Questões Java SE

Prof. Rodrigo Macedo

Escopo do Curso

- Questões sintaxe da linguagem.
- Questões tipos primitivos.
- Questões orientação a objetos.
- Questões coleções Java.
- Questões lógica de programação.

Tipos Primitivos

		Valores possíveis				
Tipos	***Primitivo***	***Menor***	***Maior***	***Valor Padrão***	***Tamanho***	***Exemplo***
Inteiro	byte	-128	127	0	8 bits	byte ex1 = (byte)1;
	short	-32768	32767	0	16 bits	short ex2 = (short)1;
	int	-2.147.483.648	2.147.483.647	0	32 bits	int ex3 = 1;
	long	-9.223.372.036.854.770.000	9.223.372.036.854.770.000	0	64 bits	long ex4 = 1l;
Ponto Flutuante	float	-1,4024E-37	3.40282347E + 38	0	32 bits	float ex5 = 5.50f;

	double	-4,94E-307	1.79769313486231570E + 308	0	64 bits	double ex6 = 10.20d;
						ou
						double ex6 = 10.20;

Caractere	char	0	65535	\0	16 bits	char ex7 = 194; ou char ex8 = 'a';
Booleano	boolean	false	true	false	1 bit	boolean ex9 = true;

Modificadores de Acesso

Visibilidade	public	protected	default	private
A partir da mesma classe	✓	✓	✓	✓
Qualquer classe no mesmo pacote	✓	✓	✓	⊘
Qualquer classe filha no mesmo pacote	✓	✓	✓	⊘
Qualquer classe filha em pacote diferente	✓	✓	⊘	⊘
Qualquer classe em pacote diferente	✓	⊘	⊘	⊘

Declaração array

```
int [ ] num = new int [ 10 ];
```

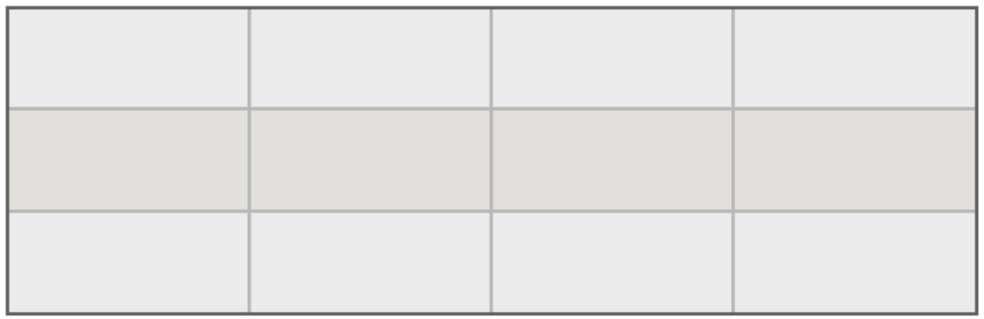
type of each element

name of array

subscript (integer or constant expression for number of elements.)

Array Unidimensional

Array Multidimensional



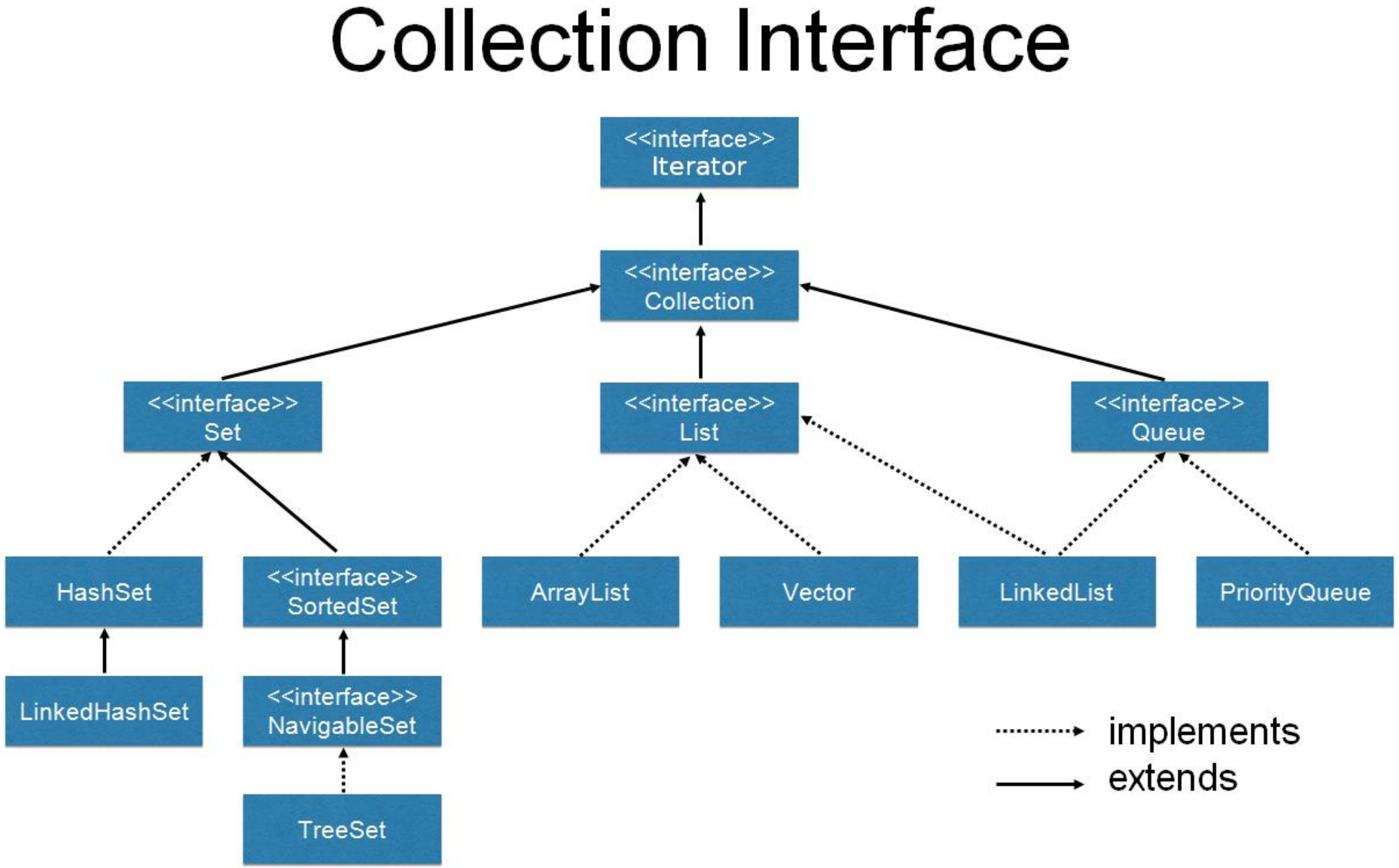
int[][] array = new int[3][4]

Criação e Inicialização de array

- Exemplo de criação e instanciação de um array:

```
int[ ] x = {1, 2, 3, 4, 5};
int[ ] x = new int[ ]{1, 2, 3, 4, 5};
```

Coleções Java



Lista

- Uma lista é uma coleção de elementos arrumados numa ordem **linear**, isto é, onde cada elemento tem um antecessor (exceto o primeiro) e um sucessor (exceto o último).
- Permite objetos duplicados.
- Normalmente implementada como "Array" ou "Lista Encadeada"

Métodos:

- add insere um elemento no final da coleção
- clear tira todos os elementos
- size retorna o número de elementos
- isEmpty testa se a coleção é vazia
- toArray move os elementos para um Array

Q1) [CESPE TJ PA 2020] Na programação orientada a objetos, a herança é uma técnica de abstração que permite categorizar as classes de objetos sob certos critérios, especificando-se as características dessas classes. As classes que são vinculadas por meio de relacionamentos de herança formam uma hierarquia de herança. Na linguagem de programação Java, o relacionamento de herança é definido pela palavra-chave

a) static.

b) extends.

c) public.

d) new.

e) this.

Q1) [CESPE TJ PA 2020] Na programação orientada a objetos, a herança é uma técnica de abstração que permite categorizar as classes de objetos sob certos critérios, especificando-se as características dessas classes. As classes que são vinculadas por meio de relacionamentos de herança formam uma hierarquia de herança. Na linguagem de programação Java, o relacionamento de herança é definido pela palavra-chave

a) static.

b) extends.

c) public.

d) new.

e) this.

Q2) [CESPE TJ PA 2020]

Assinale a opção que apresenta corretamente a saída gerada pelo código Java precedente.

```
class GeraNumeros {
   public static void main (String args []) {
      int num;
      num = 36;
      for(int i=0; i < num; i++) {
        if(i*i >= num) break;
        System.out.print(i + " ");
      }
    }
}
```

a) 36.

b) 1 2 3 4 5 6.

c) 1 2 3 4 5.

d) 0 1 2 3 4 5.

e) 0 1 2 3 4 5 6

Q2) [CESPE TJ PA 2020]

Assinale a opção que apresenta corretamente a saída gerada pelo código Java precedente.

```
class GeraNumeros {
   public static void main (String args []) {
     int num;
     num = 36;
     for(int i=0; i < num; i++) {
       if(i*i >= num) break;
       System.out.print(i + " ");
     }
    }
}
```

a) 36.

b) 1 2 3 4 5 6.

c) 1 2 3 4 5.

d) 0 1 2 3 4 5.

e) 0 1 2 3 4 5 6

I1 -> 0 * 0 = Resultado 0
I2 -> 1 * 1 = Resultado 1
I3 -> 2 * 2 = Resultado 4
I4 -> 3 * 3 = Resultado 9
I5 -> 4 * 4 = Resultado 16
I6 -> 5 * 5 = Resultado 25
I7 -> 6 * 6 = Resultado 36 - Entra no if.

Q3) [IBADE IPM-JP 2018] Java disponibiliza operadores aritméticos e relacionais para uso. Nesse sentido, considere as três situações caracterizadas a seguir.

I. Avaliar o valor do operador w e após incrementar em uma unidade.

II. Retornar TRUE se x é diferente de y.

III. Obter o resto da divisão de x por y. As sintaxes empregadas nas situações I, II e III são, respectivamente:

a) ++w, x <> y e x $ y.

b) ++w, x != y e x % y.

c) w++, x <> y e x % y.

d) w++, x != y e x % y.

e) w++, x != y e x $ y.

Q3) [IBADE IPM-JP 2018] Java disponibiliza operadores aritméticos e relacionais para uso. Nesse sentido, considere as três situações caracterizadas a seguir.

I. Avaliar o valor do operador w e após incrementar em uma unidade.

II. Retornar TRUE se x é diferente de y.

III. Obter o resto da divisão de x por y. As sintaxes empregadas nas situações I, II e III são, respectivamente:

a) ++w, x <> y e x $ y.

b) ++w, x != y e x % y.

c) w++, x <> y e x % y.

d) w++, x != y e x % y.

e) w++, x != y e x $ y.

Q4) [IBADE IF-RO 2019] Os modificadores de acesso são padrões de visibilidade de acessos às classes, atributos e métodos. Esses modificadores são palavras-chave reservadas pelo Java, ou seja, palavras reservadas não podem ser usadas como nome de métodos, classes ou atributos. Os modificadores de acesso são classificados conforme as descrições abaixo:

(1) indica que método ou variável só podem ser acessados de dentro da classe que os criou. Uma classe que herde de uma superclasse com atributos declarados de acordo com esse modificador só poderá ter acesso a eles através dos métodos públicos da própria superclasse, caso contrário, não haverá acesso a estes atributos.

(2) indica que o método ou a variável assim declarada possa ser acessada somente dentro do pacote em que está contida através de uma subclasse.

(3) indica que a classe, método ou variável assim declarada possa ser acessada em qualquer lugar e a qualquer momento da execução do programa.

Os modificadores de acesso descritos como (1), (2) e (3) são denominados, respectivamente:

a) protected, public e static.

b) dynamic, private, protected.

c) public, static e dynamic.

d) private, protected e public.

e) static, dynamic e private.

Q4) [IBADE IF-RO 2019] Os modificadores de acesso são padrões de visibilidade de acessos às classes, atributos e métodos. Esses modificadores são palavras-chave reservadas pelo Java, ou seja, palavras reservadas não podem ser usadas como nome de métodos, classes ou atributos. Os modificadores de acesso são classificados conforme as descrições abaixo:

(1) indica que método ou variável só podem ser acessados de dentro da classe que os criou. Uma classe que herde de uma superclasse com atributos declarados de acordo com esse modificador só poderá ter acesso a eles através dos métodos públicos da própria superclasse, caso contrário, não haverá acesso a estes atributos.

(2) indica que o método ou a variável assim declarada possa ser acessada somente dentro do pacote em que está contida através de uma subclasse.

(3) indica que a classe, método ou variável assim declarada possa ser acessada em qualquer lugar e a qualquer momento da execução do programa.

Os modificadores de acesso descritos como (1), (2) e (3) são denominados, respectivamente:

a) protected, public e static.

b) dynamic, private, protected.

c) public, static e dynamic.

d) private, protected e public.

e) static, dynamic e private.

Q5) [CS-UFG UFG 2019] Para clonar (deep copy) um vetor (array) de inteiros (int) em Java emprega-se a seguinte estratégia:

a) chamar o método clone() usando a referência para o vetor.

b) implementar a interface Comparable.

c) fornecer o vetor como argumento para o método java.util.Objects.deepCopy.

d) implementar a interface Serializable.

Q6) [FUNDEP INB 2018] Em Java, para identificar quantos caracteres existem em uma string, utiliza-se o método:

a) charIn()

b) charAt()

c) length()

d) size()

Q5) [CS-UFG UFG 2019] Para clonar (deep copy) um vetor (array) de inteiros (int) em Java emprega-se a seguinte estratégia:

a) chamar o método clone() usando a referência para o vetor.

b) implementar a interface Comparable.

c) fornecer o vetor como argumento para o método java.util.Objects.deepCopy.

d) implementar a interface Serializable.

Q6) [FUNDEP INB 2018] Em Java, para identificar quantos caracteres existem em uma string, utiliza-se o método:

a) charIn()

b) charAt()

c) length()

d) size()

Q7) [FUNDEP INB 2018] Qual é a sintaxe correta para se criar uma classe abstrata em Java?

a) Abstract class nome.

b) abstract class nome

c) class nome abstract

d) Class nome abstract.

Q8) [**FUNDEP INB 2018**] Assinale a linha em Java que não apresenta erro de sintaxe.

a) return int salario;

b) return salario;

c) return Int salario, Char nome;

d) return char nome;

Q9) [QUADRIX CRO-AC 2019] Os tipos de dados float e double, em Java, são destinados a armazenar números reais.

Q7) [FUNDEP INB 2018] Qual é a sintaxe correta para se criar uma classe abstrata em Java?

a) Abstract class nome.

b) abstract class nome

c) class nome abstract

d) Class nome abstract.

Q8) [**FUNDEP INB 2018**] Assinale a linha em Java que não apresenta erro de sintaxe.

a) return int salario;

b) return salario;

c) return Int salario, Char nome;

d) return char nome;

Q9) [QUADRIX CRO-AC 2019] Os tipos de dados float e double, em Java, são destinados a armazenar números reais. CERTO.

Q10) [QUADRIX CRO-AC 2019] As instruções if e switch são tipos de instruções de repetição utilizadas pela Linguagem Java.

Q11) [QUADRIX CRO-AC 2019] Na linguagem de programação Java, um array é um tipo de dado elementar e, para que um elemento seu seja referenciado, basta especificar o número de posição do elemento no array.

Q12) [QUADRIX CPM-PR] O método main (linha 3) usa o parâmetro String args[], que possibilita a execução do programa com passagem de parâmetros pela linha de comando.

Q13) [AOCP Prefeitura MG 2016] Na programação orientada a objeto, a habilidade de estender de mais de uma classe é conhecida como herança múltipla. Na linguagem de programação Java, há uma restrição para se utilizar a herança múltipla. Sendo assim, em Java, a herança múltipla só é permitida para

a) classes.

b) atributos

c) métodos.

d) interfaces

e) permissões.

Q10) [QUADRIX CRO-AC 2019] As instruções if e switch são tipos de instruções de repetição utilizadas pela Linguagem Java. ERRADO.

Q11) [QUADRIX CRO-AC 2019] Na linguagem de programação Java, um array é um tipo de dado elementar e, para que um elemento seu seja referenciado, basta especificar o número de posição do elemento no array. ERRADO.

Q12) [QUADRIX CPM-PR] O método main (linha 3) usa o parâmetro String args[], que possibilita a execução do programa com passagem de parâmetros pela linha de comando. CERTO.

Q13) [AOCP Prefeitura MG 2016] Na programação orientada a objeto, a habilidade de estender de mais de uma classe é conhecida como herança múltipla. Na linguagem de programação Java, há uma restrição para se utilizar a herança múltipla. Sendo assim, em Java, a herança múltipla só é permitida para

a) classes.

b) atributos

c) métodos.

d) interfaces

e) permissões.

Q14) [AOCP Prefeitura MG 2016] Analise o código escrito em Java apresentado a seguir.

Sobre o código, é correto afirmar que

```
01. public class Questao36
02. {
03.   public static void main( String[] args )
04.   {
05.     int cont;
06.     for ( cont = 1; cont <= 10; cont++)
07.        {
08.           if ( cont == 5 )
09.                break;
10.           System.out.printf( "%d", cont );
11.         }
12.      System.out.printf( "Loop parou no número = %d\n", cont);
13.   }
14. }
```

a) ao final da execução será apresentada na tela a sequência numérica de 1 até 4 e uma mensagem de aviso que sequência parou no número 5.

b) o programa dará erro porque nas linhas 05, 06, 08, 10 e 12 o nome da variável deveria ser "count" e não "cont", como apresentado.

c) ao final da execução será apresentada na tela toda a sequência numérica iniciando de 1 e terminando em 10, com a mensagem de aviso de parada no número 5.

d) a implementação do laço do programa deveria ser implementada com o "while", porque ele permite a utilização do comando "break" enquanto o "for" não permite.

e) ao final da execução será apresentada na tela toda a sequência numérica de 1 a 10 com a mensagem de aviso de parada para cada um dos números do laço "for".

Q14) [AOCP Prefeitura MG 2016] Analise o código escrito em Java apresentado a seguir.

Sobre o código, é correto afirmar que

a) ao final da execução será apresentada na tela a sequência numérica de 1 até 4 e uma mensagem de aviso que sequência parou no número 5.

b) o programa dará erro porque nas linhas 05, 06, 08, 10 e 12 o nome da variável deveria ser “count” e não “cont”, como apresentado.

c) ao final da execução será apresentada na tela toda a sequência numérica iniciando de 1 e terminando em 10, com a mensagem de aviso de parada no número 5.

d) a implementação do laço do programa deveria ser implementada com o “while”, porque ele permite a utilização do comando “break” enquanto o “for” não permite.

e) ao final da execução será apresentada na tela toda a sequência numérica de 1 a 10 com a mensagem de aviso de parada para cada um dos números do laço “for”.

```
01. public class Questao36
02. {
03.   public static void main( String[] args )
04.   {
05.     int cont;
06.     for ( cont = 1; cont <= 10; cont++)
07.        {
08.           if ( cont == 5 )
09.                break;
10.           System.out.printf( "%d", cont );
11.         }
12.      System.out.printf( "Loop parou no número = %d\n", cont);
13.   }
14. }
```

Q15) [IBFC Prefeitura Davinópolis 2018] Quanto a linguagem Java, assinale a alternativa que apresenta um ambiente de desenvolvimento:

a) Elixir

b) Eiffel

c) Eclipse

d) Ectase

Q16) [FCC TRF3 2019] Para concatenar a string "TRF" com a variável reg que contém o valor inteiro 3, em Java e PHP, utilizam-se, respectivamente,

a) String("TRF") +reg e str("TRF").$reg

b) "TRF"+reg e "TRF".$reg

c) "TRF"+reg e "TRF"+$reg

d) "TRF",reg e Concat("TRF",$reg)

e) String.Concat("TRF",reg) e "TRF"+$reg

Q15) [IBFC Prefeitura Davinópolis 2018] Quanto a linguagem Java, assinale a alternativa que apresenta um ambiente de desenvolvimento:

a) Elixir

b) Eiffel

c) Eclipse

d) Ectase

Q16) [FCC TRF3 2019] Para concatenar a string "TRF" com a variável reg que contém o valor inteiro 3, em Java e PHP, utilizam-se, respectivamente,

a) String("TRF") +reg e str("TRF").$reg

b) "TRF"+reg e "TRF".$reg

c) "TRF"+reg e "TRF"+$reg

d) "TRF",reg e Concat("TRF",$reg)

e) String.Concat("TRF",reg) e "TRF"+$reg

Q17) [FCC TRF3 2019]
Considere o método abaixo, retirado de uma aplicação Java em condições ideais.

```
private static double[] entrada() {
  int cont;
  double n[];
  n = new double[3];
  for (cont = 0; cont <=3; cont++) {
    n[cont] = Double.parseDouble(JOptionPane.showInputDialog("Valor" + (cont + 1)));
  }
  return n;
}
```

a) cont<=3 no comando for para cont <=2.

b) n = new double[3] para n = new array[3].

c) cont<=3 no comando for para cont>=3.

d) cont++ para cont=cont+l.

e) "Valor" + (cont + 1) para "Valor" . (cont + 1).

Q17) [FCC TRF3 2019]
Considere o método abaixo, retirado de uma aplicação Java em condições ideais.

```
private static double[] entrada() {
  int cont;
  double n[];
  n = new double[3];
  for (cont = 0; cont <=3; cont++) {
    n[cont] = Double.parseDouble(JOptionPane.showInputDialog("Valor" + (cont + 1)));
  }
  return n;
}
```

a) cont<=3 no comando for para cont <=2.

b) n = new double[3] para n = new array[3].

c) cont<=3 no comando for para cont>=3.

d) cont++ para cont=cont+l.

e) "Valor" + (cont + 1) para "Valor" . (cont + 1).

Q18) [FCC TRF3 2019] Em uma aplicação Java, um vetor n foi criado por meio da instrução double n=new double [3] ; e alimentado com 3 valores reais. Para exibir o conteúdo da segunda posição (índice) deste vetor utiliza-se a instrução

a) JOptionPane.showMessageDialog(null, n[2]);

b) System.out.println(n[1]);

c) JOptionPane.ShowMessageDialog(n[2]);

d) System.Out.Println(null, n[2]);

e) JOptionPane.showWindows(0, n[2]);

Q18) [FCC TRF3 2019] Em uma aplicação Java, um vetor n foi criado por meio da instrução double n=new double [3] ; e alimentado com 3 valores reais. Para exibir o conteúdo da segunda posição (índice) deste vetor utiliza-se a instrução

a) JOptionPane.showMessageDialog(null, n[2]);

b) System.out.println(n[1]);

c) JOptionPane.ShowMessageDialog(n[2]);

d) System.Out.Println(null, n[2]);

e) JOptionPane.showWindows(0, n[2]);

Q19) [COVEST-COPSET UFPE 2019] Considere uma String s, que armazena o valor "ALO MUNDO”. Utilizando Java e Python, respectivamente, a alternativa com as instruções que exibiriam a substring “MU” seria:

a) Java: s.substring(4,6)

Python: s.substring(5,7)

b) Java: s[5,6]

Python: s[5:6]

c) Java: s.substring(4,6)

Python: s[-5:-3]

d) Java: s.substring(-4,-3)

Python: s[5:6]

e) Java: s.substring(4,6)

Python: s.substr(4,6)

Q19) [COVEST-COPSET UFPE 2019] Considere uma String s, que armazena o valor "ALO MUNDO”. Utilizando Java e Python, respectivamente, a alternativa com as instruções que exibiriam a substring “MU” seria:

a) Java: s.substring(4,6)

Python: s.substring(5,7)

b) Java: s[5,6]

Python: s[5:6]

c) Java: s.substring(4,6)

Python: s[-5:-3]

d) Java: s.substring(-4,-3)

Python: s[5:6]

e) Java: s.substring(4,6)

Python: s.substr(4,6)

Tanto no Java, quanto Python, o último item do slice é descartado.

ALO MUNDO
01234 5 6 7 8

Q20) [AOCP Prefeitura Juiz de Fora 2016] Muitas linguagens de programação definem os tipos de dados primitivos para ponto flutuante. Quais dos tipos a seguir podem ser considerados ponto flutuante em sua linguagem?

a) Java: double; delphi: real

b) Java: Int; delphi: real

c) Java: word; delphi: currency

d) Java: byte; delphi: currency

e) Java: float; delphi: word

Q21) [AOCP Prefeitura Juiz de Fora 2016] A seguinte linha de código foi escrita em java. O que ela faz?

String [ ][ ] s = new String[10][10];

a) Cria um array de strings com 10 posições.

b) Cria uma array bidimensional de String com tamanho 10 por 10.

c) Cria um List de String com 100 posições.

d) Cria um StringBuilder bidimensional com tamanho 10 por 10.

e) Cria dois arrays de String e cada um com 10 posições.

Q20) [AOCP Prefeitura Juiz de Fora 2016] Muitas linguagens de programação definem os tipos de dados primitivos para ponto flutuante. Quais dos tipos a seguir podem ser considerados ponto flutuante em sua linguagem?

a) Java: double; delphi: real

b) Java: Int; delphi: real

c) Java: word; delphi: currency

d) Java: byte; delphi: currency

e) Java: float; delphi: word

Q21) [AOCP Prefeitura Juiz de Fora 2016] A seguinte linha de código foi escrita em java. O que ela faz?

String [ ][ ] s = new String[10][10];

a) Cria um array de strings com 10 posições.

b) Cria uma array bidimensional de String com tamanho 10 por 10.

c) Cria um List de String com 100 posições.

d) Cria um StringBuilder bidimensional com tamanho 10 por 10.

e) Cria dois arrays de String e cada um com 10 posições.

Q22) [AOCP Prefeitura Juiz de Fora 2016] Na linguagem de programação delphi, existe o tipo de dados primitivos chamado smallint e, na linguagem java, existe o tipo de dado primitivo short. Esses dois tipos compreendem uma faixa de valores que pode assumir e essa faixa é a mesma nas duas linguagens. Qual é essa faixa de valores?

a) 0 a 255

b) -2.147.483.648 a 2.147.483.647

c) -32.768 a 32.767

d) -128 a 127

e) 0 a 65.535

Q22) [AOCP Prefeitura Juiz de Fora 2016] Na linguagem de programação delphi, existe o tipo de dados primitivos chamado smallint e, na linguagem java, existe o tipo de dado primitivo short. Esses dois tipos compreendem uma faixa de valores que pode assumir e essa faixa é a mesma nas duas linguagens. Qual é essa faixa de valores?

a) 0 a 255

b) -2.147.483.648 a 2.147.483.647

c) -32.768 a 32.767

d) -128 a 127

e) 0 a 65.535

Q23) [CESGRANRIO UNIRIO 2019] Que linha de programação Java deve ser dada para obter, via JDBC, o resultado da consulta guardada na variável S, da classe String, sabendo-se que a variável st é da classe Statement?

a) ResultSet rs = jdbc.executeQuery (S);

b) ResultSet rs = st.executeQuery(S);

c) ResultSet rs = st.runQuery(S);

d) st = S.executeQuery();

e) st = S.runQuery();

Q23) [CESGRANRIO UNIRIO 2019] Que linha de programação Java deve ser dada para obter, via JDBC, o resultado da consulta guardada na variável S, da classe String, sabendo-se que a variável st é da classe Statement?

a) ResultSet rs = jdbc.executeQuery (S);

b) ResultSet rs = st.executeQuery(S);

c) ResultSet rs = st.runQuery(S);

d) st = S.executeQuery();

e) st = S.runQuery();

Q24) [FEPESE CELESC 2019] Assinale a alternativa que apresenta um mecanismo de concorrência suportado explicitamente pela linguagem Java.

a) Virtuals

b) Concurrent

c) Multithreading

d) Parathreading

e) Paralels

Q25) [CS-UFG Prefeitura Goianira 2019] A linguagem Java utiliza algumas ferramentas com o objetivo de facilitar o desenvolvimento de aplicações, como o NetBeans, que é um exemplo de

a) ambiente de desenvolvimento integrado.

b) framework de aplicação de projetos web.

c) linguagem de programação alto nível.

d) banco de dados objeto-relacional.

Q24) [FEPESE CELESC 2019] Assinale a alternativa que apresenta um mecanismo de concorrência suportado explicitamente pela linguagem Java.

a) Virtuals

b) Concurrent

c) Multithreading

d) Parathreading

e) Paralels

Q25) [CS-UFG Prefeitura Goianira 2019] A linguagem Java utiliza algumas ferramentas com o objetivo de facilitar o desenvolvimento de aplicações, como o NetBeans, que é um exemplo de

a) ambiente de desenvolvimento integrado.

b) framework de aplicação de projetos web.

c) linguagem de programação alto nível.

d) banco de dados objeto-relacional.

Q26) [FEPESE CELESC 2019] Analise as afirmativas abaixo com relação aos modificadores de acesso (qualificadores) em java.

1. Os modificadores de acesso, também conhecidos como qualificadores são a forma de visibilidade das classes, dos métodos e atributos, ou seja, define quem poderá acessá-los, como por exemplo, se for colocado o qualificador public será visível em todo o projeto, todas as classes e pacotes.

2. O que declaramos como private será visível apenas para a classe em que foi declarado, ou seja, fica restrito à classe. Pode ser utilizado em atributos, métodos e construtores.

3. Quando declaramos com qualificador anonymous somente é possível acessar o método ou a variável se estiver no mesmo pacote, ou seja, se torna visível pela própria classe, por subclasses e pelas classes do mesmo pacote.

Assinale a alternativa que indica todas as afirmativas corretas.

a) É correta apenas a afirmativa 1.

b) É correta apenas a afirmativa 3.

c) São corretas apenas as afirmativas 1 e 2.

d) São corretas apenas as afirmativas 1 e 3.

e) São corretas apenas as afirmativas 2 e 3.

Q26) [FEPESE CELESC 2019] Analise as afirmativas abaixo com relação aos modificadores de acesso (qualificadores) em java.

1. Os modificadores de acesso, também conhecidos como qualificadores são a forma de visibilidade das classes, dos métodos e atributos, ou seja, define quem poderá acessá-los, como por exemplo, se for colocado o qualificador public será visível em todo o projeto, todas as classes e pacotes.

2. O que declaramos como private será visível apenas para a classe em que foi declarado, ou seja, fica restrito à classe. Pode ser utilizado em atributos, métodos e construtores.

3. Quando declaramos com qualificador anonymous somente é possível acessar o método ou a variável se estiver no mesmo pacote, ou seja, se torna visível pela própria classe, por subclasses e pelas classes do mesmo pacote.

Assinale a alternativa que indica todas as afirmativas corretas.

a) É correta apenas a afirmativa 1.

b) É correta apenas a afirmativa 3.

c) São corretas apenas as afirmativas 1 e 2.

d) São corretas apenas as afirmativas 1 e 3.

e) São corretas apenas as afirmativas 2 e 3.

Q27) [CS-UFG Prefeitura Goianira 2019] Considere o trecho de código em Java a seguir.

Quais são os seis algarismos resultantes deste trecho de código?

a) 5; 4; 3; 2; 1; 0

b) 0; 1; 2; 3; 4; 5.

c) 0; 1; 1; 2; 3; 5.

d) 5; 3; 2; 1; 1; 0.

```
public class Sequencia {
static long seq(int n) {
 return (n < 2) ? n : seq(n - 1) + seq(n - 2);
}
public static void main(String[] args) {
for (int i = 0; i < 6; i++) {

System.out.print("(" + i + "):" + Sequencia.seq(i) +
"\t");

}

}

}
```

Q27) [CS-UFG Prefeitura Goianira 2019] Considere o trecho de código em Java a seguir.

Quais são os seis algarismos resultantes deste trecho de código?

a) 5; 4; 3; 2; 1; 0

b) 0; 1; 2; 3; 4; 5.

c) 0; 1; 1; 2; 3; 5.

d) 5; 3; 2; 1; 1; 0.

```
public class Sequencia {
static long seq(int n) {
 return (n < 2) ? n : seq(n - 1) + seq(n - 2);
}
public static void main(String[] args) {
for (int i = 0; i < 6; i++) {

System.out.print("(" + i + "):" +
Sequencia.seq(i) + "\t");

}

}

}
```

I0 (valor 0) - Condição 1 - Resultado = 0
I1 (valor 1) - Condição 1 - Resultado = 1
I2 (valor 2)- Condição 2 - Resultado seq(2 - 1) + seq(2 - 2) = 1
I3 (valor 3)- Condição 2 - Resultado seq(3 - 1) + seq(3 - 2) = 2
I4 (valor 4)- Condição 2 - Resultado seq(4 - 1) + seq(4 - 2) = 3
I5 (valor 5)- Condição 2 - Resultado seq(5 - 1) + seq(5 - 2) = 5

Q28) [IF-MT IF-MT 2019] Considere o trecho de código escrito na linguagem Java, apresentado a seguir:

```
1 - public class MinhaClasse
2 - {
3 -     public static void main(String[] args)
4 -     {
5-          int numero1 = 13;
6 -         int numero2 = 31;
7 -         if (numero1 = numero2)
8 -             System.out.print("Os numeros sao iguais");
9 -         else
10 -          System.out.print("Os numeros sao diferentes");
11 -    }
12 - }
```

É CORRETO afirmar que:

a) A sua execução apresentará a mensagem: Os números sao iguais.

b) A sua execução apresentará a mensagem: Os números sao diferentes.

c) A sua compilação apresentará uma mensagem de erro na linha 1.

d) A sua compilação apresentará uma mensagem de erro na linha 3.

e) A sua compilação apresentará uma mensagem de erro na linha 7.

Q28) [IF-MT IF-MT 2019] Considere o trecho de código escrito na linguagem Java, apresentado a seguir:

```
1 - public class MinhaClasse
2 - {
3 -     public static void main(String[] args)
4 -     {
5-          int numero1 = 13;
6 -         int numero2 = 31;
7 -         if (numero1 = numero2)
8 -             System.out.print("Os numeros sao iguais");
9 -         else
10 -          System.out.print("Os numeros sao diferentes");
11 -    }
12 - }
```

É CORRETO afirmar que:

a) A sua execução apresentará a mensagem: Os números sao iguais.

b) A sua execução apresentará a mensagem: Os números sao diferentes.

c) A sua compilação apresentará uma mensagem de erro na linha 1.

d) A sua compilação apresentará uma mensagem de erro na linha 3.

e) A sua compilação apresentará uma mensagem de erro na linha 7.

Q29) [VUNESPE Câmara de Monte Alto 2019] Observe o trecho de código Java a seguir:

boolean x, y, z; // ... if (!x && y && !z) System.out.println("SIM"); else System.out.println("NAO");

A mensagem impressa na tela será “SIM” somente quando o valor das variáveis x, y e z forem, respectivamente:

a) false, true e false.

b) false, false e false.

c) true, true e true.

d) true, false e true.

e) true, false, true.

Q29) [VUNESPE Câmara de Monte Alto 2019] Observe o trecho de código Java a seguir:

boolean x, y, z; // ... if (!x && y && !z) System.out.println("SIM"); else System.out.println("NAO");

A mensagem impressa na tela será “SIM” somente quando o valor das variáveis x, y e z forem, respectivamente:

a) false, true e false.

b) false, false e false.

c) true, true e true.

d) true, false e true.

e) true, false, true.

Q29) [VUNESPE Prefeitura de Itapevi-SP 2019] No Java, o modificador de nível de acesso que torna o método acessível apenas dentro do escopo da classe na qual foi declarado, ou em subclasses, é:

a) abstract.

b) private.

c) protected.

d) public.

e) virtual.

Q30) [FCC TJ-MA 2019] Em um programa na linguagem Java, um Técnico declarou uma variável e atribuiu a ela um valor inteiro por meio do comando short a = 32768;. Ao tentar executar a aplicação, este comando gerou uma exceção porque variáveis do tipo short só podem conter valores na faixa de:

a) -128 a 127.

b) 0 a 256.

c) -32768 a 32767.

d) -2048 a 2047.

e) 0 a 32667.

Q29) [VUNESPE Prefeitura de Itapevi-SP 2019] No Java, o modificador de nível de acesso que torna o método acessível apenas dentro do escopo da classe na qual foi declarado, ou em subclasses, é:

a) abstract.

b) private.

c) protected.

d) public.

e) virtual.

Q30) [FCC TJ-MA 2019] Em um programa na linguagem Java, um Técnico declarou uma variável e atribuiu a ela um valor inteiro por meio do comando short a = 32768;. Ao tentar executar a aplicação, este comando gerou uma exceção porque variáveis do tipo short só podem conter valores na faixa de:

a) -128 a 127.

b) 0 a 256.

c) -32768 a 32767.

d) -2048 a 2047.

e) 0 a 32667.

Q31) [FCC SABESP 2019] Considere o trecho de código em Java.

```
String s = "Situação dos mananciais. Volume operacional.";
System.out.println("String inicial: "+ s);
 I
.....
System.out.println("String modificado: "+ s);
```

Deseja-se substituir a expressão Volume operacional por Pluviometria na variável s. Para isso, deve-se preencher o trecho I com

a) s = s.trim("Volume operacional","Pluviometria") ;

b) String s = replace ('Volume operacional', 'Pluviometria') ;

c) s = s.replace ("Volume operacional","Pluviometria") ;

d) s = s.concat('Volume operacional', 'Pluviometria') ;

e) s = string.replace("Pluviometria","Volume operacional") ;

Q31) [FCC SABESP 2019] Considere o trecho de código em Java.

```
String s = "Situação dos mananciais. Volume operacional.";
System.out.println("String inicial: "+ s);
 I
.....
System.out.println("String modificado: "+ s);
```

Deseja-se substituir a expressão Volume operacional por Pluviometria na variável s. Para isso, deve-se preencher o trecho I com

a) s = s.trim("Volume operacional","Pluviometria") ;

b) String s = replace ('Volume operacional', 'Pluviometria') ;

c) s = s.replace ("Volume operacional","Pluviometria") ;

d) s = s.concat('Volume operacional', 'Pluviometria') ;

e) s = string.replace("Pluviometria","Volume operacional") ;

Q32) [FCC SANASA 2019] Um Analista de TI está programando em Java e deseja relacionar a classe AcompanhaManancial de tal maneira que esta herde tudo que a classe Manancial tem, criando uma relação de superclasse e subclasse. Isso é conseguido em Java usando, inicialmente:

a) @Override public class Manancial { public class AcompanhaManancial {

b) public class AcompanhaManancial extends Manancial {

c) public class Manancial { @Override public class AcompanhaManancial {

d) protected interface AcompanhaManancial extends Manancial {

e) protected class Manancial includes AcompanhaManancial {

Q32) [FCC SANASA 2019] Um Analista de TI está programando em Java e deseja relacionar a classe AcompanhaManancial de tal maneira que esta herde tudo que a classe Manancial tem, criando uma relação de superclasse e subclasse. Isso é conseguido em Java usando, inicialmente:

a) @Override public class Manancial { public class AcompanhaManancial {

b) public class AcompanhaManancial extends Manancial {

c) public class Manancial { @Override public class AcompanhaManancial {

d) protected interface AcompanhaManancial extends Manancial {

e) protected class Manancial includes AcompanhaManancial {

Q33) [VUNESP UFABC 2019] Na linguagem Java, o método “add(E e)” da classe ArrayList, adiciona um elemento

a) em uma nova lista.

b) em uma posição especificada da lista.

c) em uma posição aleatória da lista.

d) ao final da lista

e) no início da lista.

Q34) [IADES BRB 2019] Em relação à linguagem de programação Java, qual o método estático que apenas exibe uma caixa de diálogo contendo uma mensagem?

a) Graphics.showMessage

b) JFrame.showInputDialog

c) JFrame.showMessage

d) JOptionPane.showInputDialog

e) JOptionPane.showMessageDialog

Q33) [VUNESP UFABC 2019] Na linguagem Java, o método “add(E e)” da classe ArrayList, adiciona um elemento

a) em uma nova lista.

b) em uma posição especificada da lista.

c) em uma posição aleatória da lista.

d) ao final da lista

e) no início da lista.

Q34) [IADES BRB 2019] Em relação à linguagem de programação Java, qual o método estático que apenas exibe uma caixa de diálogo contendo uma mensagem?

a) Graphics.showMessage

b) JFrame.showInputDialog

c) JFrame.showMessage

d) JOptionPane.showInputDialog

e) JOptionPane.showMessageDialog

Q35) [PR-4 UFRJ UFRJ 2018] Considere o programa Java a seguir, executado em um ambiente com Java 8 (JDK 1.8) e implementado no IDE Eclipse versão 4.6.3.

```
1.public class Perfil {
2.      public static void main(String[] args) {
3.              int i = 29 % 3;
4.              switch(i) {
5.                      case 1: System.out.print("UFRJ "); break;
6.                      case 2: System.out.print("Aluno ");
7.                      case 3: { System.out.print("Professor "); break; }
8.                      case 4: System.out.print("Técnico ");
9.                      default: System.out.print("Indefinido ");
10.             }
11.     }
12.}
```

Na execução do programa apresentado, a saída obtida será:

a) Aluno Professor Técnico.

b) Aluno Professor Técnico Indefinido.

c) Aluno Professor Indefinido.

d) Aluno Professor UFRJ.

e) Aluno Professor.

Q35) [PR-4 UFRJ UFRJ 2018] Considere o programa Java a seguir, executado em um ambiente com Java 8 (JDK 1.8) e implementado no IDE Eclipse versão 4.6.3.

```
1.public class Perfil {
2.      public static void main(String[] args) {
3.              int i = 29 % 3;
4.              switch(i) {
5.                      case 1: System.out.print("UFRJ "); break;
6.                      case 2: System.out.print("Aluno ");
7.                      case 3: { System.out.print("Professor "); break; }
8.                      case 4: System.out.print("Técnico ");
9.                      default: System.out.print("Indefinido ");
10.             }
11.     }
12.}
```

Na execução do programa apresentado, a saída obtida será:

a) Aluno Professor Técnico.

b) Aluno Professor Técnico Indefinido.

c) Aluno Professor Indefinido.

d) Aluno Professor UFRJ.

e) Aluno Professor.

Q36) [IDECAN IFPB 2019] Dadas as seguintes classes, todas no mesmo pacote:

Qual o resultado da impressão ao executamos a classe Homem?

a) Mamífero andando Mamífero ouvindo Mamífero vendo.

b) Primata andando Mamífero ouvindo Homem vendo.

c) Uma exception será lançada: MethodNotFoundException.

d) Mamífero andando Mamífero ouvindo Homem vendo.

e) Primata andando Mamífero ouvindo Mamífero vendo

```
public class Mamifero {
    protected void andar(){
        System.out.print("Mamífero andando ");
        ouvir();
    }
    protected void ver(){
        System.out.print("Mamífero vendo ");

    }
    protected void ouvir(){
        System.out.print("Mamífero ouvindo ");
        ver();
    }
}
public class Primata extends Mamifero{
    protected void andar(){
        System.out.print("Primata andando ");
        ouvir();
    }
}
public class Homem extends Primata{
    protected void ver(){
        System.out.print("Homem vendo ");
    }
    public static void main(String args[]){
        Mamifero m = new Homem();
        m.andar();
    }
}
```

Q36) [IDECAN IFPB 2019] Dadas as seguintes classes, todas no mesmo pacote:

Qual o resultado da impressão ao executamos a classe Homem?

a) Mamífero andando Mamífero ouvindo Mamífero vendo.

b) Primata andando Mamífero ouvindo Homem vendo.

c) Uma exception será lançada: MethodNotFoundException.

d) Mamífero andando Mamífero ouvindo Homem vendo.

e) Primata andando Mamífero ouvindo Mamífero vendo

```
public class Mamifero {
    protected void andar(){
        System.out.print("Mamífero andando ");
        ouvir();
    }
    protected void ver(){
        System.out.print("Mamífero vendo ");

    }
    protected void ouvir(){
        System.out.print("Mamífero ouvindo ");
        ver();
    }
}
public class Primata extends Mamifero{
    protected void andar(){
        System.out.print("Primata andando ");
        ouvir();
    }
}
public class Homem extends Primata{
    protected void ver(){
        System.out.print("Homem vendo ");
    }
    public static void main(String args[]){
        Mamifero m = new Homem();
        m.andar();
    }
}
```

Q37) [COMPERVE UFRN 2019] A linguagem de programação Java possibilita a criação de classes e objetos usando os conceitos de orientação a objetos para o desenvolvimento de programas. Diante disso, analise a classe Java, denominada Y, abaixo.

```
package concurso;
public class Y {

    int id;
   String descricao;

   public Y(int id, String descricao) {
        this.id = id;
        this.descricao = descricao;
   }

   public static void main(String[] args) {
        Y y = new Y();
        y.id = 1;
        y.descricao = "concurso";
        System.out.println(y.descricao);
   }
}
```

Com base nessas informações, conclui-se:

a) quando compilada e executada, o console apresentará uma exceção.

b) o console apresentará a palavra “concurso”.

c) quando compilada e executada, o console não apresentará nada..

d) a classe apresentará um erro de compilação.

Q37) [COMPERVE UFRN 2019] A linguagem de programação Java possibilita a criação de classes e objetos usando os conceitos de orientação a objetos para o desenvolvimento de programas. Diante disso, analise a classe Java, denominada Y, abaixo.

```
package concurso;

public class Y {


    int id;
   String descricao;


   public Y(int id, String descricao) {
            this.id = id;
            this.descricao = descricao;
   }


   public static void main(String[] args) {
            Y y = new Y();
            y.id = 1;
            y.descricao = "concurso";
            System.out.println(y.descricao);
   }
}
```

Com base nessas informações, conclui-se:

a) quando compilada e executada, o console apresentará uma exceção.

b) o console apresentará a palavra “concurso”.

c) quando compilada e executada, o console não apresentará nada..

d) a classe apresentará um erro de compilação.

Q38) [COMPERVE UFRN 2019] A linguagem de programação Java permite a criação de programas que façam uso de recursividade. Isto posto, analise a classe Java, denominada X, abaixo.

```
package concurso;
public class X {
        public int x(int x) {
                if(x == 0)
                        return 1;
                return x* x(x-1);
        }

        public static void main(String[] args) {
                X x = new X();
                System.out.println(x.x(5));
        }
}
```

Com base nessas informações, conclui-se:

a) quando compilada e executada, o console apresentará o valor 120.

b) quando compilada e executada, a classe entrará em loop infinito.

c) quando compilada e executada, o console apresentará o valor 720.

d) quando compilada e executada, o console apresentará uma exceção.

Q38) [COMPERVE UFRN 2019] A linguagem de programação Java permite a criação de programas que façam uso de recursividade. Isto posto, analise a classe Java, denominada X, abaixo.

```
package concurso;
public class X {
        public int x(int x) {
                if(x == 0)
                        return 1;
                return x* x(x-1);
        }

        public static void main(String[] args) {
                X x = new X();
                System.out.println(x.x(5));
        }
}
```

Com base nessas informações, conclui-se:

a) quando compilada e executada, o console apresentará o valor 120.

b) quando compilada e executada, a classe entrará em loop infinito.

c) quando compilada e executada, o console apresentará o valor 720.

d) quando compilada e executada, o console apresentará uma exceção.

return 1 * x(0) é igual a 1 * 1 = 1
return 2 * x(1) é igual a 2 * 1 = 2
return 3 * x(2) é igual a 3 * 2 = 6
return 4 * x(3) é igual a 4 * 6 = 24
return 5 * x(4) é igual a 5 * 24 = 120

Q39) [COMPERVE UFRN 2019] Em aplicações em que se utilizam os conceitos de orientação a objetos, precisa-se constantemente manipular muitas informações e muitos objetos ao mesmo tempo. Então, utilizam-se estruturas que permitem armazená-los e recuperá-los sempre que se desejar. A linguagem Java oferece várias dessas estruturas de dados em um conjunto de classes chamadas de coleções. Nesse contexto, uma API da linguagem Java que tem como característica implementar coleções ordenadas (sequências) é

a) java.util.HashSet<E>

b) java.util.List<E>

c) java.util.TreeSet<E>

d) java.util.HashMap<K,V>

Q40) [INAZ CORE-SP 2019] A linguagem de programação Java, apresenta uma série de “operadores” que são utilizados, principalmente, na etapa de processamento para a construção da lógica, possibilitando realizar ações específicas sobre os dados. Um determinado operador é utilizado para definir o valor inicial ou sobrescrever o valor de uma variável. Em seu uso, o operando à esquerda representa a variável para a qual desejamos indicar o valor informado à direita. Estamos falando do operador:

a) Aritmético.

b) De atribuição.

c) Relacional.

d) Lógico.

e) De variação.

Q39) [COMPERVE UFRN 2019] Em aplicações em que se utilizam os conceitos de orientação a objetos, precisa-se constantemente manipular muitas informações e muitos objetos ao mesmo tempo. Então, utilizam-se estruturas que permitem armazená-los e recuperá-los sempre que se desejar. A linguagem Java oferece várias dessas estruturas de dados em um conjunto de classes chamadas de coleções. Nesse contexto, uma API da linguagem Java que tem como característica implementar coleções ordenadas (sequências) é

a) java.util.HashSet<E>

b) java.util.List<E>

c) java.util.TreeSet<E>

d) java.util.HashMap<K,V>

Entre List, Set e Map, List é a única que preserva a posição (índice) dos elementos e a ordem em que foram inseridos.

Q40) [INAZ CORE-SP 2019] A linguagem de programação Java, apresenta uma série de "operadores" que são utilizados, principalmente, na etapa de processamento para a construção da lógica, possibilitando realizar ações específicas sobre os dados. Um determinado operador é utilizado para definir o valor inicial ou sobrescrever o valor de uma variável. Em seu uso, o operando à esquerda representa a variável para a qual desejamos indicar o valor informado à direita. Estamos falando do operador:

a) Aritmético.

b) De atribuição.

c) Relacional.

d) Lógico.

e) De variação.

Operadores de atribuição compostos: +=, -=, *=, %= ...

Operadores de atribuição binário: =

Q41) [COMPERVE UFRN 2019]Considerando a utilização da linguagem Java, suponha que exista uma classe denominada Bicicleta e que, dentro dela, exista, entre outros, o trecho de código mostrado no Quadro 2 abaixo:

```
public Bicicleta(int inicioCadencia, int inicioVel, int inicioMarcha) {
    marcha = inicioMarcha;
    cadencia = inicioCadencia;
    vel = inicioVel;
}
```

Quadro 2 – Trecho de código Java

Em relação ao trecho mostrado e às características da linguagem Java, analise as assertivas a seguir:

I. O trecho de código mostrado é um exemplo de construtor.

II. Para criar um novo objeto Bicicleta, utiliza-se o operador new, como, por exemplo: Bicicleta minhaBike = new Bicicleta(25, 0, 7);.

III. O trecho mostrado não é um método.

IV. Para evitar conflitos, deve haver apenas um construtor por classe.

Quais estão corretas?

a) Apenas I e II.

b) Apenas II e III.

c) Apenas III e IV.

d) Apenas II, III e IV.

e) I, II, III e IV.

Q41) [COMPERVE UFRN 2019]Considerando a utilização da linguagem Java, suponha que exista uma classe denominada Bicicleta e que, dentro dela, exista, entre outros, o trecho de código mostrado no Quadro 2 abaixo:

```
public Bicicleta(int inicioCadencia, int inicioVel, int inicioMarcha) {
    marcha = inicioMarcha;
    cadencia = inicioCadencia;
    vel = inicioVel;
}
```

Quadro 2 – Trecho de código Java

Em relação ao trecho mostrado e às características da linguagem Java, analise as assertivas a seguir:

I. O trecho de código mostrado é um exemplo de construtor.

II. Para criar um novo objeto Bicicleta, utiliza-se o operador new, como, por exemplo: Bicicleta minhaBike = new Bicicleta(25, 0, 7);.

III. O trecho mostrado não é um método.

IV. Para evitar conflitos, deve haver apenas um construtor por classe.

Quais estão corretas?

a) Apenas I e II.

b) Apenas II e III.

c) Apenas III e IV.

d) Apenas II, III e IV.

e) I, II, III e IV.

Q42) [INAZ CORE-SP 2019] “O desenvolvimento de software é extremamente amplo. Nesse mercado, existem diversas linguagens de programação, que seguem diferentes paradigmas. Um desses paradigmas é a Orientação a Objetos, que atualmente é o mais difundido entre todos. Isso acontece porque se trata de um padrão que tem evoluído muito, principalmente em questões voltadas para segurança e reaproveitamento de código, o que é muito importante no desenvolvimento de qualquer aplicação moderna.”

Considere o programa abaixo escrito na linguagem Java:

```
Public class veículo {}

Public class carro extends veículo {}

Public class avião extends veículo {}
```

Qual a afirmativa correta?

a) A classe veículo é subclasse da classe avião.

b) A classe avião é subclasse da classe carro.

c) A classe veículo é superclasse das classes carro e avião.

d) As classes carro e avião são superclasses da classe veículo.

e) As classes veículo e carro são subclasses da classe maquinas.

Q42) [INAZ CORE-SP 2019] “O desenvolvimento de software é extremamente amplo. Nesse mercado, existem diversas linguagens de programação, que seguem diferentes paradigmas. Um desses paradigmas é a Orientação a Objetos, que atualmente é o mais difundido entre todos. Isso acontece porque se trata de um padrão que tem evoluído muito, principalmente em questões voltadas para segurança e reaproveitamento de código, o que é muito importante no desenvolvimento de qualquer aplicação moderna.”

Considere o programa abaixo escrito na linguagem Java:

Public class veículo {}

Public class carro extends veículo {}

Public class avião extends veículo {}

Qual a afirmativa correta?

a) A classe veículo é subclasse da classe avião.

b) A classe avião é subclasse da classe carro.

c) A classe veículo é superclasse das classes carro e avião.

d) As classes carro e avião são superclasses da classe veículo.

e) As classes veículo e carro são subclasses da classe maquinas.

Q43) [INAZ CORE-SP 2019] “A Programação Orientada a Objetos (POO), foi criada para tentar aproximar o mundo real e o mundo virtual: a ideia fundamental é tentar simular o mundo real dentro do computador. Para isso, nada mais natural do que utilizar objetos, afinal, nosso mundo é composto de objetos”.

Qual o nome de todas as classes utilizadas pela classe Discente?

a) Vector, String, NotaDisciplina, Object.

b) Vector, NotaDisciplina, Object, int.

c) String, int, Vector, Object.

d) int, String, Vector.

e) NotaDisciplina, Vector, int, Object, String.

```
public class Discente{
        private static final int NUM_ANOS = 5;
        private String nome;
        private int nome;
        private Vector[] notas = new Vector[NUM_ANOS];
        public Discente (String nome, int num){}
        public Discente (){}
        protected void adicionaNota (int nota, String disciplina,
int ano) {
                NotaDisciplina nd = new
NotaDisciplina(nota,disciplina);
                notas[ano].addElement(nd);
        }
        protected void igualNota (Object outroDiscente, String
disciplina){
                return false;
        }
}
```

Q43) [INAZ CORE-SP 2019] "A Programação Orientada a Objetos (POO), foi criada para tentar aproximar o mundo real e o mundo virtual: a ideia fundamental é tentar simular o mundo real dentro do computador. Para isso, nada mais natural do que utilizar objetos, afinal, nosso mundo é composto de objetos".

Qual o nome de todas as classes utilizadas pela classe Discente?

a) Vector, String, NotaDisciplina, Object.

b) Vector, NotaDisciplina, Object, int.

c) String, int, Vector, Object.

d) int, String, Vector.

e) NotaDisciplina, Vector, int, Object, String.

```
public class Discente{
        private static final int NUM_ANOS = 5;
        private String nome;
        private int nome;
        private Vector[] notas = new Vector[NUM_ANOS];
        public Discente (String nome, int num){}
        public Discente (){}
        protected void adicionaNota (int nota, String disciplina,
int ano) {
                NotaDisciplina nd = new
NotaDisciplina(nota,disciplina);
                notas[ano].addElement(nd);
        }
        protected void igualNota (Object outroDiscente, String
disciplina){
                return false;
        }
}
```

Q44) [INAZ CORE-SP 2019] “Uma linguagem de programação é um método padronizado para comunicar instruções para um computador. É um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador.”

Qual trecho de código em Java está declarando uma classe corretamente?

a) public class clienteVip extends cliente, pessoa.

b) public class clienteVip extends cliente implements clienteVip, pessoa.

c) public class clienteVip extends cliente implements funcionário.

d) public class cliente extends pessoaFisica, pessoa.

e) public abstract class cliente extends pessoaFisica implements <pessoa>.

Q44) [INAZ CORE-SP 2019] “Uma linguagem de programação é um método padronizado para comunicar instruções para um computador. É um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador.”

Qual trecho de código em Java está declarando uma classe corretamente?

a) public class clienteVip extends cliente, pessoa.

b) public class clienteVip extends cliente implements clienteVip, pessoa.

c) public class clienteVip extends cliente implements funcionário.

d) public class cliente extends pessoaFisica, pessoa.

e) public abstract class cliente extends pessoaFisica implements <pessoa>.

Q45) [CCV-UFC UFC 2019] Utilizando Java 8, qual o resultado do código abaixo?

```
public class Test
{
   public static void main(String[] args)
   {
      int value = 554;
      String var = (String)value;  //linha 1
      String temp = "123";
      int data = (int)temp; //linha 2
      System.out.println(data + var);
   }
}
```

a) Erro de compilação devido as linhas 1 e 2

b) Erro de compilação devido à linha 1

c) Erro de compilação devido à linha 2

d) ”554123”

e) 677

Q45) [CCV-UFC UFC 2019] Utilizando Java 8. qual o resultado do código abaixo?

```
public class Test
{
   public static void main(String[] args)
   {
      int value = 554;
      String var = (String)value;  //linha 1
      String temp = "123";
      int data = (int)temp; //linha 2
      System.out.println(data + var);
   }
}
```

a) Erro de compilação devido as linhas 1 e 2

b) Erro de compilação devido à linha 1

c) Erro de compilação devido à linha 2

d) ”554123”

e) 677

Por ser uma linguagem fortemente tipada, não é permitido concatenar variáveis de tipos de dado diferente. O correto seria utilizar esses métodos:

String var = String.valueOf(value); // Linha 1

int data = Integer.parseInt(temp); // Linha 2

Q46) [CCV-UFC UFC 2019] Utilizando Java 8, sobre a classe Cliente abaixo, o que podemos afirmar corretamente?

```
public abstract class Cliente {
    private String nome;

    public Cliente(String nome) {
        this.nome = nome;
    }

    public String getNome() {
        return nome;
    }

    public abstract void compra();
}
```

a) A classe Cliente não pode ser estendida.

b) Subclasses de Cliente devem implementar o método compra().

c) Subclasses concretas de Cliente devem utilizar um construtor padrão.

d) Subclasses concretas de Cliente devem implementar o método compra().

e) Subclasses de Cliente não podem sobrescrever o método getNome().

Q46) [CCV-UFC UFC 2019] Utilizando Java 8, sobre a classe Cliente abaixo, o que podemos afirmar corretamente?

```
public abstract class Cliente {
    private String nome;

    public Cliente(String nome) {
        this.nome = nome;
    }

    public String getNome() {
        return nome;
    }

    public abstract void compra();
}
```

a) A classe Cliente não pode ser estendida.

b) Subclasses de Cliente devem implementar o método compra().

c) Subclasses concretas de Cliente devem utilizar um construtor padrão.

d) Subclasses concretas de Cliente devem implementar o método compra().

e) Subclasses de Cliente não podem sobrescrever o método getNome().

Q47) [CCV-UFC UFC 2019] Em Java 8, qual modificador de acesso torna um membro disponível somente para classes dentro do mesmo pacote ou subclasses?

a) public

b) default

c) private

d) protected

e) package-private

Q48) [CCV-UFC UFC 2019] Sobre a linguagem de programação Java, assinale a alternativa correta.

a) Java é derivada da linguagem JavaScript.

b) Podemos utilizar somente o paradigma de programação Orientado a objetos.

c) Java suporta herança múltipla em classes por meio da palavra chave extends.

d) Java Swing foi criada para substituir JavaFx na criação de aplicativos Desktop.

e) A partir da versão Java 8 é possível declarar métodos concretos dentro de uma interface.

Q47) [CCV-UFC UFC 2019] Em Java 8, qual modificador de acesso torna um membro disponível somente para classes dentro do mesmo pacote ou subclasses?

a) public

b) default

c) private

d) protected

e) package-private

Q48) [CCV-UFC UFC 2019] Sobre a linguagem de programação Java, assinale a alternativa correta.

a) Java é derivada da linguagem JavaScript.

b) Podemos utilizar somente o paradigma de programação Orientado a objetos.

c) Java suporta herança múltipla em classes por meio da palavra chave extends.

d) Java Swing foi criada para substituir JavaFx na criação de aplicativos Desktop.

e) A partir da versão Java 8 é possível declarar métodos concretos dentro de uma interface.

Q49) [CCV-UFC UFC 2019] Utilizando Java 8, qual o resultado do código abaixo?

a) Programando

b) Escrevendo...

c) Escrevendo um livro

d) O código não compila.

e) Erro em tempo de execução

```
abstract class Escritor {
    public static void write() {
        System.out.println("Escrevendo...");
    }
}

class Autor extends Escritor {
    public static void write() {
        System.out.println("Escrevendo um livro");
    }
}

public class Programmer extends Escritor {
    public static void write() {
        System.out.println("Programando");
    }

public static void main(String[] args) {
    Escritor e = new Programmer();
    e.write();
    }
}
```

Q49) [CCV-UFC UFC 2019] Utilizando Java 8, qual o resultado do código abaixo?

a) Programando

b) Escrevendo...

c) Escrevendo um livro

d) O código não compila.

e) Erro em tempo de execução

Método estático não pode ser sobescrito!!

```
abstract class Escritor {
    public static void write() {
        System.out.println("Escrevendo...");
    }
}

class Autor extends Escritor {
    public static void write() {
        System.out.println("Escrevendo um livro");
    }
}

public class Programmer extends Escritor {
    public static void write() {
        System.out.println("Programando");
    }

public static void main(String[] args) {
    Escritor e = new Programmer();
    e.write();
    }
}
```

Q50) [FUNDEP Prefeitura de Lagoa Santa 2019] Analise as linhas de código em Java a seguir.

Em relação às linhas mostradas, assinale a que apresenta erro em Java.

a) Linha 2

b) Linha 3

c) Linha 4

d) Linha 6

```
1.  public class Produto {
2.    private string descricao;
3.    private int valor;
4.      public Produto() {
5.    }
6.      public void apresentarProduto() {
7.    }
8.  }
```

Q50) [FUNDEP Prefeitura de Lagoa Santa 2019] Analise as linhas de código em Java a seguir.

Em relação às linhas mostradas, assinale a que apresenta erro em Java.

```
1.  public class Produto {
2.    private string descricao;
3.    private int valor;
4.      public Produto() {
5.    }
6.      public void apresentarProduto() {
7.    }
8.  }
```

a) Linha 2

b) Linha 3

c) Linha 4

d) Linha 6

Q51) [UFMA UFMA 2019] Duas características importantes e relacionadas entre si, presentes em Java por ser uma linguagem orientada a objetos, são a herança e o polimorfismo. Considere as afirmativas I e II a seguir e depois marque a alternativa correta.

I. Herança múltipla é um recurso existente em Java para permitir que uma classe possa herdar atributos e métodos de mais de uma classe.

II. Polimorfismo em Java é a capacidade de duas ou mais classes derivadas de uma mesma superclasse possuírem a mesma assinatura de um método, porém com comportamento diferente.

a) Apenas a afirmativa II está correta.

b) Ambas as afirmativas I e II estão corretas.

c) Ambas as afirmativas I e II estão erradas.

d) Apenas a afirmativa I está correta.

e) A correção ou não das afirmativas I e II depende de qual versão de Java se está levando em consideração.

Q51) [UFMA UFMA 2019] Duas características importantes e relacionadas entre si, presentes em Java por ser uma linguagem orientada a objetos, são a herança e o polimorfismo. Considere as afirmativas I e II a seguir e depois marque a alternativa correta.

I. Herança múltipla é um recurso existente em Java para permitir que uma classe possa herdar atributos e métodos de mais de uma classe.

II. Polimorfismo em Java é a capacidade de duas ou mais classes derivadas de uma mesma superclasse possuírem a mesma assinatura de um método, porém com comportamento diferente.

a) Apenas a afirmativa II está correta.

b) Ambas as afirmativas I e II estão corretas.

c) Ambas as afirmativas I e II estão erradas.

d) Apenas a afirmativa I está correta.

e) A correção ou não das afirmativas I e II depende de qual versão de Java se está levando em consideração.

Q52) [FGV DPE-RJ 2019] Considere as seguintes afirmativas sobre class constructor na linguagem Java.

I. Deve receber o mesmo nome da classe a ele associada.

II. Não deve ser especificado um tipo de retorno na sua declaração.

III. É útil para a definição de valores iniciais para os atributos da classe.

IV. É sempre declarado como public.

É correto somente o que se afirma em:

a) I e II;

b) II e III;

c) III e IV;

d) I, II e III;

e) I, III e IV.

Q52) [FGV DPE-RJ 2019] Considere as seguintes afirmativas sobre class constructor na linguagem Java.

I. Deve receber o mesmo nome da classe a ele associada.

II. Não deve ser especificado um tipo de retorno na sua declaração.

III. É útil para a definição de valores iniciais para os atributos da classe.

IV. É sempre declarado como public.

É correto somente o que se afirma em:

a) I e II;

b) II e III;

c) III e IV;

d) I, II e III;

e) I, III e IV.

Q53) [VUNESP Câmara de Sertãozinho - SP 2019] Na linguagem Java, a sintaxe correta de um bloco de controle de exceção é:

a) try { // código a ser executado } catch (TipoExcecao nomeExcecao) { // tratamento da exceção }

b) try { // código a ser executado } catch (nomeExcecao: TipoExcecao) { // tratamento da exceção }

c) try { // código a ser executado } except { // tratamento da exceção }

d) if (TipoExcecao) { // código a ser executado } else { // tratamento da exceção }

e) switch (nomeExcecao){ default: // código a ser executado break; case TipoExcecao: // tratamento da exceção break; }

Q54) [IFPA IFPA 2019] Ao analisarmos uma classe Java, nos deparamos com um método que implementa diversas funcionalidades, tornando-se um método com muitas linhas de código, de difícil compreensão e manutenção. Para melhorar essa situação, decidimos dividi-lo em métodos menores, mais fáceis de entender e de efetuar manutenções. A esse processo de organizar e melhorar a estrutura interna de uma aplicação, denominamos de:

a) indentação.

b) depuração.

c) inspeção.

d) integração.

e) refatoração.

Q53) [VUNESP Câmara de Sertãozinho - SP 2019] Na linguagem Java, a sintaxe correta de um bloco de controle de exceção é:

a) try { // código a ser executado } catch (TipoExcecao nomeExcecao) { // tratamento da exceção }

b) try { // código a ser executado } catch (nomeExcecao: TipoExcecao) { // tratamento da exceção }

c) try { // código a ser executado } except { // tratamento da exceção }

d) if (TipoExcecao) { // código a ser executado } else { // tratamento da exceção }

e) switch (nomeExcecao){ default: // código a ser executado break; case TipoExcecao: // tratamento da exceção break; }

Q54) [IFPA IFPA 2019] Ao analisarmos uma classe Java, nos deparamos com um método que implementa diversas funcionalidades, tornando-se um método com muitas linhas de código, de difícil compreensão e manutenção. Para melhorar essa situação, decidimos dividi-lo em métodos menores, mais fáceis de entender e de efetuar manutenções. A esse processo de organizar e melhorar a estrutura interna de uma aplicação, denominamos de:

a) indentação.

b) depuração.

c) inspeção.

d) integração.

e) refatoração.

Q55) [IFPA IFPA 2019] Em relação as funções de saída nos algoritmos e JAVA respectivamente, marque a alternativa CORRETA:

a) enquanto e system.out.print

b) escreva e write

c) enquanto e while

d) escreva e system.out.print

e) leia e scanner (system.in)

Q56) [I**FPA IFPA 2019**] Em relação aos delimitadores de blocos em JAVA, marque a alternativa VERDADEIRA:

a) “ [e] ”

b) “ { ” e “ } ”

c) “ [” e “] ”

d) “Início” e “Fim”

e) “ (” e “) ”

Q55) [IFPA IFPA 2019] Em relação as funções de saída nos algoritmos e JAVA respectivamente, marque a alternativa CORRETA:

a) enquanto e system.out.print

b) escreva e write

c) enquanto e while

d) escreva e system.out.print

e) leia e scanner (system.in)

Q56) [I**FPA IFPA 2019**] Em relação aos delimitadores de blocos em JAVA, marque a alternativa VERDADEIRA:

a) “ [e] ”

b) “ { ” e “ } ”

c) “ [” e “] ”

d) “Início” e “Fim”

e) “ (” e “) ”

Q57) [IFSP IFSP 2019] Pilhas são consideradas as mais simples e também as mais importantes de todas as estruturas de dados. Define-se pilha como uma coleção de objetos que são inseridos e retirados, seguindo o princípio “o último que entra é o primeiro que sai”, também conhecido como LIFO, por seu acrônimo em inglês Last-In First-Out.

A estrutura de dados pilha é uma classe “embutida” no pacote java.util de Java. A classe (I) é uma estrutura de dados que armazena objetos Java genéricos e inclui, entre outros, o método (II) para inserir o objeto no topo da pilha e o método (III) para remover o elemento no topo da pilha e o retorna.

Os itens (I), (II) e (III) são, respectivamente:

a) java.util.Stack – push() – pop()

b) java.util.Collections – top() – remove()

c) java.util.Stack – top() – remove()

d) java.util.Collections – push() – pop()

Q57) [IFSP IFSP 2019] Pilhas são consideradas as mais simples e também as mais importantes de todas as estruturas de dados. Define-se pilha como uma coleção de objetos que são inseridos e retirados, seguindo o princípio “o último que entra é o primeiro que sai”, também conhecido como LIFO, por seu acrônimo em inglês Last-In First-Out.

A estrutura de dados pilha é uma classe “embutida” no pacote java.util de Java. A classe (I) é uma estrutura de dados que armazena objetos Java genéricos e inclui, entre outros, o método (II) para inserir o objeto no topo da pilha e o método (III) para remover o elemento no topo da pilha e o retorna.

Os itens (I), (II) e (III) são, respectivamente:

a) java.util.Stack – push() – pop()

b) java.util.Collections – top() – remove()

c) java.util.Stack – top() – remove()

d) java.util.Collections – push() – pop()

Q58) [IDECAN CRF-SP 2019] “Uma declaração do método final nunca pode mudar, assim todas as subclasses utilizam a mesma implementação do método; e chamadas métodos final são resolvidas em tempo de compilação – isso é conhecido como _____.” Acerca da linguagem Java, assinale a alternativa que completa corretamente a afirmativa anterior.

a) subclasse

b) superclasse

c) vinculação estática

d) interface de marcação

Q59) [IBADE Câmara de Porto Velho 2018] A linguagem de programação considerada como "nativa" dos ambientes Android é:

a) HTML.

b) Java .

c) Basic.

d) C++.

e) Live Code.

Q58) [IDECAN CRF-SP 2019] “Uma declaração do método final nunca pode mudar, assim todas as subclasses utilizam a mesma implementação do método; e chamadas métodos final são resolvidas em tempo de compilação – isso é conhecido como _____.” Acerca da linguagem Java, assinale a alternativa que completa corretamente a afirmativa anterior.

a) subclasse

b) superclasse

c) vinculação estática

d) interface de marcação

Q59) [IBADE Câmara de Porto Velho 2018] A linguagem de programação considerada como "nativa" dos ambientes Android é:

a) HTML.

b) Java .

c) Basic.

d) C++.

e) Live Code.

Q60) [UFES UFES 2018] O polimorfismo permite escrever programas com objetos que compartilham, direta ou indiretamente, a mesma superclasse, como se todos fossem objetos da superclasse. A linguagem Java provê vários recursos, como a declaração de métodos com o modificador final. Sobre um método final em Java, é INCORRETO afirmar:

a) Um método final em uma superclasse não pode ser sobrescrito em uma subclasse.

b) As chamadas do método final são resolvidas em tempo de execução, por meio de vinculação dinâmica.

c) Um método declarado como private é implicitamente final.

d) Um método declarado como static é implicitamente final.

e) Uma declaração do método final nunca pode mudar; dessa forma, todas as subclasses utilizam a mesma implementação do método.

Q60) [UFES UFES 2018] O polimorfismo permite escrever programas com objetos que compartilham, direta ou indiretamente, a mesma superclasse, como se todos fossem objetos da superclasse. A linguagem Java provê vários recursos, como a declaração de métodos com o modificador final. Sobre um método final em Java, é INCORRETO afirmar:

a) Um método final em uma superclasse não pode ser sobrescrito em uma subclasse.

b) As chamadas do método final são resolvidas em tempo de execução, por meio de vinculação dinâmica.

c) Um método declarado como private é implicitamente final.

d) Um método declarado como static é implicitamente final.

e) Uma declaração do método final nunca pode mudar; dessa forma, todas as subclasses utilizam a mesma implementação do método.

Q61) [FADESP IF-PA 2018]
Considere a execução dos códigos Java abaixo.

A saída correta é

a) Prova.

b) Prova x=12 e y=18.

c) x= 12 e y= 18.

d) x= 11 e y= 17.

e) x= 10 e y= 17.

S1.java

```
abstract class S1 {
    protected int x;
    protected int y;
    public void mostra(){
       System.out.print("Prova");
    }
}
```

S2.java

```
public class S2 extends S1{
    int y;
    public void mostra(){
        super.y=10;
        System.out.println("x=" +(++x)+" e "+"y= "+(y+x));
    }
    public static void main(String args[]){
        S2 s;
        s=new S2();
        s.x=10;
        s.y=6;
        s.mostra();
    }
}
```

Q61) [FADESP IF-PA 2018]
Considere a execução dos códigos Java abaixo.

A saída correta é

a) Prova.

b) Prova x=12 e y=18.

c) x= 12 e y= 18.

d) x= 11 e y= 17.

e) x= 10 e y= 17.

S1.java

```
abstract class S1 {
    protected int x;
    protected int y;
    public void mostra(){
       System.out.print("Prova");
    }
}
```

S2.java

```
public class S2 extends S1{
    int y;
    public void mostra(){
        super.y=10;
        System.out.println("x=" +(++x)+" e "+"y= "+(y+x));
    }
    public static void main(String args[]){
        S2 s;
        s=new S2();
        s.x=10;
        s.y=6;
        s.mostra();
    }
}
```

Q62) [FCC CREMESP 2016] Considere o seguinte fragmento do pseudocódigo:

```
resto ← a mod b
enquanto resto <> 0 faça
```

Em Java e PHP, mod e <> são representados, respectivamente, por:

a) % e ! =

b) \ e ! =

c) % e <>

d) & e !

e) \ e #

Q62) [FCC CREMESP 2016] Considere o seguinte fragmento do pseudocódigo:

```
resto ← a mod b
enquanto resto <> 0 faça
```

Em Java e PHP, mod e <> são representados, respectivamente, por:

a) % e ! =

b) \ e ! =

c) % e <>

d) & e !

e) \ e #

Q63) [COPS-UEL UEL 2015] Sobre a programação orientada a objetos no Java, considere as afirmativas a seguir.

I. Métodos static herdados não podem ser sobrescritos.

II. Uma classe abstrata é criada com a palavra chave abstract.

III. Uma classe abstrata está impedida de possuir métodos abstratos.

IV. Construtores static podem ser declarados como abstract.

Assinale a alternativa correta.

a) Somente as afirmativas I e II são corretas.

b) Somente as afirmativas I e IV são corretas.

c) Somente as afirmativas III e IV são corretas.

d) Somente as afirmativas I, II e III são corretas.

e) Somente as afirmativas II, III e IV são corretas.

Q63) [COPS-UEL UEL 2015] Sobre a programação orientada a objetos no Java, considere as afirmativas a seguir.

I. Métodos static herdados não podem ser sobrescritos.

II. Uma classe abstrata é criada com a palavra chave abstract.

III. Uma classe abstrata está impedida de possuir métodos abstratos.

IV. Construtores static podem ser declarados como abstract.

Assinale a alternativa correta.

a) Somente as afirmativas I e II são corretas.

b) Somente as afirmativas I e IV são corretas.

c) Somente as afirmativas III e IV são corretas.

d) Somente as afirmativas I, II e III são corretas.

e) Somente as afirmativas II, III e IV são corretas.

Q64) [CESPE MPE-PI 2018] ulgue o próximo item, relativo a lógica de programação e linguagens de programação.

A execução do código Java seguinte retornará o resultado numérico 20.

```
public class Prova {
public static int prova(int num) {
if (num <= 1) {
return 1;
} else {
return prova(num - 1) * num;
}
}
public static void main(String[] args) {
int numero;
numero = 5;
System.out.println("O resultado é " + prova(numero));
}
}
```

Q64) [CESPE MPE-PI 2018] ulgue o próximo item, relativo a lógica de programação e linguagens de programação.

A execução do código Java seguinte retornará o resultado numérico 20. ERRADO.

```
public class Prova {
public static int prova(int num) {
if (num <= 1) {
return 1;
} else {
return prova(num - 1) * num;
}
}
public static void main(String[] args) {
int numero;
numero = 5;
System.out.println("O resultado é " + prova(numero));
}
}
```

Fibonaci -> = 5 x 4 x 3 x 2 = **120.**

Q65) [CCV-UFC UFC 2016] Na orientação a objetos no Java, quando se tem um método herdado que tem seu comportamento alterado afim de torná-lo mais específico mantendo a sua mesma assinatura, tem-se a característica de:

a) Sobrescrita.

b) Sobrecarga.

c) Polimorfismo.

d) Encapsulamento.

e) Herança múltipla.

Q66) [CCV-UFC UFC 2016] Utilizando a linguagem de programação orientada a objetos Java, a partir das afirmações abaixo, qual item está correto?

- A e E são classes - B e D são interfaces - C é uma classe abstrata

a) class F implements D{ }

b) class F implements B, C{ }

c) class F extends B implements D{ }

d) class F extends E implements A{ }

e) class F extends C implements E{ }

Q65) [CCV-UFC UFC 2016] Na orientação a objetos no Java, quando se tem um método herdado que tem seu comportamento alterado afim de torná-lo mais específico mantendo a sua mesma assinatura, tem-se a característica de:

a) Sobrescrita.

b) Sobrecarga.

c) Polimorfismo.

d) Encapsulamento.

e) Herança múltipla.

Q66) [CCV-UFC UFC 2016] Utilizando a linguagem de programação orientada a objetos Java, a partir das afirmações abaixo, qual item está correto?

- A e E são classes - B e D são interfaces - C é uma classe abstrata

a) class F implements D{ }

b) class F implements B, C{ }

c) class F extends B implements D{ }

d) class F extends E implements A{ }

e) class F extends C implements E{ }

Q67) [FGV AL RO 2018] No contexto da linguagem Java, assinale o modificador (modifier) que se refere ao nível de acesso.

a) abstract

b) final

c) protected

d) static

e) volatile

Q68) [FGV AL RO 2018] Sobre construtores (constructors), no contexto da linguagem Java, analise as afirmativas a seguir.

I. Os construtores devem ser declarados como private.

II. Uma interface não pode ter um construtor.

III. Uma classe abstrata pode ter um construtor.

Está correto o que se afirma em

a) I, somente.

b) II, somente.

c) III, somente.

d) I e II, somente.

e) II e III, somente.

Q67) [FGV AL RO 2018] No contexto da linguagem Java, assinale o modificador (modifier) que se refere ao nível de acesso.

a) abstract

b) final

c) protected

d) static

e) volatile

Q68) [FGV AL RO 2018] Sobre construtores (constructors), no contexto da linguagem Java, analise as afirmativas a seguir.

I. Os construtores devem ser declarados como private.

II. Uma interface não pode ter um construtor.

III. Uma classe abstrata pode ter um construtor.

Está correto o que se afirma em

a) I, somente.

b) II, somente.

c) III, somente.

d) I e II, somente.

e) II e III, somente.

Q69) [UECE-CEV Funceme 2018] Atente para o seguinte bloco de código:

```
1  public class Gerente extends Funcionario
   implements Pessoa {
2       private int matricula ;
3       private double salario ;
4
5      public Gerente (String nome, int idade,
        int matricula, double salario) {
6           super (nome, idade);
7           this.matricula = matricula;
8           this.salario = salario;
9         }
10
11      public void mostrar () {
12        System.out.println(nome + ",” +
          matricula) ;
13      }
14   }
```

Considerando o bloco de código acima apresentado, pode-se afirmar corretamente que

a) a classe Funcionario é uma classe abstrata.

b) os atributos nome e idade são herdados de Pessoa.

c) o método mostrar não é definido na interface Pessoa.

d) a classe Funcionario possui um construtor que recebe como parâmetros um String e um inteiro.

Q69) [UECE-CEV Funceme 2018] Atente para o seguinte bloco de código:

```
1   public class Gerente extends Funcionario
    implements Pessoa {
2        private int matricula ;
3        private double salario ;
4
5       public Gerente (String nome, int idade,
         int matricula, double salario) {
6            super (nome, idade);
7            this.matricula = matricula;
8            this.salario = salario;
9         }
10
11       public void mostrar () {
12          System.out.println(nome + ",” +
            matricula) ;
13       }
14    }
```

Considerando o bloco de código acima apresentado, pode-se afirmar corretamente que

a) a classe Funcionario é uma classe abstrata.

b) os atributos nome e idade são herdados de Pessoa.

c) o método mostrar não é definido na interface Pessoa.

d) a classe Funcionario possui um construtor que recebe como parâmetros um String e um inteiro.

Q70) [CS-UFG SANEAGO 2018] Programa em Java contendo as classes A, B e C, todas elas contendo o método void m(), onde a classe A representa um conceito mais genérico que B, e a classe C representa um conceito mais específico que A. Esse programa está representado em:

a) class A extends B {}

class B extends C {void m(){}}

class C {}

b) class B extends A {}

class C extends B {void m(){}}

class A {}

c) class C extends B {}

class B {void m(){}}

class A extends B {}

d) class B extends C {}

class C extends A {}

class A {void m(){}}

Q70) [CS-UFG SANEAGO 2018] Programa em Java contendo as classes A, B e C, todas elas contendo o método void m(), onde a classe A representa um conceito mais genérico que B, e a classe C representa um conceito mais específico que A. Esse programa está representado em:

a) class A extends B {}

class B extends C {void m(){}}

class C {}

b) class B extends A {}

class C extends B {void m(){}}

class A {}

c) class C extends B {}

class B {void m(){}}

class A extends B {}

d) class B extends C {}

class C extends A {}

class A {void m(){}}

Q71) [CS-UFG SANEAGO 2018] A linguagem Java tem acrescentado recursos a cada nova versão. Qual das sentenças a seguir provoca um erro de compilação?

a) s.forEach(System.out::print);

b) for (String $s : s) System.out.println($s);

c) public static void main(String[] args);

d) private void x(final int y) {System.out.print(y);}

Q72) [IF-RS IF-RS 2018] Acerca da linguagem Java versão 6+, assinale a instrução INCORRETA, isto é, que não compila:

a) Object $o = "s";

b) Object o = new String("s");

c) Object _o = new Object();

d) String 7s = "sssssss";

e) String s = new String("s");

Q71) [CS-UFG SANEAGO 2018] A linguagem Java tem acrescentado recursos a cada nova versão. Qual das sentenças a seguir provoca um erro de compilação?

a) s.forEach(System.out::print);

b) for (String $s : s) System.out.println($s);

c) public static void main(String[] args);

d) private void x(final int y) {System.out.print(y);}

Q72) [IF-RS IF-RS 2018] Acerca da linguagem Java versão 6+, assinale a instrução INCORRETA, isto é, que não compila:

a) Object $o = "s";

b) Object o = new String("s");

c) Object _o = new Object();

d) String 7s = "sssssss";

e) String s = new String("s");

Q73) [IF-RS IF-RS 2018] Acerca da linguagem Java versão 6+, é INCORRETO afirmar que:

a) A classe "Stack" possui métodos "push", "pop" e "peek".

b) A classe "ArrayList" não possui métodos "push" e "pop".

c) Um "HashSet" não permite elementos duplicados.

d) Um "HashMap" não permite valores duplicados.

e) Um "LinkedList" permite elementos duplicados.

Q74) [IF-RS IF-RS 2018] Acerca da linguagem Java versão 6+, é CORRETO afirmar que:

a) Java possui herança múltipla.

b) Uma classe abstrata não pode ser instanciada.

c) Uma interface pode implementar duas ou mais interfaces.

d) Uma classe abstrata pode estender duas ou mais classes.

e) Todas e quaisquer classes podem ser estendidas.

Q73) [IF-RS IF-RS 2018] Acerca da linguagem Java versão 6+, é INCORRETO afirmar que:

a) A classe "Stack" possui métodos "push", "pop" e "peek".

b) A classe "ArrayList" não possui métodos "push" e "pop".

c) Um "HashSet" não permite elementos duplicados.

d) Um "HashMap" não permite valores duplicados.

e) Um "LinkedList" permite elementos duplicados.

Q74) [IF-RS IF-RS 2018] Acerca da linguagem Java versão 6+, é CORRETO afirmar que:

a) Java possui herança múltipla.

b) Uma classe abstrata não pode ser instanciada.

c) Uma interface pode implementar duas ou mais interfaces.

d) Uma classe abstrata pode estender duas ou mais classes.

e) Todas e quaisquer classes podem ser estendidas.

Q75) [UFPR UFPR 2018] Considere o programa em Java abaixo:

```
public class Teste {
    public static void main(String[] args) {
        int x=5,y=7;
        System.out.print(((y * 2) % x));
        System.out.print(" " + (y % x));
    }
}
```

A saída do programa ao ser executado será:

a) 1 1

b) 2 1

c) 4 1

d) 4 2

e) 2 4

Q75) [UFPR UFPR 2018] Considere o programa em Java abaixo:

```
public class Teste {
    public static void main(String[] args) {
        int x=5,y=7;
        System.out.print(((y * 2) % x));
        System.out.print(" " + (y % x));
    }
}
```

A saída do programa ao ser executado será:

a) 1 1

b) 2 1

c) 4 1

d) 4 2

e) 2 4

Q76) [UFPR UFPR 2018] Qual das APIs abaixo NÃO faz parte dos pacotes java.lang e java.util?

a) Math.

b) Swing.

c) Zip.

d) Collections.

e) Logging.

Q77) [FAURGS BANRISUL 2018] Assinale a afirmativa correta sobre herança em JAVA.

a) Subclasses têm acesso aos campos privados da sua superclasse.

b) Toda classe que não estenda especificamente uma outra classe é uma subclasse de Object.

c) Propriedades mais especializadas da hierarquia ficam em superclasses.

d) Subclasses herdam o comportamento e o estado da superclasse tanto na herança de classe quanto na de interface.

e) Propriedades comuns a todas as classes da hierarquia ficam em subclasses.

Q76) [UFPR UFPR 2018] Qual das APIs abaixo NÃO faz parte dos pacotes java.lang e java.util?

a) Math.

b) Swing.

c) Zip.

d) Collections.

e) Logging.

Q77) [FAURGS BANRISUL 2018] Assinale a afirmativa correta sobre herança em JAVA.

a) Subclasses têm acesso aos campos privados da sua superclasse.

b) Toda classe que não estenda especificamente uma outra classe é uma subclasse de Object.

c) Propriedades mais especializadas da hierarquia ficam em superclasses.

d) Subclasses herdam o comportamento e o estado da superclasse tanto na herança de classe quanto na de interface.

e) Propriedades comuns a todas as classes da hierarquia ficam em subclasses.

Q78) [FCC TRT2 2018] Considere a interface Java declarada abaixo.

```
public interface NewInterface {
}
```

A instrução que não causará erro se colocada no corpo desta interface (entre as chaves) é:

a) default int obterDados();

b) public NewInterface();

c) protected void consultarProcesso();

d) public void NewInterface();

e) public int aumentarSalario(int s){return s*1.1;}

Q78) [FCC TRT2 2018] Considere a interface Java declarada abaixo.

```
public interface NewInterface {
}
```

A instrução que não causará erro se colocada no corpo desta interface (entre as chaves) é:

a) default int obterDados();

b) public NewInterface();

c) protected void consultarProcesso();

d) public void NewInterface();

e) public int aumentarSalario(int s){return s*1.1;}

Q79) [FGV MPE AL 2018] Sobre as variáveis e os métodos declarados como private, em Java, analise as afirmativas a seguir. I. Ficam acessíveis somente aos membros da própria classe. II. Ficam acessíveis somente às classes definidas no mesmo package. III. Ficam acessíveis somente para suas classes derivadas. Está correto o que se afirma em

a) I, apenas.

b) II, apenas.

c) III, apenas.

d) I e II, apenas.

e) II e III, apenas.

Q80) [**FGV MPE AL 2018**] No Java, a classe Error e Exception derivam da classe

a) ClassNotFoundException.

b) IOException.

c) MainException.

d) RuntimeException.

e) Throwable.

Q79) [FGV MPE AL 2018] Sobre as variáveis e os métodos declarados como private, em Java, analise as afirmativas a seguir. I. Ficam acessíveis somente aos membros da própria classe. II. Ficam acessíveis somente às classes definidas no mesmo package. III. Ficam acessíveis somente para suas classes derivadas. Está correto o que se afirma em

a) I, apenas.

b) II, apenas.

c) III, apenas.

d) I e II, apenas.

e) II e III, apenas.

Q80) [**FGV MPE AL 2018**] No Java, a classe Error e Exception derivam da classe

a) ClassNotFoundException.

b) IOException.

c) MainException.

d) RuntimeException.

e) Throwable.

Q81) [FUNDEP CODEMIG 2018] Qual função do ambiente de programação Java deve ser utilizada para se retornar o caractere de uma determinada posição da string?

a) valueOf

b) compateTo

c) charAt

d) split

Q82) [SUGEP-UFRPE UFRPE 2018] Qual das alternativas abaixo apresenta um método que concatena dois objetos do tipo String em Java?

a) Método ‘endsWith’ em ‘String’

b) Método ‘charAt’

c) Sobrecarga do operador ‘+’

d) Método ‘substring’

e) Sobrecarga do operador ‘++’

Q81) [FUNDEP CODEMIG 2018] Qual função do ambiente de programação Java deve ser utilizada para se retornar o caractere de uma determinada posição da string?

a) valueOf

b) compateTo

c) charAt

d) split

Indexação de uma string começa pela posição 0.

Q82) [SUGEP-UFRPE UFRPE 2018] Qual das alternativas abaixo apresenta um método que concatena dois objetos do tipo String em Java?

a) Método ‘endsWith’ em ‘String’

b) Método ‘charAt’

c) Sobrecarga do operador ‘+’

d) Método ‘substring’

e) Sobrecarga do operador ‘++’

Q83) [SUGEP-UFRPE UFRPE 2018] No que diz respeito à linguagem de programação Java, analise as afirmações abaixo.

1) Um método declarado como ‘static’ não pode acessar variáveis de instância da classe, pois o método pode ser chamado mesmo quando não há nenhum objeto da classe instanciado.

2) Java permite herança múltipla de classes, portanto uma nova classe pode ser herdeira de duas ou mais classes já definidas.

3) O operador ‘+’ só pode ser utilizado para dados de tipo numérico.

Está(ão) correta(s), apenas:

a) 1.

b) 2.

c) 1 e 2.

d) 1 e 3.

e) 2 e 3.

Q83) [SUGEP-UFRPE UFRPE 2018] No que diz respeito à linguagem de programação Java, analise as afirmações abaixo.

1) Um método declarado como ‘static’ não pode acessar variáveis de instância da classe, pois o método pode ser chamado mesmo quando não há nenhum objeto da classe instanciado.

2) Java permite herança múltipla de classes, portanto uma nova classe pode ser herdeira de duas ou mais classes já definidas.

3) O operador ‘+’ só pode ser utilizado para dados de tipo numérico.

Está(ão) correta(s), apenas:

a) 1.

b) 2.

c) 1 e 2.

d) 1 e 3.

e) 2 e 3.

Q84) [FCC SABESP 2018] As interfaces são usadas nas aplicações Java quando se deseja permitir que diversas classes implementem determinados métodos, mesmo que de formas diferentes. Em uma interface Java

a) os métodos não podem ter os modificadores protected ou private.

b) não pode haver assinaturas de métodos cujo tipo de retorno seja void.

c) pode haver múltiplos construtores, desde que recebam parâmetros diferentes.

d) não pode haver dois ou mais métodos com o mesmo nome, mesmo que recebam parâmetros diferentes.

e) todo método deverá ser implementado por uma das subclasses da aplicação pelo menos uma vez.

Q84) [FCC SABESP 2018] As interfaces são usadas nas aplicações Java quando se deseja permitir que diversas classes implementem determinados métodos, mesmo que de formas diferentes. Em uma interface Java

a) os métodos não podem ter os modificadores protected ou private.

b) não pode haver assinaturas de métodos cujo tipo de retorno seja void.

c) pode haver múltiplos construtores, desde que recebam parâmetros diferentes.

d) não pode haver dois ou mais métodos com o mesmo nome, mesmo que recebam parâmetros diferentes.

e) todo método deverá ser implementado por uma das subclasses da aplicação pelo menos uma vez.

Q85) [FGV Banestes 2018] Analise o código Java a seguir.

```
public class Test
{
    public void method()
    {
        for(int i = 0; i < 3; i++){
            System.out.print(i);
        }
        System.out.print(i);
    }
}
```

A execução desse código provoca:

a) a exibição dos números 012;

b) a exibição dos números 0122;

c) a exibição dos números 0123;

d) erro em tempo de compilação;

e) erro em tempo de execução.

Q85) [FGV Banestes 2018] Analise o código Java a seguir.

```
public class Test
{
    public void method()
    {
        for(int i = 0; i < 3; i++){
            System.out.print(i);
        }
        System.out.print(i);
    }
}
```

A execução desse código provoca:

a) a exibição dos números 012;

b) a exibição dos números 0122;

c) a exibição dos números 0123;

d) erro em tempo de compilação;

e) erro em tempo de execução.

A variável i é uma variável local dentro do for. Ela só terá acesso dentro do escopo do for.

Q86) [FGV Banestes 2018] Considere a compilação de um ou mais programas por meio da linha de comando, num ambiente Java. Nesse caso, o comando que está corretamente formado para esse fim é:

a) compile teste.java –type java

b) java teste.java

c) javac *.java

d) jvm Teste1.java teste2.java

e) parse java teste.java

Q87) [CESPE EBSERH 2018] Acerca de programação orientada a objetos, Java e PHP, julgue o item a seguir.

O suporte para a implementação de diversas interfaces em uma única classe é considerado uma solução alternativa para contornar a restrição de herança única própria da linguagem Java.

Q86) [FGV Banestes 2018] Considere a compilação de um ou mais programas por meio da linha de comando, num ambiente Java. Nesse caso, o comando que está corretamente formado para esse fim é:

a) compile teste.java –type java

b) java teste.java

c) javac *.java

d) jvm Teste1.java teste2.java

e) parse java teste.java

Q87) [CESPE EBSERH 2018] Acerca de programação orientada a objetos, Java e PHP, julgue o item a seguir.

O suporte para a implementação de diversas interfaces em uma única classe é considerado uma solução alternativa para contornar a restrição de herança única própria da linguagem Java. CERTO.

Q88) [CESGRANRIO Basa 2018] Considere o código Java listado a seguir, onde a numeração de linhas está sendo utilizada apenas como referência:

```
i.   package teste;
ii.  public class Classe {
iii. private static int a = 5;
iv.  public static void main(String[] args) {
v.       int a = 8;
vi.
vii. }
viii.}
```

Que comando deve ser inserido na linha vi para exibir o valor 5 na console?

a) System.out.println(a);

b) System.out.println(Classe.a);

c) System.out.println(Integer.parseInt(a));

d) System.out.println(super.a);

e) System.out.println(this.a);

Q88) [CESGRANRIO Basa 2018] Considere o código Java listado a seguir, onde a numeração de linhas está sendo utilizada apenas como referência:

```
i.   package teste;
ii.  public class Classe {
iii. private static int a = 5;
iv.  public static void main(String[] args) {
v.       int a = 8;
vi.
vii. }
viii.}
```

Que comando deve ser inserido na linha vi para exibir o valor 5 na console?

a) System.out.println(a);

b) System.out.println(Classe.a);

c) System.out.println(Integer.parseInt(a));

d) System.out.println(super.a);

e) System.out.println(this.a);

Q89) [CS-UFG UFG 2018] O seguinte trecho de código cria um array em linguagem Java e o inicializa:

int[] array; array = new int[2];

int[0] = 1; int[1] = 2; int[2] = 3;

Este array é exemplo de

a) uma variável composta homogênea.

b) uma expressão.

c) uma string heterogênea.

d) uma constante.

Q90) [CESPE STJ 2018] Julgue o item que se segue, a respeito de EJB, Clean Code, desenvolvimento orientado a testes, lógica de programação e paradigmas de programação.

Em virtude do polimorfismo implementado na linguagem Java, um método da forma empregado.calculasalario(), tem sua invocação resolvida em tempo de compilação do código.

Q89) [CS-UFG UFG 2018] O seguinte trecho de código cria um array em linguagem Java e o inicializa:

```
int[] array; array = new int[2];
int[0] = 1; int[1] = 2; int[2] = 3;
```

Este array é exemplo de

a) uma variável composta homogênea.

b) uma expressão.

c) uma string heterogênea.

d) uma constante.

Q90) [CESPE STJ 2018] Julgue o item que se segue, a respeito de EJB, Clean Code, desenvolvimento orientado a testes, lógica de programação e paradigmas de programação.

Em virtude do polimorfismo implementado na linguagem Java, um método da forma empregado.calculasalario(), tem sua invocação resolvida em tempo de compilação do código. ERRADO.

Q91) [IADES CFM 2018] O fragmento de código a seguir, escrito em Java, descreve o uso de alguns artifícios que essa linguagem fornece para o programador:

```
public class Cachorro extends Mamifero implements AnimalDomesticado, AnimalEstimacao {
        private int tamanho;
        private String raça;

    public Cachorro(String nome ,String raça, int tamanho){
        super.setNome(nome);
        this.raça=raça;
        this.tamanho=tamanho;
    }
        .
        .
        .
}// end class Cachorro
```

Com base no fragmento de código apresentado, assinale a alternativa correta.

a) Mamifero é uma interface e AnimalDomesticado e AnimalEstimacao são superclasses utilizadas pela subclasse Cachorro.

b) Cachorro é uma interface e AnimalDomesticado e AnimalEstimacao são superclasses utilizadas pela subclasse Mamifero.

c) A palavra private significa que os atributos poderão ser acessados por qualquer outra classe Java.

d) A classe Cachorro utiliza os recursos de herança e interface.

e) Na linguagem Java existe o recurso de múltiplas heranças.

Q91) [IADES CFM 2018] O fragmento de código a seguir, escrito em Java, descreve o uso de alguns artifícios que essa linguagem fornece para o programador:

```
public class Cachorro extends Mamifero implements AnimalDomesticado, AnimalEstimacao {
        private int tamanho;
        private String raça;

    public Cachorro(String nome ,String raça, int tamanho){
        super.setNome(nome);
        this.raça=raça;
        this.tamanho=tamanho;
    }
        .
        .
        .
}// end class Cachorro
```

Com base no fragmento de código apresentado, assinale a alternativa correta.

a) Mamifero é uma interface e AnimalDomesticado e AnimalEstimacao são superclasses utilizadas pela subclasse Cachorro.

b) Cachorro é uma interface e AnimalDomesticado e AnimalEstimacao são superclasses utilizadas pela subclasse Mamifero.

c) A palavra private significa que os atributos poderão ser acessados por qualquer outra classe Java.

d) A classe Cachorro utiliza os recursos de herança e interface.

e) Na linguagem Java existe o recurso de múltiplas heranças.

Q92) [CESPE ABIN 2018] Acerca da linguagem Java, julgue o próximo item.

O método no código

System.out.println(valor.charAt(0));

retornará todos os caracteres referentes à string avaliada

Q93) [CS-UFG SANEAGO 2018] Com base nas boas práticas da programação e manutenção de software orientado a objetos em JAVA, uma programadora deve escolher uma opção para explicitar que um método do cálculo matemático falhou, tendo em vista o estado das entradas ou da aplicação. Qual é essa opção?

a) Retornar uma constante relacionada a código de erro.

b) Realizar o lançamento de exceção.

c) Retornar nulo.

d) Imprimir uma mensagem de erro, usando o método System.out.println.

Q92) [CESPE ABIN 2018] Acerca da linguagem Java, julgue o próximo item.

O método no código

System.out.println(valor.charAt(0));

retornará todos os caracteres referentes à string avaliada ERRADO.

Q93) [CS-UFG SANEAGO 2018] Com base nas boas práticas da programação e manutenção de software orientado a objetos em JAVA, uma programadora deve escolher uma opção para explicitar que um método do cálculo matemático falhou, tendo em vista o estado das entradas ou da aplicação. Qual é essa opção?

a) Retornar uma constante relacionada a código de erro.

b) Realizar o lançamento de exceção.

c) Retornar nulo.

d) Imprimir uma mensagem de erro, usando o método System.out.println.

Q94) [CS-UFG SANEAGO 2018] No linguagem JAVA 9, uma interface não pode

a) herdar de múltiplas interfaces.

b) conter métodos com implementações padrão (finais).

c) herdar de uma classe abstrata.

d) conter métodos privados.

Q95) [FUNDEP CODEMIG 2018] Analise a linha do programa Java a seguir.

1. public class testando { 2. public doubly somar(doubly a, doubly b) { 3. doublya=5.11; 4. doublyb=5.122; 5. return doublya + doublyb; 6. } 7. System.out.println(somar); 8. }

Qual linha desse programa apresentaria erro durante a compilação?

a) Linha 1.

b) Linha 2.

c) Linha 5.

d) Linha 7.

Q94) [CS-UFG SANEAGO 2018] No linguagem JAVA 9, uma interface não pode

a) herdar de múltiplas interfaces.

b) conter métodos com implementações padrão (finais).

c) herdar de uma classe abstrata.

d) conter métodos privados.

Q95) [FUNDEP CODEMIG 2018] Analise a linha do programa Java a seguir.

1. public class testando { 2. public doubly somar(doubly a, doubly b) { 3. doublya=5.11; 4. doublyb=5.122; 5. return doublya + doublyb; 6. } 7. System.out.println(somar); 8. }

Qual linha desse programa apresentaria erro durante a compilação?

a) Linha 1.

b) Linha 2.

c) Linha 5.

d) Linha 7.

Q96) [CONSULPLAN Câmara BH 2018] Acerca da linguagem de programação Java, “um método declarado _____ não pode acessar as variáveis de instância e os métodos de instância da classe, porque um método _____ pode ser chamado mesmo quando nenhum objeto da classe foi instanciado.” Assinale a alternativa que completa correta e sequencialmente a afirmativa anterior.

a) static / static

b) public / static

c) static / public

d) public / public

Q96) [CONSULPLAN Câmara BH 2018] Acerca da linguagem de programação Java, “um método declarado _____ não pode acessar as variáveis de instância e os métodos de instância da classe, porque um método _____ pode ser chamado mesmo quando nenhum objeto da classe foi instanciado.” Assinale a alternativa que completa correta e sequencialmente a afirmativa anterior.

a) static / static

b) public / static

c) static / public

d) public / public

Q97) [CONSULPLAN Câmara BH 2018] Sobre a linguagem Java, em relação à entrada/saída e operadores, assinale a alternativa INCORRETA.

a) As condições em instruções if podem ser formadas utilizando-se os operadores de igualdade (== e !=) e relacionais (>, <, >= e <=).

b) Uma instrução if começa com a palavra-chave if, seguida por uma condição entre parênteses, e espera uma instrução no seu corpo.

c) Uma barra (/) em uma string é um caractere de escape. O Java o combina com o próximo caractere para formar uma sequência de escape. A sequência de escape /n representa o caractere de nova linha.

d) Variáveis do tipo char representam caracteres individuais, como uma letra maiúscula (por exemplo, A), um dígito (por exemplo, 7), um caractere especial (por exemplo, * ou %), ou uma sequência de escape (por exemplo, tab, \t).

Q97) [CONSULPLAN Câmara BH 2018] Sobre a linguagem Java, em relação à entrada/saída e operadores, assinale a alternativa INCORRETA.

a) As condições em instruções if podem ser formadas utilizando-se os operadores de igualdade (== e !=) e relacionais (>, <, >= e <=).

b) Uma instrução if começa com a palavra-chave if, seguida por uma condição entre parênteses, e espera uma instrução no seu corpo.

c) Uma barra (/) em uma string é um caractere de escape. O Java o combina com o próximo caractere para formar uma sequência de escape. A sequência de escape /n representa o caractere de nova linha.

d) Variáveis do tipo char representam caracteres individuais, como uma letra maiúscula (por exemplo, A), um dígito (por exemplo, 7), um caractere especial (por exemplo, * ou %), ou uma sequência de escape (por exemplo, tab, \t).

Q98) [CONSULPLAN Câmara BH 2018] String é um tipo texto que corresponde à união de um conjunto de caracteres. Em Java, uma variável do tipo string é uma instância da classe String, isto é, gera objetos que possuem propriedades e métodos, diferentemente dos tipos primitivos como int, float, double etc. Acerca das operações com strings, em Java, este método “remove todos os espaços em branco que aparecem no início e no final de uma determinada string, porém não são removidos os espaços entre as palavras”; assinale-o.

a) Trim.

b) Charat.

c) Replace.

d) Substring.

Q99) [INAZ do Pará CFF 2017] Como qualquer linguagem de programação, a linguagem Java tem sua própria estrutura, regras de sintaxe e paradigma de programação. Portanto, a estrutura try...catch...finally da linguagem Java tem o objetivo de controlar o fluxo de execução do tipo:

a) Estrutura de desvio de fluxo.

b) Mecanismo de finalização.

c) Estrutura de controle de erros.

d) Estrutura de repetição condicional.

e) Mecanismo de inicialização de métodos.

Q98) [CONSULPLAN Câmara BH 2018] String é um tipo texto que corresponde à união de um conjunto de caracteres. Em Java, uma variável do tipo string é uma instância da classe String, isto é, gera objetos que possuem propriedades e métodos, diferentemente dos tipos primitivos como int, float, double etc. Acerca das operações com strings, em Java, este método “remove todos os espaços em branco que aparecem no início e no final de uma determinada string, porém não são removidos os espaços entre as palavras”; assinale-o.

a) Trim.

b) Charat.

c) Replace.

d) Substring.

Q99) [INAZ do Pará CFF 2017] Como qualquer linguagem de programação, a linguagem Java tem sua própria estrutura, regras de sintaxe e paradigma de programação. Portanto, a estrutura try...catch...finally da linguagem Java tem o objetivo de controlar o fluxo de execução do tipo:

a) Estrutura de desvio de fluxo.

b) Mecanismo de finalização.

c) Estrutura de controle de erros.

d) Estrutura de repetição condicional.

e) Mecanismo de inicialização de métodos.

Q100) [CONSULPLAN TRE RJ 2017] Acerca da programação orientada a objetos, usando Java, analise a seguinte assertiva: “O Java contém três tipos de instruções de seleção”. Assinale-as.

a) if; for; while.

b) if; while; do while.

c) while; switch; else.

d) if; if ... else; switch.

Q101) [FCC TST 2017] A instrução r ← r + 1.0 / aux, na linguagem Java, é corretamente escrita na forma

a) r = r++/aux;

b) r += 1.0/aux;

c) r = r + Math.div(1.0,aux);

d) r = r + 1 % aux;

e) r += Math.div(1,aux);

Q100) [CONSULPLAN TRE RJ 2017] Acerca da programação orientada a objetos, usando Java, analise a seguinte assertiva: “O Java contém três tipos de instruções de seleção”. Assinale-as.

a) if; for; while.

b) if; while; do while.

c) while; switch; else.

d) if; if ... else; switch.

Q101) [FCC TST 2017] A instrução r ← r + 1.0 / aux, na linguagem Java, é corretamente escrita na forma

a) r = r++/aux;

b) r += 1.0/aux;

c) r = r + Math.div(1.0,aux);

d) r = r + 1 % aux;

e) r += Math.div(1,aux);

GABARITO

Q1 - LETRA B
Q2 - LETRA D
Q3 - LETRA D
Q4 - LETRA D
Q5 - LETRA A
Q6 - LETRA C
Q7 - LETRA B
Q8 - LETRA B
Q9 - CERTO
Q10 - ERRADO
Q11 - ERRADO
Q12 - CERTO
Q13 - LETRA D
Q14 - LETRA A
Q15 - LETRA C
Q16 - LETRA B
Q17 - LETRA A
Q18 - LETRA B
Q19 - LETRA C
Q20 - LETRA A
Q21 - LETRA B
Q22 - LETRA C
Q23 - LETRA B
Q24 - LETRA C
Q25 - LETRA A
Q26 - LETRA C
Q27 - LETRA C
Q28 - LETRA E
Q28 - LETRA A
Q29 - LETRA C
Q30 - LETRA C
Q31 - LETRA C
Q32 - LETRA B
Q33 - LETRA D
Q34 - LETRA E
Q35 - LETRA E
Q36 - LETRA B
Q37 - LETRA D
Q38 - LETRA A
Q39 - LETRA B
Q40 - LETRA B
Q41 - LETRA A
Q42 - LETRA C
Q43 - LETRA A
Q44 - LETRA C
Q45 - LETRA A
Q46 - LETRA D
Q47 - LETRA D
Q48 - LETRA E
Q49 - LETRA B
Q50 - LETRA A
Q51 - LETRA A
Q52 - LETRA D
Q53 - LETRA A
Q54 - LETRA E
Q55 - LETRA D
Q56 - LETRA B
Q57 - LETRA A
Q58 - LETRA C
Q59 - LETRA B
Q60 - LETRA B
Q61 - LETRA D
Q62 - LETRA A
Q63 - LETRA A
Q64 - ERRADO
Q65 - LETRA A
Q66 - LETRA A
Q67 - LETRA C
Q68 - LETRA E
Q69 - LETRA D
Q70 - LETRA D
Q71 - LETRA C
Q72 - LETRA D
Q73 - LETRA D
Q74 - LETRA B
Q75 - LETRA D
Q76 - LETRA B
Q77 - LETRA B
Q78 - LETRA D
Q79 - LETRA A
Q80 - LETRA E
Q81 - LETRA C
Q82 - LETRA C
Q83 - LETRA A
Q84 - LETRA A
Q85 - LETRA D
Q86 - LETRA C
Q87 - CERTO
Q88 - LETRA B
Q89 - LETRA A
Q90 - ERRADO
Q91 - LETRA A
Q92 - ERRADO
Q93 - LETRA B
Q94 - LETRA C
Q95 - LETRA B
Q96 - LETRA A
Q97 - LETRA C
Q98 - LETRA A
Q99 - LETRA C
Q100 - LETRA D
Q101 - LETRA B