



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância
Curso de Tecnologia em Sistemas de Computação
Disciplina: Projeto e Desenvolvimento de Algoritmos
AP3 2º semestre de 2019

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

Para a análise dos algoritmos nessa prova, considere a existência da função **tamanho()**, cuja documentação é mostrada a seguir:

função tamanho(entradas: vetor[])

Retorna o número de elementos no vetor passado como parâmetro. Retorna zero se o vetor está vazio.

Exemplos:

```
v[1] ← 1
v[2] ← 2
imprima tamanho(v)      # imprimiria 2
```

Considere ainda a existência de uma forma alternativa de inicializar variáveis indexadas em PETEQS:

```
v ← [1, 2, 3]
```

que é equivalente a:

```
v[1] ← 1
v[2] ← 2
v[3] ← 3
```

1ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir se as entradas fornecidas pelo usuário forem, nesta ordem, 4, 235, 100, 134, 0, 0, 200, 200, 150 e 150?

```
início
  leia N
  leia min
  conta ← 0
  para i ← 1 até N faça
    leia nota1
    leia nota2
    se nota1 + nota2 >= min então
      conta ← conta + 1
    fim se
  próximo i
  imprima conta
fim
```

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

2ª questão (valor 1.0)

Na solução dessa questão considere a existência em PETEQS das funções **tamanho(entradas: str)** que retorna o número de caracteres na string **str**,

charAt(entradas: palavra, pos) que retorna o caractere na posição **pos** da string **palavra**, e **concat(entradas: str1, str2)** que retorna a concatenação das strings **str1** e **str2**.

Exemplo: **concat("a", "b") = "ab"**

O que será impresso pelo algoritmo a seguir?

início

```
a ← "0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ-"
b ← "012345678922233344455566677778889999-"
tel ← "0800-FALE-SBC"
saida ← ""
para i ← 1 até tamanho(tel) faça
  para j ← 1 até tamanho(a) faça
    se charAt(a, j) = charAt(tel, i) então
      saida ← concat(saida, charAt(b, j))
    fim se
  próximo j
próximo i
imprima saida
fim
```

- A) 0800-FALE-SBC
- B) 55-98-234-5678
- C) 617-88-74
- D) 0800-3253-722
- E) Nenhuma das respostas anteriores

3ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir se as entradas fornecidas pelo usuário forem, nesta ordem, 4, 2, 5, 3, 4, 2, 1, 4, 5?

```
início
  maior ← 0
  quem ← 0
  leia N
  para i ← 1 até N faça
    leia b
    leia e
    prod ← 1
    para j ← 1 até e faça
      prod ← prod * b
    próximo j
    se prod > maior então
      maior ← prod
      quem ← i
    fim se
  próximo i
  imprima quem
fim
```

- A) 1
- B) 2
- C) 3
- D) 4
- E) Nenhuma das respostas anteriores

4ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir se as entradas fornecidas pelo usuário forem, nesta ordem, 10, 2, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2?

```

início
  leia N
  leia K
  para i ← 1 até N faça
    leia num
    V[num] ← V[num] + 1
  próximo i
  min ← V[1]
  para i ← 2 até K faça
    se V[i] < min então
      min ← V[i]
    fim se
  próximo i
  imprima min
fim

```

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

5ª questão (valor 1.0)

O algoritmo a seguir foi executado duas vezes consecutivas, primeiro com as entradas 10, 5, +, 5, e então com as entradas 44, 23, *, 2. Nessa ocasião, as saídas impressas pelo algoritmo foram, na ordem de execução:

- A) OK OK
- B) OK OVERFLOW
- C) OVERFLOW OK
- D) OVERFLOW OVERFLOW
- E) Nenhuma das respostas anteriores

6ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Considere que o comando imprima não muda de linha ao final de sua execução.

```

função gera(entradas: a, b)
início
  para i ← 0 até a - 1 faça
    para j ← 0 até b - 1 faça
      imprima i + j
    próximo j
  próximo i
fim

início
  gera(3, 4)
fim

```

- A) 012312342345
- B) 012301230123
- C) 012345678901
- D) 321043215432
- E) Nenhuma das respostas anteriores

7ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Considere que o comando imprima não muda de linha ao final de sua execução.

```

função caracter(entradas: a, b)
  i ← a
  enquanto i > 0 faça
    j ← b
    enquanto j > 0 faça
      se (i * j MOD 2) = 1 então
        imprima '*'
      senão
        imprima '#'
      fim se
    j ← j - 1
  fim enquanto
  i ← i - 1
fim enquanto
fim

início
  caracter(2, 3)
fim

```

- A) #####
- B) #####*
- C) #####
- D) #####
- E) Nenhuma das respostas anteriores

8ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

```

função calcula(entradas: ini, j[])
início
  para i ← 1 até tamanho(j) faça
    ini ← ini * (1 + j[i])
  próximo i
  resultado ← ini
fim

início
  j ← [0.1, 0.25, -0.1]
  ini ← 1.0
  fim ← calcula(ini, j)
  r ← (fim - ini)/ini
  imprima r
fim

```

- A) 0.25
- B) 0.2375
- C) 0.1
- D) -0.25
- E) Nenhuma das respostas anteriores

9ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Considere que o comando imprima não muda de linha ao final de sua execução.

```

função pr (entradas: v[])
  para i ← 1 até tamanho(v) faça
    imprima v[i], ' '
  próximo i
fim

função mix(entradas: x1[], x2[], x3[])
início
  i ← 1
  j ← 1
  k ← 1
  enquanto i < tamanho(x1) E
    j < tamanho(x2) faça
    se (x1[i] > x2[j]) então
      x3[k] ← x1[i]
      i ← i + 1
    senão
      x3[k] ← x2[j]
      j ← j + 1
    fim se
    k ← k + 1
  fim enquanto
  enquanto (i < tamanho(x1)) faça
    x3[k] ← x1[i]
    i ← i + 1
    k ← k + 1
  fim enquanto
  enquanto (j < tamanho(x2)) faça
    x3[k] ← x2[j]
    j ← j + 1
    k ← k + 1
  fim enquanto
  resultado ← x3
fim

início
  v1 ← [11, 10, 7, 6]
  v2 ← [13, 12, 9, 3]
  v3 ← mix(v1, v2, v3)
  pr(v3)
fim

```

- A) 14 13 12 11 10 8 7 4
 B) 11 13 10 12 7 9 3 6
 C) 13 12 9 3 11 10 7 6
 D) 13 12 11 10 9 7 6 3
 E) Nenhuma das respostas anteriores

10ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

```

função f1(entradas: n)
  r ← 1
  para i ← 1 até n faça
    r ← r * i
  próximo i
  resultado ← r
fim

função f2(entradas: a, b)
início
  resultado ← f1(a) / (f1(b) - f1(a-b))
fim

início
  imprima f2(5, 3)
fim

```

- A) 10
 B) 8
 C) 6
 D) 4
 E) Nenhuma das respostas anteriores