



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina: Projeto e Desenvolvimento de Algoritmos**  
**AP2 1º semestre de 2016**

Nome –

Assinatura –

---

Observações:

- A) Prova sem consulta e sem uso de máquina de calcular.
- B) Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
- C) Você pode usar lápis para responder as questões.
- D) Ao final da prova devolva as folhas de questões e as de respostas.
- E) **Essa prova não contém "pegadinhas", mas os professores que a elaboraram algumas vezes, por distração, cometem pequenos erros no enunciado ou nas alternativas de respostas. Assim, se você achar à primeira vista que uma alternativa está correta, esta provavelmente é a resposta da questão. Não fique procurando por espaços em branco ou quebras de linha sobrando ou faltando e não acredite que, por exemplo, um 15 como resposta quando você esperava um 15.0 é motivo para marcar "Nenhuma das respostas anteriores" como resposta da questão.**
- F) **Todas as respostas devem ser transcritas no local apropriado, no cartão de respostas a seguir.**
- G) Boa Prova!

---

Questão					
1	A	<input checked="" type="checkbox"/>	C	D	E
2	A	B	C	<input checked="" type="checkbox"/>	E
3	<input checked="" type="checkbox"/>	B	C	D	E
4	A	B	C	<input checked="" type="checkbox"/>	E
5	A	B	<input checked="" type="checkbox"/>	D	E
6	A	B	<input checked="" type="checkbox"/>	D	E
7	A	B	C	<input checked="" type="checkbox"/>	E
8	A	B	C	<input checked="" type="checkbox"/>	E
9	<input checked="" type="checkbox"/>	B	C	D	E
10	<input checked="" type="checkbox"/>	B	C	D	E

Para a análise dos algoritmos nesta prova, considere a existência de uma forma alternativa de inicializar variáveis indexadas em PETEQS:

$v \leftarrow [1, 2, 3]$

que é equivalente a:

$v[1] \leftarrow 1$

$v[2] \leftarrow 2$

$v[3] \leftarrow 3$

## 1ª questão (valor 1.0)

O algoritmo a seguir calcula o seno de um número real  $x$  através de uma aproximação dada pela série:

$$\text{sen}(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \frac{x^9}{9!} - \dots$$

Preencha o espaço assinalado de modo a completar o algoritmo.

**início**

leia  $x$

leia  $n$

soma  $\leftarrow x$

sinal  $\leftarrow -1$

**para**  $i \leftarrow 1$  **até**  $n$  **faça**

termo  $\leftarrow 1.0$

**para**  $j \leftarrow 1$  **até**  $(2*i - 1)$  **faça**

termo  $\leftarrow$  *(complete essa expressão)*

**próximo**  $j$

soma  $\leftarrow$  soma + sinal \* termo

sinal  $\leftarrow$  -sinal

**próximo**  $i$

**imprima** soma

**fim**

A)  $x / j$

B) termo \*  $x / j$

C)  $- x / j$

D)  $- \text{termo} * x / (j * (j - 1))$

E) Nenhuma das respostas anteriores

## 2ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Assuma que o comando **imprima** não causa uma mudança de linha após a impressão.

```

início
  x ← 2
  v ← 15 / 2 * x
  imprima v MOD 10 + 1
  x ← 15 - 25 / 2
  y ← 2.4 / x
  imprima y
  b ← 12 / 2 + 1
  z ← b / 2.0 + b MOD 2
  imprima z
  w ← 7.5
  se ((w * 4.0) / 10.0 < 3) então
    u ← w + 1
  senão
    u ← w + 1.5
  fim se
  imprima u
fim

```

- A) 6   0.8   4.0   8.5  
 B) 2.5   0.48   2.0   9.0  
 C) 4   -0.48   4.0   8.5  
 D) 5   0.8   4.5   9.0  
 E) Nenhuma das respostas anteriores

### 3ª questão (valor 1.0)

O algoritmo a seguir calcula o perímetro de um triângulo dado. Determine o que será impresso pelo algoritmo se as entradas fornecidas ao mesmo forem 4, 1, 2, 3 e 4, nessa ordem. Assuma que o comando **imprima** não causa uma mudança de linha após a impressão.

```

função testaTriangulo(entradas:a, b, c)
início
  se (a + b) > c E (a + c) > b E (b + c) > a então
    resultado ← a + b + c
  senão
    resultado ← -1
  fim se
fim

início
  v1 ← [1, 1, 7, 10, 1, 1]
  v2 ← [2, 2, 8, 10, 2, 1]
  v3 ← [3, 5, 9, 10, 4, 1]
  leia n
  para i ← 1 até n faça
    leia k
    perimetro ← testaTriangulo(v1[k], v2[k], v3[k])
    se perimetro > 0 então
      imprima perimetro
    senão
      imprima 'inválido'
    fim se
  próximo i
fim

```

- A) inválido inválido 24 30
- B) 30 24 inválido 7
- C) 6 8 24 30
- D) 6 8 inválido inválido
- E) Nenhuma das respostas anteriores

#### 4ª questão (valor 1.0)

O algoritmo a seguir conta o número de subarrays (de tamanho maior que um) cujos elementos aparecem em ordem estritamente crescente. O que será impresso pelo algoritmo se as entradas fornecidas ao mesmo forem 5, -2, -2, 6, 9 e 9, nesta ordem?

```
função contaSubArrays(entradas: n , array[])
início
    conta ← 0
    tam ← 1
    para i ← 1 até n faça
        se i < n - 1 E array[i + 1] > array[i] então
            tam ← tam + 1
        senão
            conta ← conta + (tam * (tam - 1)) / 2
            tam ← 1
    fim se
    próximo i
    resultado ← conta
fim
```

```
início
    leia n
    para i ← 1 até n faça
        leia array[i]
    próximo i
    imprima contaSubArrays(n , array)
fim
```

- A) 0
- B) 1
- C) 2
- D) 3
- E) Nenhuma das respostas anteriores

#### 5ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir se a sequência de chamadas à função **numFilhos()** gerar como resultados 1, 2, 0, 3, 0, 1, 0 e 0, nesta ordem?

```

início
    individuosNessaGeracao ← 1
    numGeracoes ← 0
    enquanto individuosNessaGeracao > 0 faça
        numGeracoes ← numGeracoes + 1
        individuosNaProximaGeracao ← 0
        para i ← 1 até individuosNessaGeracao faça
            individuosNaProximaGeracao ← individuosNaProximaGeracao + numFilhos()
        próximo i
        individuosNessaGeracao ← individuosNaProximaGeracao
    fim enquanto
    imprima numGeracoes
fim

```

- A) 3
- B) 4
- C) 5
- D) 6
- E) Nenhuma das respostas anteriores

## 6ª questão (valor 1.0)

Observe o algoritmo a seguir.

```

função f ( entradas: q, t)
início
    s ← 0
    para i ← 0 até q faça
        se (i mod 2) = t então
            s ← s + i
        fim se
    próximo i
    resultado ← s
fim
início
    n ← 7
    s0 ← f(n, 0)
    s1 ← f(n, 1)
    imprima s0, ' - ', s1
fim

```

O valor impresso pelo algoritmo será:

- A) 0 - 7
- B) 0 - 1
- C) 12 - 16
- D) 16 - 12
- E) Nenhuma das respostas anteriores

## 7ª questão (valor 1.0)

Observe o algoritmo a seguir:

```

início
    v[1] ← 3
    v[2] ← 9
    v[3] ← 2
    v[4] ← 7
    para i ← 1 até 3 faça
        se (v[i] > v[i+1]) então
            temp ← v[i]
            v[i] ← v[i+1]
            v[i+1] ← temp
        fim se
    próximo i
    para i ← 1 até 4 faça
        imprima v[i], ' '
    próximo i
fim

```

Considerando que o comando **imprima** não muda de linha ao final de uma impressão, qual é a saída impressa pelo algoritmo?

- A) 2 3 7 9
- B) 9 7 3 2
- C) 3 9 7 2
- D) 3 2 7 9
- E) Nenhuma das respostas anteriores

## 8ª questão (valor 1.0)

Observe o algoritmo a seguir:

```

início
    a ← 1900.0
    b ← 2800.0
    tb ← 0.075
    tc ← 0.15
    leia s
    se (s < a) então
        imp ← 0.0
    senão
        se (s < b) então
            imp ← s * tb
        senão
            imp ← s * tc
        fim se
    fim se
    v ← s - imp
    imprima v
fim

```

Se o dado de entrada fornecido pelo usuário for 3000.0, a saída impressa pelo algoritmo será:

- A) 0.0
- B) 1850.0
- C) 3000.0
- D) 2550.0
- E) Nenhuma das respostas anteriores

## 9ª questão (valor 1.0)

Observe o algoritmo a seguir:

```
procedimento arruma(entradas: t, saídas: n[])
```

```
início
```

```
    para i ← 1 até t faça
```

```
        se (n[i] > 10.0) então
```

```
            n[i] ← 10.0
```

```
        senão
```

```
            se (n[i] < 0.0) então
```

```
                n[i] ← 0.0
```

```
            fim se
```

```
        fim se
```

```
    próximo i
```

```
fim
```

```
procedimento print(entradas: t, n[])
```

```
início
```

```
    para i ← 1 até t faça
```

```
        imprima n[i], ' '
```

```
    próximo i
```

```
fim
```

```
início
```

```
    n[1] ← 3.5
```

```
    n[2] ← 11.0
```

```
    n[3] ← 9.0
```

```
    n[4] ← -9.0
```

```
    n[5] ← 5.0
```

```
    arruma(5, n)
```

```
    print(5, n)
```

```
fim
```

Considerando que o comando `imprima` não muda de linha ao final da impressão, a saída impressa pelo algoritmo será:

- A) 3.5 10.0 9.0 0.0 5.0
- B) 3.5 11.0 9.0 -9.0 5.0
- C) 3.5 0.0 9.0 10.0 5.0
- D) 0.0 10.0 10.0 0.0 0.0
- E) Nenhuma das respostas anteriores

## 10ª questão (valor 1.0)

Considere que, em PETEQS, existe uma tabela do tipo ASCII que atribui um número inteiro para cada caractere. Considere ainda que estão disponíveis as seguintes funções:

```
ord(car)
```

retorna a posição do caractere car na tabela de caracteres do computador

```
LeCadeia(frase)
```

lê um conjunto de caracteres do teclado e os armazena no vetor frase

```
CompCadeia(frase)
```

retorna quantos caracteres estão armazenados no vetor frase

Os caracteres alfabéticos ocupam posições contíguas na tabela, isto é,

$\text{ord}('B') - \text{ord}('A') = 1$

Usando estas funções, um aluno de PDA escreveu o seguinte algoritmo:

```
início
  LeCadeia(frase)
  c ← 0
  para i ← 1 até CompCadeia(frase) faça
    se (ord(frase[i]) ≥ ord('a')) E (ord(frase[i]) ≤ ord('e')) então
      c ← c + 1
    fim
  próximo i
  imprima c
fim
```

Marque a opção que mostra o que será impresso pelo algoritmo caso seja digitado o seguinte conjunto de caracteres:

**abcdef**

- A) 5
- B) 4
- C) 3
- D) 2
- E) Nenhuma das respostas anteriores