

Pergunta 5

1,42 em 1,42 pontos

A classe Java HashSet é usada para criar uma Coleção que usa uma Tabela Hash para armazenamento. Ela é uma implementação da interface Set que usa um HashMap interno para armazenar os seus elementos A interface Set herda as interfaces Collection e Iterable em ordem hierárquica.

Sobre a classe HashSet e as características de conjunto em Java, avalie as afirmativas a seguir.

I. O HashSet possui como mecanismo de armazenamento de elementos o *hashing*.

II. O HashSet contém apenas elementos exclusivos, podendo ser nulos.

III. O HashSet mantém a ordem de inserção, além de salvar os hashCode.

Está correto o que se afirma em:

Resposta Selecionada: 

a. I e II, apenas.

Respostas: 

a. I e III, apenas.

b. I, apenas.

c. II e III, apenas.

d. I, II e III, apenas.

e. I e II, apenas.

Comentário da resposta: JUSTIFICATIVA

A afirmativa I está correta, uma vez que a base do HashSet é a utilização de mecanismos ligados à Tabela Hash, no caso, mecanismo de *hashing*. A classe HashSet usa o hashCode dos objetos para determinar a posição dos elementos no HashMap interno. Quando você adiciona um elemento ao HashSet, ele é adicionado ao HashMap, com a chave sendo o hashCode do elemento e o valor sendo o próprio elemento.

A afirmativa II está correta, pois o Hash possui como característica a não repetição dos elementos, permitindo elementos nulos em sua composição. O hashCode é uma representação numérica de um objeto em Java. Ele é gerado pela chamada ao método hashCode() da classe Object e é usado para identificar, de forma única, um objeto em uma estrutura de dados que usa *hashing*, como o HashMap.

A afirmativa III está incorreta, pois o HashSet baseia-se no hashCode para realizar a inserção e não mantém a ordem de chegada dos elementos. Embora seja possível adicionar elementos a um HashSet em uma determinada ordem, a ordem em que eles são armazenados no HashMap interno pode diferir, devido ao uso de hashCodes para identificar a posição dos elementos.

Pergunta 6

1,42 em 1,42 pontos

Na linguagem Java, é comum o uso da classe “Collection”, que tem classes-filhas para listas e conjuntos. Um conjunto funciona de forma análoga aos conjuntos da matemática e proporcionam grande poder para a programação orientada a objetos.

Em relação ao melhor uso dos conjuntos em linguagem Java, assinale a alternativa correta.

Resposta Selecionada: 

b. Os conjuntos são uma Collection que não permite elementos duplicados.

Respostas: 

Exemplos de conjuntos são as classes List e Queue.

b. Os conjuntos são uma Collection que não permite elementos duplicados.

c. Os conjuntos são definidos pela classe Vector, permitindo a duplicidade de elementos.

d. Os conjuntos são uma coleção ordenada de elementos.

e. Os conjuntos são definidos pela classe ArrayList.

Comentário da resposta: JUSTIFICATIVA

Um conjunto funciona de forma análoga aos conjuntos da matemática. Trata-se de uma Collection que não permite elementos duplicados. Um conjunto é representado pela interface Set, tendo como principais implementações as classes: HashSet, LinkedHashSet e TreeSet.

Pergunta 7

1,48 em 1,48 pontos

A utilização de recursividade não apenas diminui a quantidade de código a ser usada em um procedimento ou chamada, mas possibilita uma economia de tempo nas codificações como, por exemplo, calculando o fatorial de um número. “Um método recursivo pode chamar outro método, que por sua vez pode fazer uma chamada de volta ao método recursivo. Isso é conhecido como uma chamada recursiva indireta ou recursão indireta. Por exemplo, o método A chama o método B, que faz uma chamada de volta ao método A. Isso ainda é recursão, porque a segunda chamada para o método A é feita enquanto a primeira está ativa – isto é, a primeira chamada ao método A ainda não concluiu sua execução (porque está esperando o método B voltar um resultado para ela) e não retornou ao chamador original do método A” (DEITEL; DEITEL, 2016, p. 609).

DEITEL, P.; DEITEL, H. Java: como programar. 10. ed. São Paulo: Pearson Education do Brasil, 2016.

Observe o trecho de código a seguir, realizado na linguagem Java, versão 8.

```
public static BigInteger fibonacci(BigInteger number)
{
    if (number.equals(BigInteger.ZERO) ||
        number.equals(BigInteger.ONE)) // casos básicos
        return number;
    else // passo de recursão
        return fibonacci(number.subtract(BigInteger.ONE)).add(
            fibonacci(number.subtract(TWO)));
}
```

Fonte: Elaboração do autor, 2023.

Sobre o código anterior, avalie as afirmativas a seguir.

I. A codificação mostrada possui erro sintático, pois, nessa chamada recursiva, o método Fibonacci está chamando outro método para cálculo de mesmo nome, retornando uma chamada também com o mesmo nome de método.

II. A condição “number.equals(BigInteger.ZERO)” permite que todo trecho de código não necessite ser executado, uma vez que a estrutura “if...else” está limitando a execução caso o “if” inicial seja verdadeiro.

III. É possível realizar operações matemáticas por meio de métodos com cálculos predefinidos, seja para comparação, subtração ou adição numérica, passando como parâmetro o valor desejado.

Está correto o que se afirma em:

Resposta Selecionada: 

b. II e III, apenas.

Respostas: 

a. I, II e III, apenas.

b. II e III, apenas.

c. I, apenas.

d. I e III, apenas.

e. I e II, apenas.

Comentário da resposta: JUSTIFICATIVA

A afirmativa I está incorreta, pois não se apresenta erro sintático pelo motivo apresentado. Ao realizar uma chamada para si mesmo, o método atende a uma característica de desenvolvimento chamada recursão.

A afirmativa II está correta, uma vez que existem métodos predefinidos disponíveis para serem usados, e, nesse caso, a condicional realiza um procedimento limitante no bloco condicional, podendo ou não ser executado conforme o valor obtido.

A afirmativa III está correta, pois, como mostrado no código, é possível realizar operações matemáticas e lógicas por meio de métodos com cálculos predefinidos, seja para comparação, subtração ou adição.

Quinta-feira, 15 de Agosto de 2024 21h57min04s BRT

← OK