



LINGUAGEM DE PROGRAMAÇÃO ORIENTADA À OBJETOS 7967-90_43701_R_E1_20231

CONTEÚDO

Revisar envio do teste: QUESTIONÁRIO UNIDADE I

Usuário	
Curso	LINGUAGEM DE PROGRAMAÇÃO ORIENTADA À OBJETOS
Teste	QUESTIONÁRIO UNIDADE I
Iniciado	06/03/23 22:02
Enviado	06/03/23 22:35
Status	Completada
Resultado da tentativa	1,75 em 2,5 pontos
Tempo decorrido	32 minutos
Resultados exibidos	Todas as respostas, Respostas enviadas, Respostas corretas, Comentários, Perguntas respondidas incorretamente

Pergunta 1

0,25 em 0,25 pontos



Tendo-se as seguintes declarações de métodos:

```
public int metodo_01 (String x) { ... }  
public String metodo_02 (int x) { ... }  
public double metodo_03 (double x) { ... }  
public void metodo_04 (String x, int y) { ... }  
public int metodo_05 ( ) { ... }
```

Qual das alternativas mostra uma atribuição que não pode ser efetuada?

Resposta Selecionada: ☒ c. int k = metodo_04 ("José", 1363).

Respostas:

a. String k = metodo_02 (1363).

b. int k = metodo_05 ().

☒ c. int k = metodo_04 ("José", 1363).

d. int k = metodo_01 ("24").

e. double k = metodo_03 (5.45).

Comentário Resposta: C

da resposta: Comentário: o método "metodo_04" é um método que não tem retorno de informação, já que seu "tipo de retorno" é "void", o que significa que não podemos atribuir sua chamada a uma variável (já que ela não devolverá valor algum).

Pergunta 2

0,25 em 0,25 pontos



Analise as afirmações e selecione a alternativa correta:

I – Um objeto é um elemento gerado em memória a partir de um modelo, cujas características são previamente estabelecidas no programa fonte.

II – Quando criamos uma Classe, ao codificarmos um sistema, definimos todas as suas possíveis ações por seus atributos e seus possíveis dados por seus métodos.

III – O método "main" é aquele que inicializa a execução de um sistema gerado na linguagem Java, no entanto uma classe não depende do método "main" para ser utilizada.

IV – A Classe é um elemento instanciado que possui atributos e métodos.

V – Uma variável genérica é aquela que pode receber qualquer tipo de dado, porém seu tipo não pode ser alterado.

De acordo com as afirmações, quais estão corretas?

Resposta Selecionada: ☒ d. I e III.

Respostas:

a. II e IV.

b. I, II e V.

c. I, IV e V.

☒ d. I e III.

e. II, III e IV.

Comentário da resposta: D
Comentário: o objeto é a Classe instanciada (criada e existente) em memória, de forma que possui todas as características daquela Classe, como seus atributos e seus métodos. Ao codificarmos uma Classe (quando criamos o programa fonte), estamos gerando a possibilidade da existência de um "objeto" em memória que pode conter alguns dados (a partir dos atributos existentes naquela Classe) e do qual podemos executar algumas ações (a partir dos métodos que a Classe possui). Devemos lembrar que a linguagem Java é uma linguagem "fortemente tipada", em que toda variável é criada já com um tipo de dado definido em seu código e que não pode ser alterado ao longo do programa. Essa característica impede a existência de variáveis genéricas, assim como é possível em outras linguagens, como por exemplo o Python.

Pergunta 3

0,25 em 0,25 pontos



Analisando o programa a seguir, selecione a alternativa que mostra o texto que será impresso na tela da console após acionarmos a execução da Classe:

```
public class Teste {  
    public static void main(String[] args) {  
        int a = 5, b = 5, c = 0;  
        do {  
            c += a * b;  
            a++;  
        } while (a < b);  
        System.out.println(a + ", " + b + ", " + c);  
    }  
}
```

Resposta Selecionada: ☒ a. 6, 5, 25.

Respostas: ☒ a. 6, 5, 25.

☐ b. 6, 5, 30.

☐ c. 5, 6, 35.

☐ d. 6, 6, 36.

☐ e. 5, 5, 0.

Comentário da resposta: A
Comentário: os valores iniciais das variáveis a, b e c são, respectivamente, 5, 5 e 0. A partir da estrutura de repetição utilizada, sabemos que seu bloco de código será executado ao menos uma vez (já que estamos utilizando a estrutura "do-while"), o que significa que o valor inicial da variável "c" será acrescido (soma recursiva) do valor da multiplicação de "a" por "b" (ou seja, 5 * 5, que resulta em 25), e será somente após esta conta que o valor da variável "a" será alterado (acrescentado de uma unidade, ou seja, alterado de 5 para 6). Observa-se que em nenhum momento o valor da variável "b" é alterado, de forma que, ao final da primeira iteração (do bloco do-while), o valor de "a" (6) será maior que o de "b" (5), e com isso a iteração não se repetirá, pois a condição de repetição ("a < b") será falsa e, ao sair do bloco, os valores finais de a, b e c são, respectivamente, 6, 5 e 25.

Pergunta 4

0 em 0,25 pontos



Sobre as estruturas condicionais, analise as afirmações e selecione a alternativa correta:

I – A estrutura "if – else" é uma estrutura condicional simples controlada por apenas uma condição, o que significa que seu bloco será executado somente se aquela condição for verdadeira.

II – Qualquer bloco de comparação construído a partir da lógica "if – else if" pode ser substituído por um outro bloco correspondente construído a partir da lógica "switch – case".

III – A estrutura condicional "switch – case" somente aceita lógicas de comparação baseadas na comparação de igualdade.

IV – O bloco "default" da estrutura "switch – case" tem sua funcionalidade equivalente ao bloco "else" da estrutura "if – else if", de modo que somente será executado se todas as outras comparações existentes na estrutura tiverem valor "falso".

De acordo com as afirmações, quais estão corretas?

Resposta Selecionada: ☒ d. III e IV.

Respostas: ☐ a. II e IV.

☐ b. I, II e III.

☐ c. I e IV.

☐ d. III e IV.

☒ e. I, III e IV.

Pergunta 5

0,25 em 0,25 pontos



Sobre as estruturas de repetição, analise as afirmações e selecione a alternativa correta:

- I – Uma estrutura de repetição é aquela que permite que um mesmo bloco de código seja executado mais de uma vez, mas sempre controlado por uma condição.
- II – A estrutura "for" de repetição possui uma variável de controle cujo valor é sempre verificado por uma comparação que acontece antes da execução de uma iteração.
- III – A estrutura "while" possui uma condição que é verificada apenas no final da execução de uma iteração, diferentemente da estrutura "do – while", que verifica antes da execução da iteração.
- IV – É sempre possível converter uma estrutura "for" em uma estrutura "while", porém um bloco construído com uma estrutura "while" não pode ser convertido utilizando-se a estrutura "for".

De acordo com as afirmações, quais estão corretas?

Resposta Selecionada: ☒ b. I e II.

Respostas: a. II e III.

☒ b. I e II.

c. II, III e IV.

d. I e IV.

e. III e IV.

Comentário da resposta:

Resposta: B
Comentário: todas as estruturas de repetição são controladas por uma condição, de forma que a execução do seu bloco (a iteração) se dá enquanto essa condição for verdadeira. Essa regra vale também para a estrutura "for", já que a condição faz parte da declaração de sua estrutura. Em relação às estruturas "while" e "do – while", a diferença entre elas está na verificação da condição, de forma que na estrutura "while", essa verificação se dá antes da iteração. Na estrutura "do – while", essa verificação se dá depois da iteração. Quanto à conversão das estruturas, sempre é possível converter entre as estruturas "for" e "while", porém para se fazer a conversão de uma estrutura "while" para uma estrutura "for", apesar de ser possível convertermos qualquer forma de comparação, costumamos utilizar a estrutura "for" apenas nos casos em que conhecemos a quantidade de iterações que o programa executará.

Pergunta 6

0,25 em 0,25 pontos



O que significa o termo (linha de código) "package com;", descrito no início do código de uma classe?

Resposta Selecionada: ☒ c. Significa que essa classe está localizada em um pacote cujo nome é simplesmente "com".

Respostas:

a.

Significa que essa classe está localizada em um pacote cujo nome possui o termo "com" (como por exemplo "com.model").

b. Significa que essa classe está utilizando classes que pertencem ao pacote "com".

☒ c. Significa que essa classe está localizada em um pacote cujo nome é simplesmente "com".

d. Significa que essa classe está utilizando classes que pertencem a pacotes cujo nome possuem o termo "com".

e. Significa que essa classe se utiliza de classes localizadas em pacotes, independentemente do nome desse pacote.

Comentário da resposta:

Resposta: C
Comentário: quando um sistema está organizado em pacotes, suas classes estão distribuídas nesses pacotes, que "fisicamente" seria o equivalente a uma organização de arquivos em diversos diretórios em que em cada um deles colocamos arquivos que se encaixam em uma determinada categoria. Sendo assim, a linguagem Java exige (para seu próprio controle) que, quando uma Classe está localizada em um pacote, este esteja definido (declarado) no início de seu código, de forma que o comando "package" indica ao compilador a qual pacote aquela classe pertence.

Pergunta 7

0,25 em 0,25 pontos



Têm-se as seguintes classes a seguir:

```
package pacoteA;  
public class ClasseA {  
    public static void main (String[] args) {  
        //linha de comando A  
    }  
}
```

```

}

package pacoteB;
public class ClasseB {
    public static void main (String[] args) {
        //linha de comando B
    }
}

```

A partir dessas duas classes, analise as seguintes afirmações:

- I – Sem realizarmos alterações extras no código, o comando "ClasseA ca;" não pode ser inserido em substituição da "//linha de comando B".
- II – Sem realizarmos alterações extras no código, o comando "ClasseA ca;" não pode ser inserido em substituição da "//linha de comando A".
- III – Sem realizarmos alterações extras no código, o comando "ClasseB cb;" não pode ser inserido em substituição da "//linha de comando A".
- IV – Podemos inserir o comando "ClasseA ca;" em substituição à "//linha de comando A" sem precisarmos realizar alterações extras no código.
- V – Para que possamos inserir o comando "ClasseB cb;" em substituição à "//linha de comando A", precisaríamos realizar alterações extras no código, como por exemplo a inclusão da linha de comando "import pacoteB.*;" antes da declaração da classe.
- VI – Para que possamos inserir o comando "ClasseA ca;" em substituição à "//linha de comando A", precisaríamos realizar alterações extras no código, como por exemplo a inclusão da linha de comando "import pacoteA.*;" antes da declaração da classe.

De acordo com as afirmações, quais estão corretas?

Resposta Selecionada: ☒ a. I, III, IV e V.

Respostas: ☒ a. I, III, IV e V.

b. I, II e III.

c. II, III e IV.

d. III, IV e VI.

e. II e VI.

Comentário da resposta: Resposta: A
Comentário: para que possamos utilizar (declarar ou instanciar) classes que pertencem a outros pacotes, precisamos importá-las antes da declaração da classe (pelo comando "import"). Por meio do comando "import", podemos importar especificamente uma classe ("import pacote.NomeDaClasse;") ou podemos importar todas as classes daquele pacote utilizando o "asterisco" ("import pacote.*;").

Pergunta 8

0 em 0,25 pontos



Imagine as seguintes linhas de comandos:

```

public class ClasseA {
    public static void main (String[] args) {
        int a = 0, b = 0, c = 0;
        int d = ?;
        if (d > 100) {
            a = 2;
        } else if (d > 50) {
            b = 2;
        } else if (d > 20) {
            c = 2;
        }
    }
}

```

Analisando esse código, ao final de sua execução, para quais valores de "d", apresentados a seguir, seria possível termos os valores das variáveis a, b e c alterados para o número 2?

Resposta Selecionada: ☒ a.

As 3 variáveis terão seus valores alterados para 2 se a variável d inicialmente tiver um valor acima de 100 (como por exemplo o valor 200).

Respostas: ☐ a.

As 3 variáveis terão seus valores alterados para 2 se a variável d inicialmente tiver um valor acima de 100 (como por exemplo o valor 200).

b. As 3 variáveis terão seus valores alterados para 2 se a variável d inicialmente tiver um valor igual a 100.

c. As 3 variáveis terão seus valores alterados para 2 se a variável d inicialmente tiver um valor igual a 10.

☒ d.

No código apresentado não é possível alterarmos o valor das 3 variáveis a, b e c, mas sim apenas de uma delas, dependendo do valor inicial da variável "d".

e. As 3 variáveis terão seus valores alterados para 2 se a variável d inicialmente tiver um valor igual a 0.

Pergunta 9

0 em 0,25 pontos



Imagine as seguintes linhas de comandos:

```
public class ClasseA {
    public static void main (String[] args) {
        int a = 0, b = 0, c = 0;
        int d = 300;
        if (d > 100) {
            a = 2;
        } else if (d > 50) {
            b = 2;
        } else if (d > 20) {
            c = 2;
        }
    }
}
```

Analisando o código anterior, qual das alterações permite que, ao final de sua execução, seja possível termos os valores das variáveis a, b e c alterados para o número 2?

Resposta Selecionada: ☒ d. Alterar a estrutura condicional para a estrutura "switch – case".

Respostas:

- ☐ a. Alterar o valor inicial de "d" para 0 (zero).
- ☒ b. Retirar as palavras "else" que aparecem na estrutura.
- ☐ c. Alterar o operador de comparação para "<".
- ☐ d. Alterar a estrutura condicional para a estrutura "switch – case".
- ☐ e. Alterar a estrutura condicional para uma estrutura de repetição.

Pergunta 10

0,25 em 0,25 pontos



Analise o código a seguir e selecione a opção que mostra qual o texto que será impresso na tela da console, após acionarmos a classe a seguir:

```
public class ClasseB {
    public static void main (String[] args) {
        for (int x = 2; x <=20; x += 4) {
            System.out.print((x/2) + " ");
        }
    }
}
```

Resposta Selecionada: ☒ e. 1 3 5 7 9

Respostas:

- ☐ a. 2 6 10 14 18 22
- ☐ b. 2 4 6 8 10 12 14 16 18
- ☐ c. 1 2 3 4 5 6 7 8 9 10
- ☐ d. 3 6 9 12 15 18
- ☒ e. 1 3 5 7 9

Comentário Resposta: E

da
resposta: Comentário: o programa anterior poderá ser acionado, pois é um método "main". Ao ser acionado, será iniciado um *looping* de 5 iterações em que a variável "x" terá, ao longo de suas iterações, os seguintes valores: 2, 6, 10, 14 e 18 (isso por que essa variável inicia no valor 2 e vai sendo incrementada de 4 unidades a cada iteração, de forma a permitir uma nova iteração até que seu valor se torne maior que 20, quando cessa a repetição). No entanto, o valor impresso é a metade do valor de x (no caso, 2/2=1, 6/2=3, 10/2=5, 14/2=7 e 18/2=9).