

CADERNO DE QUESTÕES

**PROCESSO SELETIVO DE INGRESSO DE ALUNOS  
2º SEMESTRE DE 2019**

EDITAL Nº 012/2019/PRPGI

**CURSO DE PÓS-GRADUAÇÃO *LATO SENSU* EM  
DESENVOLVIMENTO WEB**



**NOME**

## INSTRUÇÕES AOS CANDIDATOS

Esta é uma prova única contendo questões que abordam conteúdos programáticos constantes no ANEXO VI do Edital nº 012/2019/PRPGI.

### Caderno de Questões

Este caderno contém 10 questões objetivas, com 5 alternativas cada uma.

Existe apenas UMA alternativa correta para cada questão da prova.

Leia, cuidadosamente, cada questão e marque a resposta desejada no campo correspondente à questão na folha de respostas.

### Folha de Respostas

A Folha de Respostas é pré-identificada. Confira os seus dados e, em caso de divergência, chame o Fiscal de Sala.

Depois da conferência dos seus dados, assine a folha no campo indicado.

A marcação das respostas só será aceita quanto feita com caneta esferográfica azul ou preta. NÃO use lápis.

É de responsabilidade do candidato o manuseio e marcação da Folha de Respostas, não sendo admitido amassá-la, molhá-la, dobrá-la, rasgá-la, manchá-la ou, de modo algum, danificá-la.

**Marcação Correta**



Serão consideradas marcações indevidas aquelas que estiverem rasuradas ou emendadas ou com campo de marcação não preenchido integralmente.

A Folha de Respostas é única e não será substituída.

## ATENÇÃO

O tempo de prova é de 4 horas. Ao final, a Folha de Respostas deve ser entregue ao Fiscal da Sala.

O candidato só poderá se retirar da sala após 1 hora do início da prova, e só poderá levar o caderno de questões após 1 hora e meia do início da prova.

### QUESTÃO 01

A função abaixo, escrita na linguagem C++, calcula o cosseno de um ângulo através da série abaixo, com uma precisão de 0,00001 (ou seja, serão adicionados à série somente os termos cujos valores absolutos sejam iguais ou superiores a este valor). A função recebe como parâmetro o valor do ângulo em radianos.

$$\text{cosseno}(\theta) = 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \dots$$

```
01 float cosseno(float a) {
02     int fat;
03     int n    = 0;
04     int sinal = 1;
05     float s   = 0.0;
06     float t   = 1;
07     while (t >= 0.00001) {
08         s = s + t * sinal;
09         n += 2;
10         fat = 1;
11         for (int i=1; i<=n;i++)
12             fat = fat * i;
13         t = pow(a, n) / fat;
14         sinal = sinal * (-1);
15     }
16     return s;
17 }
```

A partir da análise do código da função acima, assinale a alternativa correta:

- A) A função, quando executada, entrará em loop infinito, pois os termos gerados por ela são crescentes, enquanto que os termos da série são decrescentes;
- B) A função, quando executada, entrará em loop infinito, pois permite incluir termos menores que 0,00001;
- C) A função, quando executada, retornará 0,0, pois a condição para entrar no loop nunca será atendida;
- D) A função, quando executada, não acrescentará à série termos com valores negativos;
- E) A função, quando executada, retornará corretamente o cosseno do ângulo fornecido.

### QUESTÃO 02

A função abaixo, escrita na linguagem C++, reposiciona os elementos de um vetor em ordem decrescente, recebendo como parâmetro um vetor e o seu correspondente tamanho.

```
01 void ordena(int V[], int tam) {
02     int i, j, aux;
03     for (i=0; i < tam-1; i++) {
04         for (j=i+1; j < tam; j++) {
05             if (V[j] > V[i]) {
06                 aux = V[j];
07                 V[j] = V[i];
08                 V[i] = aux;
09             }
10         }
11     }
12 }
```

A partir da análise do código da função, assinale a alternativa correta.

- A) A condição `if (v[j] > v[i])` será verificada o mesmo número de vezes a cada execução da função, independentemente do tamanho do vetor `V`, que é sempre o mesmo a cada execução;
- B) A condição `if (v[j] > v[i])` será verificada o mesmo número de vezes a cada execução da função, independentemente do tamanho do vetor `V`, que pode variar a cada execução;
- C) O número de vezes que a condição `if (v[j] > v[i])` será verificada a cada execução da função depende do tamanho do vetor `V`, que pode variar a cada execução;
- D) A cada execução da função, a instrução `v[i] = aux` será executada o mesmo número de vezes que a condição `if (v[j] > v[i])` será verificada, independentemente do tamanho do vetor `V`;
- E) A instrução `v[i] = aux` será executada sempre o mesmo número de vezes a cada execução da função, independentemente dos valores dos elementos do vetor `V`.

### QUESTÃO 03

Considere o trecho de código na linguagem de programação C que se segue abaixo:

```
01 int i=0, j=-6, k=7, cont;  
02 for (cont= 4; cont<11; cont++) {  
03     i = cont + 7;  
04     while (k > 0) {  
05         k = k - 2;  
06         j = k + j/2;  
07     }  
08 }  
09 printf("i: %d j: %d k: %d ", i, j, k);
```

Assinale a alternativa que corresponda à saída gerada pelo código, quando executado:

- A) i: 1 j: 4 k: 1
- B) i: 15 j: 1 k: 0
- C) i: 10 j: 14 k: 2
- D) i: 17 j: 0 k: -1
- E) i: 3 j: 1 k: 4

### QUESTÃO 04

Considere o trecho de código na linguagem de programação C que se segue abaixo:

```
01 #include <stdio.h>  
02  
03 int main() {  
04     int a = 11, b = 11;  
05     while (a >= 0) {  
06         int b = 11;  
07         a--;  
08         b--;  
09     }  
10  
11     do {  
12         a--;  
13         if (b % 2 == 1) {
```

```
14         a++;
15         b++;
16     }
17 } while (a > 0);
18
19 printf("%d %d", a, b);
20
21 return 0;
22 }
```

Assinale a alternativa que corresponda, respectivamente, aos valores finais das variáveis A e B, quando da execução do programa:

- A) -1 e -11;
- B) -1 e 12;
- C) -2 e 12;
- D) 1 e 12;
- E) -2 e 11.

### QUESTÃO 05

Considere as implementações, na linguagem Java, das 4 classes que pertencem a um mesmo pacote, conforme se segue abaixo:

#### C1.java

```
01 public abstract class C1 {
02     abstract public Object criar();
03
04     public void mostrar() {
05         System.out.print("Pós Web 2019");
06     }
07 }
```

#### C2.java

```
01 public class C2 extends C1 {
02     static int i = 0;
03     int j;
04
05     public Object criar() {
06         i += 2;
07         j = i;
08         return j;
09     }
10
11     public void mostrar() {
12         System.out.print("j = " + j);
13     }
14 }
```

#### C3.java

```
01 public class C3 extends C1 {
02     double d = 3.14;
03     float f;
04
05     public Object criar() {
```

### C3.java (continuação)

```
06         d = d + 2.0;
07         f = (float)d;
08         return f;
09     }
10
11     public void mostrar() {
12         System.out.print("f = " + f);
13     }
14 }
```

### PosWeb2019.java

```
01 public class PosWeb2019 {
02
03     public static void main(String[] args) {
04         C1 a, b, c;
05         Object o1, o2, o3;
06         a = new C2();
07         b = new C2();
08         c = new C3();
09         o1 = a.criar();
10         o1 = a.criar();
11         o2 = b.criar();
12         o3 = c.criar();
13         o3 = c.criar();
14         a.mostrar();
15         System.out.print(" ");
16         b.mostrar();
17         System.out.print(" ");
18         c.mostrar();
19         System.out.print(" " + o1);
20         System.out.print(" " + o2);
21         System.out.print(" " + o3);
22     }
23
24 }
```

Assinale a alternativa que apresenta, corretamente, os valores impressos a partir da execução da função estática principal da classe `PosWeb2019`:

- A) Nenhuma, pois a sintaxe empregada na implementação das classes apresenta erros;
- B) `j = 4 j = 6 f = 7.14 4 6 7.14`
- C) `j = 4 j = 5 f = 7.14 4 5 7.14`
- D) `Pós Web 2019 Pós Web 2019 Pós Web 2019 4 6 7.14`
- E) `Pós Web 2019 Pós Web 2019 Pós Web 2019 4 5 7.14`

### QUESTÃO 06

Considere uma classe implementada na linguagem Java, denominada de `Bicicleta`, e que, dentro dela, exista, entre outros, o trecho de código que se segue abaixo:

```
01 public Bicicleta(int inicioCadencia, int inicioVeloc, int inicioMarcha) {
02     marcha = inicioMarcha;
03     cadencia = inicioCadencia;
04     velocidade = inicioVeloc;
05 }
```

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

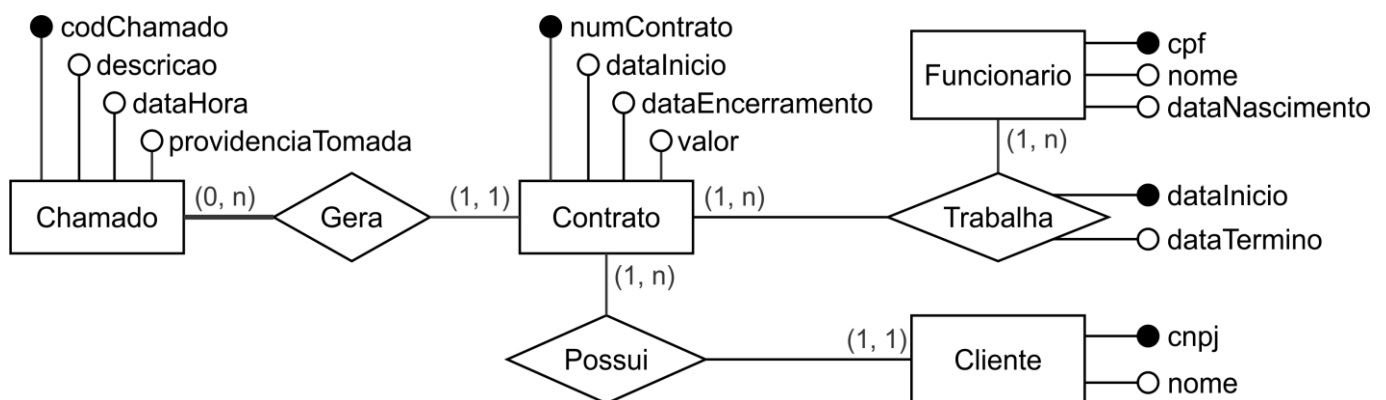
- I. O trecho de código mostrado é um exemplo de construtor;
- II. Para criar um novo objeto `Bicicleta`, utiliza-se o operador `new`, como, por exemplo:  
`Bicicleta minhaBike = new Bicicleta(25, 0, 7);`
- III. O trecho mostrado não é um método;
- IV. Para evitar conflitos, deve haver apenas um construtor por classe.

Assinale a alternativa que corresponda às assertivas 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.

### QUESTÃO 07

O Diagrama Entidade Relacionamento a seguir apresenta três relacionamentos: o primeiro é "Possui", que representa os clientes que possuem contratos; o segundo relacionamento é "Trabalha", que representa as associações dos funcionários aos contratos em que trabalham e o terceiro é o relacionamento "Gera", que representa associações entre contratos e chamados.

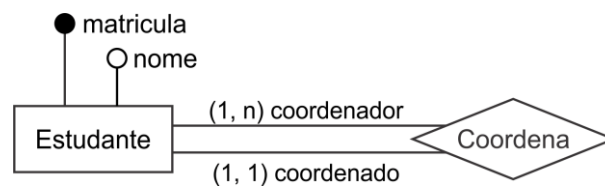


Assinale a alternativa que contém informações que não podem ser recuperadas com base no diagrama acima:

- A) O cliente pode possuir um ou vários contratos;
- B) O chamado é controlado exclusivamente por um contrato;
- C) O funcionário que resolveu um determinado chamado;
- D) O funcionário pode trabalhar em um ou vários contratos;
- E) Os chamados de um determinado cliente.

### QUESTÃO 08

Um autorrelacionamento representa uma associação entre ocorrências de uma mesma entidade. No Diagrama Entidade Relacionamento a seguir a associação entre a entidade **Estudante** e o relacionamento **Coordena** é um exemplo de autorrelacionamento.



Sabendo disto, assinale a alternativa que representa CORRETAMENTE o mapeamento do diagrama para o esquema relacional (em cada alternativa, os atributos sublinhados compõem as chaves primárias e os atributos com “referencia” indicam as chaves estrangeiras):

- A) Estudante(matricula, nome)  
Coordena(matricula, matricula-coordenador)  
matricula-coordenador referencia Estudante
- B) Estudante(matricula, nome)  
Coordena(matricula, matricula-coordenador)  
matricula-coordenador referencia Estudante
- C) Estudante(matricula, nome, matricula-coordenador)  
matricula-coordenador referencia Estudante
- D) Estudante(matricula, nome, matricula-coordenador)  
matricula-coordenador referencia Estudante
- E) Estudante(matricula, nome, matricula-coordenador, matricula-coordenado)  
matricula-coordenador e matricula-coordenado referencia Estudante

Responda as questões 09 e 10 com base nas relações apresentadas a seguir.

Empregado		
<u>codEmpregado</u>	nome	cargaHorariaSemanal
1111	João	40
2222	Pedro	20
3333	Marta	40
4444	Júlia	20
5555	Patrick	40
6666	Adriana	20
7777	Caio	20

Projeto			
<u>codProjeto</u>	Titulo	cancelado	codEmpregado
1	Projeto A	não	1111
2	Projeto B	não	2222
3	Projeto C	não	1111
4	Projeto D	não	3333
5	Projeto E	não	4444
6	Projeto F	sim	1111
7	Projeto G	sim	2222
8	Projeto H	sim	5555
9	Projeto I	sim	Null

### QUESTÃO 09

Considere a seguinte consulta SQL:

```

SELECT nome
FROM Empregado
WHERE codEmpregado NOT IN (
    SELECT DISTINCT codEmpregado
    FROM Projeto
    WHERE codEmpregado IS NOT NULL
);
  
```



Assinale a alternativa que representa uma consulta SQL através da qual se obtém o mesmo resultado obtido pela consulta SQL descrita acima:

- A) `SELECT DISTINCT a.nome  
FROM Empregado a  
INNER JOIN Projeto b ON a.codEmpregado = b.codEmpregado;`
- B) `SELECT a.nome  
FROM Projeto b  
RIGHT JOIN Empregado a ON a.codEmpregado = b.codEmpregado  
WHERE b.codProjeto IS NULL;`
- C) `SELECT a.nome  
FROM Empregado a  
RIGHT JOIN Projeto b ON a.codEmpregado = b.codEmpregado  
WHERE a.nome IS NULL;`
- D) `SELECT DISTINCT a.nome  
FROM Empregado a  
LEFT JOIN Projeto b ON a.codEmpregado = b.codEmpregado  
WHERE a.nome IS NULL;`
- E) `SELECT a.nome  
FROM Projeto b  
LEFT JOIN Empregado a ON a.codEmpregado = b.codEmpregado  
WHERE b.codProjeto IS NULL;`

### QUESTÃO 10

Considere a seguinte consulta SQL:

```
SELECT codEmpregado, MAX(codProjeto) AS Maximo
FROM Projeto
WHERE codEmpregado IS NOT NULL
GROUP BY codEmpregado
ORDER BY Maximo;
```

Assinale a alternativa que corresponde ao resultado obtido com a consulta:

A) 

codEmpregado	Maximo
1111	1, 3, 6
2222	2, 7
3333	4
4444	5
5555	8

B) 

codEmpregado	Maximo
3333	4
4444	5
1111	6
2222	7
5555	8

C) 

codEmpregado	Maximo
3333	4
1111	6

D) 

codEmpregado	Maximo
1111	1, 3, 6
2222	2, 7

E) 

codEmpregado	Maximo
5555	8