

Nome: _____

Matrícula: _____

Disciplina: ARA0075 / PROGRAMAÇÃO ORIENTADA A OBJETOS EM JAVA

Data: __/__/__

Período: 2025.2 / SM1

Turma: 3006 NSQ: 13772816

Leia com atenção as questões antes de responder.

É proibido o uso de equipamentos eletrônicos portáteis e consulta a materiais de qualquer natureza durante a realização da prova.

Boa prova.

1.

_____ de 0,10

Julgue as afirmativas a seguir:

I-Wrappers em Java possuem a função de envolver as coisas, ou seja, adiciona funcionalidades às classes

II-Um dos Wrappers disponíveis no Java é o Integer, que é um Wrapper do tipo primitivo

III-Com o Wrapper é possível, por exemplo, adicionar métodos que podem tratar tipos primitivos como classes

Estão corretas somente:

- A ☐ I
- B ☐ I, III
- C ☐ I, II
- D ☒ Todas as afirmações
- E ☐ II, III

2.

_____ de 0,10

Sabemos que quando há herança entre classes, as subclasses podem reescrever os métodos da superclasse. Então como se chama o processo quando uma subclasse reescreve um método da superclasse e mantém sua assinatura?

- A ☐ synchronized
- B ☐ polimorfismo
- C ☐ encapsulamento
- D ☐ overloading
- E ☒ overriding

3.

_____ de 0,10

Considerando o trecho a seguir na linguagem Java, observe as afirmativas a seguir:

```
1. try {  
2.   // Código ...  
3. }  
4. catch (Exception e) {  
5.   // Código ...  
6. }
```

I - Se ocorrer uma exceção no bloco do try, então a execução é automaticamente desviada para o bloco catch.

II - No catch devemos definir a exceção a ser tratada. Quando definimos uma exceção estamos tratando também todas

as suas subclasses.

III - O "e" mostrado na linha do catch, referencia a exceção que ocorreu, sendo possível acessar informações sobre essa exceção.

Está(ão) correta(s) a(s) afirmativa(s) :

- A ☐ Apenas a I.
- B ☐ I e III.
- C ☒ I, II e III.
- D ☐ II e III.
- E ☐ I e II.

4.

_____ de 0,10

Quando algo dá errado durante a execução de um programa, ocorre uma exceção em tempo de execução. Exceções interrompem o fluxo do programa e, caso nada seja feito, o programa para de executar. O programador deve permitir que o programa lide com esses problemas sem travar a sua execução.

Analise o exemplo abaixo:

```
public classe AV {  
    public static void main(String[] args) {  
        String var = "ABC";  
        Try {  
            Integer i = new Integer(var);  
            System.out.print("A variável i vale: "+i);  
        } catch (NumberFormatException nfe) {  
            System.out.println("Não foi possível atribuir a String " + var +  
                " a um objeto inteiro. " + "\n" +  
                " A seguinte mensagem foi retornada: " +  
                Nfe.getMessage());  
        }  
    }  
}
```

Considerando o tratamento de exceção e o código acima, é correto afirmar que:

- A ☐ Em Java, assim como em C, pode-se acessar uma posição de um *array* (vetor) que esteja fora dos limites declarados. Assim, em Java não é necessário fazer o tratamento de exceção para limites de *arrays*.
- B ☐ Deve-se usar o tratamento de exceção estruturado *On Error*, que é o método recomendado, pois torna o tratamento de erro fácil de ser depurado
- C ☐ Deve-se fazer o tratamento de exceções utilizando as cláusulas *Try*, *Except* e *Finally* juntas, sempre. Mas, se o bloco que estiver em *Except* contiver a exceção, o programa será terminado.
- D ☐ Deve-se usar as palavras-chave *try* e *catch* junto com *finally*, obrigatoriamente. Usa-se *try* e *catch* para delimitar o bloco de código que pode gerar uma exceção e *finally* para conter o código que será executado se a exceção ocorrer
- E ☒ Ao ser executado, o programa Java acima retorna: Nao foi possivel atribuir a string ABC a um objeto inteiro. A seguinte mensagem foi retornada:For input string: "ABC"

5.

_____ de 0,10

Na hierarquia de exceções em Java, é correto afirmar que

- A ☐ as classes *NullPointerException* e *ArrayIndexOutOfBoundsException* não são válidas no tratamento de exceção em Java.
- B ☐ a classe *Error* herda da classe *Exception*.
- C ☐ a classe *Exception* e a classe *Error* são subclasses da classe *NullPointerException*.
- D ☒ a classe *RuntimeException* é uma subclasse da classe *Exception*.
- E ☐ a classe *Exception* é uma subclasse da classe *IOException*.

6.

_____ de 0,10

Sobre a programação orientada a objetos com Java, é correto afirmar:

- A ☐ No interior do método *main* de uma classe, mesmo os métodos públicos e estáticos de outra classe só podem ser acessados por meio de um objeto explicitamente instanciado dessa classe
- B ☐ Em uma hierarquia de herança, a superclasse herda todos os métodos públicos, privados e protegidos das subclasses
- C ☐ Uma interface possui métodos com conteúdo implementado que não precisam ser reescritos nas classes que a implementam
- D ☐ É obrigatório escrever em toda classe um construtor vazio que não recebe parâmetros. Além desse construtor, pode-se criar outros construtores vazios
- E ☒ Uma classe pode ter mais de um método com o mesmo nome, desde que receba parâmetros diferentes

7.

_____ de 0,10

Como será o construtor com atributos de uma Classe Aluno possuindo nome e matricula?

- A ☐ public Aluno{}
- B ☐ public void Aluno(nome, matricula){}
- C ☐ public void Aluno(String nome, String matricula){...}
- D ☐ public void Aluno(){}
- E ☒ public Aluno(String nome, String matricula){...}

8.

_____ de 0,10

A reusabilidade de código é uma das vantagens de linguagens orientadas a objetos como Java. Escolha a opção correta em que temos a palavra reservada que permite o programador aplicar a reusabilidade de código.

- A ☐ public
- B ☐ class
- C ☐ import
- D ☒ extends
- E ☐ implements

9.

_____ de 0,10

Analise o seguinte trecho de programa e marque a alternativa que demonstra os valores impressos pelo programa:

```
public class Avalia {  
    public static void main(String args[]) {  
        int i;  
        int v [] = {5,12,18,3,5,66,7,23,4,67};  
        for (i=1; i<10; i++){  
            if ( (i%2) == 0 )  
                System.out.println (v[i]);  
        }  
    }  
}
```

- A ☐ 12 - 3 - 66 - 23 - 67
- B ☐ 5 - 18 - 5 - 7 - 4
- C ☐ 12 - 18 - 66 - 4
- D ☒ 18 - 5 - 7 - 4
- E ☐ 18 - 66 - 7 - 4

10.

_____ de 0,10

Seja o código a seguir:

```
public class DemoVetor{  
    public static void main(String[] args) {  
        int[] vet={10,20,34,56,60};  
        int i;  
        for(i=0; i < vet.length;i++) {  
            System.out.println("vet["+i+"]=" + vet[i]);  
        }  
        System.out.print("i="+i);  
    }  
}
```

Marque a alternativa que apresenta o valor da variável i, ao final do programa

- A ☐ i = 4;
- B ☐ i = 6;
- C ☐ i = 1;
- D ☐ i = 3;
- E ☒ i = 5;

Campus:
POLO UNIRUY PARALELA - SALVADOR - BA

Prova Impressa em 06/10/2025 por
HELENO CARDOSO DA SILVA FILHO

Ref.: 7064251463

Prova Montada em 18/09/2025