

**INSTRUÇÕES:** Essa prova contém 25 questões. Verifique se a prova está completa e legível. Qualquer problema, fale com o fiscal de sala. Marque as respostas no cartão usando caneta azul ou preta. A duração da prova é de 3 horas.

- 1) Ao entrar numa floresta, Alice perdeu a noção dos dias da semana. O leão tornou-se amigo de Alice. Ela sabia que o leão mentia às segundas, terças e quartas e dizia a verdade nos outros dias da semana. Um dia o leão disse para a Alice "Eu mentirei daqui a três dias" e disse também "Eu menti ontem". Qual era o dia da semana?
- a) Segunda-feira.      b) Terça-feira.      c) Sábado.      d) Quinta-feira.      e) Domingo.
- 2) Determine em que situação a seguinte frase é falsa: "**Se um político mente e rouba então ele vai preso ou é cassado**".
- a) Um político mente e rouba, é cassado mas não é preso.  
b) Um político não rouba, mas mente, no entanto não é preso nem é cassado.  
c) Um político mente e rouba mas não é preso nem é cassado.  
d) Um político mente e rouba, é preso e cassado.  
e) Um político não mente, mas rouba, no entanto não é preso nem é cassado.
- 3) Determine a negação da seguinte frase: "**Todos os números perfeitos são pares e existem infinitos números perfeitos**".
- a) Todos os números perfeitos são ímpares ou existe uma quantidade finita de números perfeitos.  
b) Nem todos os números perfeitos são pares e a quantidade de números perfeitos é finita.  
c) Não é verdade que todos os números perfeitos são pares nem é verdade que existem infinitos números perfeitos.  
d) Existem números perfeitos ímpares ou a quantidade de números perfeitos é finita.  
e) Nenhuma das alternativas anteriores.
- 4) Considere a seguinte sentença: "**Essa frase é falsa**". Sobre essa sentença, é correto afirmar que:
- a) A sentença é verdadeira.  
b) A sentença é falsa.  
c) A sentença é uma tautologia.  
d) A sentença é um paradoxo.  
e) Nenhuma das alternativas anteriores é correta.
- 5) Suponha que as seguintes afirmações são verdadeiras: "**Se o sol está brilhando e o clima está ameno então João fica feliz e satisfeito**" e "**João está feliz**". Podemos concluir que:
- a) O sol está brilhando.  
b) O clima está ameno.  
c) João está satisfeito.  
d) Todas as alternativas anteriores são corretas.  
e) Nenhuma das alternativas anteriores é correta.
- 6) Em uma cidade, cada homem ou é cavaleiro ou é bandido. Os cavaleiros sempre dizem a verdade, já os bandidos nunca dizem a verdade. Você está conversando com Rui e Gil, dois homens da cidade que se conhecem, e Rui diz: "**Se Gil é um cavaleiro então eu sou um bandido**". Baseado nessa afirmação, concluímos que:
- a) Rui é bandido e Gil é cavaleiro.  
b) Rui é cavaleiro e Gil é bandido.  
c) Ambos são bandidos.  
d) Ambos são cavaleiros.  
e) Não podemos afirmar se Rui é bandido ou cavaleiro.



7) Supondo que a frase "Todo ser humano é mortal" é verdadeira, podemos concluir que:

- a) Se algo não é mortal então não é ser humano.
- b) Se algo é mortal então é ser humano.
- c) Se algo não é ser humano então não é mortal.
- d) Todas as alternativas anteriores são corretas.
- e) Nenhuma das alternativas anteriores é correta.

8) Suponha que cada trabalhador constrói dois metros de muro em um dia de trabalho. Quantos dias de trabalho serão necessários para que três trabalhadores construam doze metros de muro?

- a) 1 dia.
- b) 2 dias.
- c) 3 dias.
- d) 4 dias.
- e) 5 dias.

9) A condição " $(x \geq y \text{ e } y \geq z)$  ou  $x < z$ " é verdadeira se:

- a)  $x = 8, y = 5 \text{ e } z = 6$ .
- b)  $x = 7, y = 10 \text{ e } z = 6$ .
- c)  $x = 9, y = 5 \text{ e } z = 8$ .
- d)  $x = 1, y = 2, z = 1$ .

e)  $x = y$ .

10) Suponha ser verdadeira a seguinte sentença: "Se um número é quadrado perfeito então esse número não é primo". Seja  $x$  um número. É correto afirmar que:

- a) Se  $x$  não for quadrado perfeito então  $x$  é primo.  $\sim a \rightarrow p$
- b) Se  $x$  não for primo então  $x$  é quadrado perfeito.  $\sim p \rightarrow a$
- c) Se  $x$  for primo então  $x$  não é quadrado perfeito.  $p \rightarrow \sim a$
- d) Se  $x$  não for primo então  $x$  não é quadrado perfeito.  $\sim p \rightarrow \sim a$
- e) Nenhuma das alternativas anteriores é correta.

11) Foi na terceira geração das linguagens de programação, conhecidas também como linguagens de alto nível, que surgiu a programação estruturada. O advento deste paradigma, em detrimento das linguagens de montagem, foi um grande avanço, melhorando a clareza dos códigos, a qualidade e o tempo de desenvolvimento. Sobre a programação estruturada, considere as seguintes afirmações:

- I. Faz extenso uso de rotinas ou sub-rotinas de código.
- II. Entre os principais motivos para este aumento na velocidade de desenvolvimento estão a modularização, a repetição e a herança.
- III. Por ser puramente orientada a objetos, não é possível aplicar os conceitos da programação estruturada à linguagem JAVA.
- IV. Segundo os conceitos da programação estruturada, todos os programas podem ser vistos como uma composição das seguintes estruturas: sequências ordenadas de declarações ou sub-rotinas, seleção de trechos que dependem do estado do programa e repetição de blocos de código.

Marque a alternativa que contém a(s) afirmação(ões) correta(s):

- a) III e IV.
- b) II e III.
- c) Somente a IV.
- d) I, II e III.
- e) I e IV.

12) Sobre a linguagem de programação C, assinale a alternativa correta:

a) Dentre os seus tipos primitivos, podemos citar: *int*, *float*, *double* e *string*.

b) Por ser de baixo nível, tendo extenso uso em microcontroladores e sistemas embarcados, pode-se classificar a linguagem como sendo de propósito específico.

c) Devido à tipagem estática da linguagem, não é possível fazer conversões implícitas de tipos.

d) Por ser uma linguagem estruturada, e não orientada a objetos, o C não dá suporte para que dados relacionados sejam combinados e manipulados como um todo.

e) Sabendo que a linguagem só permite passagem de parâmetros por valor, é necessário utilizar-se de ponteiros para que uma variável local externa a uma função, ao ser passada como parâmetro para a mesma, possa ter seu valor modificado dentro dessa função.

$5 + 28 = 33$   
 $5 + 28 = 33$



o programa abaixo, em C, assinale a alternativa que contém o trecho de código que deve substituir o comentário, a fim de que se obtenha a saída esperada:

```
#include <stdio.h>

int main () {
    int v[6], *p, i;
    for (i = 0; i < 6; ++i) {
        *(v + i) = i;
    }

    /* <LINHAS DE CÓDIGO> */

    return 0;
}
```

SAÍDA ESPERADA:

0 1 2 3 4 5

- a) for (i = 0; i < 6; i++) { printf("%d ", &v[i]); }  
b) for (p = v; p <= &v[5]; p++) { printf("%d ", \*p); }  
c) for (i = 0; i < v.size(); i++) { printf("%d ", v[i]); }  
d) for (p = v; p <= &v[5]; p++) { printf("%d ", &p); }  
e) for (i = 0; i < 6; i++) { printf("%d ", \*(p + i)); }

14) Marque a alternativa que apresenta corretamente a saída do programa abaixo, escrito utilizando a linguagem C.

```
#include <stdio.h>

void p(int *p1, int p2) { *p1 = p2; }

int main(int argc, const char * argv[]) {
    int *x, **y;
    int a = 10;
    x = &a;
    y = &x;
    p(x, 5);
    printf("%d", *x);
    p(*y, 4);
    printf("%d", **y);
    return 0;
}
```

\*x 10  
\*\*y 10  
a 10  
5

- a) 10, 10      b) 5, 10      c) 5, 4      d) 10, 4      e) O programa não compila.

15) Considere as sentenças abaixo para marcar a alternativa que apresenta a(s) sentença(s) verdadeira(s).

- I. Uma pilha é uma estrutura de dados na qual o elemento a ser removido é o que está na estrutura há menos tempo. Uma pilha pode ser implementada com vetor, mas não com lista encadeada.  
II. Uma lista duplamente encadeada é uma estrutura de dados em que cada nó da lista aponta apenas para o próximo nó da lista e, portanto, o acesso aos nós anterior e posterior deste nó é possível a partir do mesmo.  
III. Uma fila é uma estrutura de dados na qual o primeiro elemento inserido é também o primeiro a ser removido.

- a) I, II e III.      b) I e II.      c) I e III.      d) II.      e) III.

16) Considere o seguinte trecho de código em linguagem C referente a uma lista duplamente encadeada.

```
typedef struct no{
    int valor;
    struct no *ant, *prox;
} NO;

NO *insere(NO *inicio, NO *no){
    if(inicio == NULL){
        no->ant = no->prox = NULL;
        inicio = no;
    }else{
        /* <LINHAS DE CÓDIGO> */
    }
    return inicio;
}
```

Observe que na cláusula *else* da função *insere*, que visa inserir um novo elemento no início da lista, falta um trecho de código. Marque a alternativa que contém as linhas de código que devem substituir o comentário, de modo a manter o duplo encadeamento e fazer com que o nó inserido seja o primeiro elemento da lista e seja acessível através do ponteiro *inicio*.

a) `no->ant = NULL;`  
`no->prox = inicio;`  
`inicio->ant = NULL;`

b) `no->ant = NULL;`  
`no->prox = inicio;`  
`inicio->ant = no;`  
`inicio = no;`

c) `no->ant = NULL;`  
`inicio->ant = no;`  
`inicio = no;`

d) `no->ant = NULL;`  
`no->prox = inicio;`  
`inicio->ant = NULL;`  
`inicio = no;`

e) Nenhuma das alternativas anteriores.

17) O código em Java abaixo pode ser usado para implementar uma pilha.

```
public class Pilha {
    private int topo;
    private Object pilha[];

    public Pilha(int tamanho){
        pilha = new Object[tamanho];
        topo = -1;
    }

    public Object push(Object obj) throws Exception{
        if(topo + 1 == pilha.length){
            throw new Exception();
        }
        /* <LINHAS DE CÓDIGO> */
    }

    public Object pop() throws Exception{
        if(topo == -1) throw new Exception();
        Object obj = pilha[topo];
        pilha[topo] = null;
        topo--;
        return obj;
    }

    public String toString() {
        String aux = "[";
        for (int i = 0; i <= topo; i++) {
            aux += pilha[i] + " ";
        }
        return aux + "]";
    }
}
```

Observe que algum código deve ser adicionado ao método *push* da classe *Pilha* para que o mesmo permita inserir um novo elemento na pilha. Assim, marque a alternativa sintaticamente e semanticamente correta que contém o código que deve substituir o comentário no método *push*.

a) `topo++;`  
`pilha[topo] = obj;`

b) `topo--;`  
`pilha[topo] = obj;`  
`return obj;`

c) `topo--;`  
`pilha[topo] = obj;`

d) `topo++;`  
`pilha[topo] = obj;`  
`return obj;`

e) Nenhuma das alternativas.



- 18) Considere a classe *Pilha* declarada na questão anterior com o método *push* completado de forma correta. Considere o código em Java abaixo para marcar a alternativa que mostra o que será impresso após sua execução.

```
public class AppPilha {
    public static void main(String[] args) {
        Pilha p1 = new Pilha(15);
        Pilha p2 = new Pilha(15);
        Integer v[] = {3, 4, 1, 2, 6, 0, 8, 5, 7, 9};
        try {
            for (int i = 0; i < v.length; i++) {
                p1.push(v[i]);
                if (v[i] % 2 == 0) {
                    p2.push(v[i]);
                }
            }
            System.out.println(p1);
            System.out.println(p2);
            int soma = 0;
            for (int i = 0; i < 5; i++) {
                int num = (int) p1.pop();
                soma = soma + num;
                if (soma % 2 == 0) {
                    p2.push(num);
                }
            }
            System.out.println(p1);
            System.out.println(p2);
        } catch (Exception e) { System.out.println("Erro!!!"); }
    }
}
```

- a) [ 3 4 1 2 6 0 8 5 7 9 ]  
[ 4 2 6 8 ]  
[ 3 4 1 2 6 ]  
[ 4 2 6 0 8 7 ]
- b) [ 3 4 1 2 6 0 8 5 7 9 ]  
[ 4 2 6 0 8 ]  
[ 3 4 1 2 6 ]  
[ 4 2 6 8 7 ]
- c) [ 3 4 1 2 6 0 8 5 7 9 ]  
[ 4 2 6 0 8 ]  
[ 3 4 1 2 6 ]  
[ 4 2 6 0 8 ]
- d) [ 3 4 1 2 6 0 8 5 7 9 ]  
[ 4 2 6 0 8 ]  
[ 3 4 1 2 6 ]  
[ 4 2 6 0 8 7 ]
- e) Nenhuma das alternativas.

- 19) Considere o trecho de código em linguagem C apresentado abaixo para escolher a alternativa correta.

```
include <stdlib.h>
typedef struct { char *codigo, descricao[81]; } ESTADO;
typedef struct { char *codigo, descricao[81]; ESTADO *estado; } CIDADE;

int main(){
    ESTADO *e = malloc(sizeof(CIDADE));
    CIDADE c;
    /* <LINHAS DE CÓDIGO> */
    return 0;
}
```

- a) O código acima não compila, pois as *structs* apresentadas no código não foram declaradas de forma correta.
- b) O código acima não compila, pois no código há alocação dinâmica de memória para uma variável do tipo *struct*.
- c) A declaração "CIDADE c;" não apresenta erro em tempo de compilação e, depois da mesma, podemos acessar o campo *estado* de c usando o operador ponto, tal como em "c.estado".
- d) Para que o código acima compile sem erros, exige-se que seja usada uma função para liberar a memória alocada dinamicamente.
- e) O comando "ESTADO \*e = malloc(sizeof(CIDADE));" está errado, e o correto seria "ESTADO e = malloc(sizeof(CIDADE));". Assim, temos erro de compilação nessa linha de código.

- 20) Alguns conceitos de orientação a objetos estão disponibilizados na Coluna I. Estabeleça a correta correspondência com seus significados disponibilizados na Coluna II.

Coluna I

1. Abstração
2. Classe
3. Polimorfismo
4. Herança
5. Encapsulamento

Coluna II

- ( ) Restringe o acesso a atributos e métodos.
- ( ) Define a estrutura e o comportamento de objetos de mesmo tipo.
- ( ) Permite ao usuário definir tipos de forma incremental, a partir de tipos existentes.
- ( ) Habilidade de concentrar nos aspectos essenciais, ignorando características menos importantes.
- ( ) Define que uma determinada operação pode se comportar de diferentes formas em diferentes classes.

A sequência correta, de cima para baixo, é:

- a) 5, 2, 4, 1, 3.      b) 2, 1, 3, 4, 5.      c) 5, 2, 4, 3, 1.      d) 4, 1, 3, 2, 5.      e) 5, 4, 2, 1, 3.

- 21) Assinale a alternativa correta.

- a) A linguagem Java permite que uma classe seja uma extensão direta de duas ou mais classes, ou seja, permite herança múltipla. ☒
- b) O modificador *abstract* pode ser aplicado a classes, atributos e métodos. ☒
- c) A linguagem Java não permite que uma classe implemente diversas interfaces. ☒
- d) O modificador *final* do Java pode ser aplicado a classes, atributos e métodos. ☒
- e) Os modificadores de acesso em Java são *public*, *protected*, *extern* e *private*. ☒

- 22) Considere o seguinte código em Java:

```
class Foo {
    int num;
    Baz comp = new Baz();
}

class Bar {
    boolean flag;
}

class Baz extends Foo {
    Bar thing = new Bar();
    double limit;
}
```

Quais das seguintes afirmações são verdadeiras a respeito dessas classes?

- I. Um objeto *Bar* é um objeto *Baz*.
- II. Um objeto *Bar* tem um objeto *Baz*. ☒
- III. Um objeto *Baz* é um objeto *Foo*.
- IV. Um objeto *Foo* é um objeto *Baz*.
- V. Um objeto *Baz* tem um objeto *Bar*. ☒

- a) I e II.      b) II e IV.      c) III e V.      d) I e IV.      e) Todas são verdadeiras.

- 23) Dada a seguinte definição de classe, em Java, quais das seguintes declarações de métodos são válidas se adicionadas à classe *Rid*?

```
public class Rid {
    public void aMethod(int i, String s) { ... }
}
```

- I. `public void aMethod(String s, int i) { ... }`
- II. `public int aMethod(int i, String s) { ... }`
- III. `public void aMethod(int i, String myString) { ... }`
- IV. `public void AMethod(int i, String s) { ... }`

- a) Somente II.      b) Somente III.      c) Somente II e III.      d) Somente I e IV.      e) I, II, III, IV.

24) Qual será o resultado apresentado pela execução do seguinte programa em Java?

```
class A {  
    public void f( ) { System.out.print("Inside A "); }  
}  
  
class B extends A {  
    public void f( ) { System.out.print("Inside B "); }  
}  
  
class C extends B {  
    public void f( ) { System.out.print("Inside C "); }  
}  
  
class D {  
    public static void main(String[ ] args) {  
        C c = new C( );  
        c.f( );  
        B b = c;  
        b.f( );  
        A a = (B) c;  
        a.f( );  
    }  
}
```

- a) Imprime "Inside C Inside B Inside A"  
b) Imprime "Inside C Inside C Inside B"  
c) Imprime "Inside C, Inside B Inside B"  
d) Imprime "Inside C Inside C Inside A"  
e) Nenhuma das anteriores.

25) O que será impresso ao executar o programa abaixo escrito em Java?

```
class Enigma{  
    public static void main(String args[]){  
        Integer x = new Integer(8);  
        Integer y = new Integer(5);  
        enigma(x, y);  
        System.out.println(x + " e " + y);  
    }  
  
    static void enigma(Integer a, Integer b)  
    { Integer aux = a; a = b; b = aux; }  
}
```

- a) 5 e 8  
b) 5 e 5  
c) 8 e 5  
d) 8 e 8  
e) A execução desse código resulta num erro.