



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina: Projeto e Desenvolvimento de Algoritmos**  
**AP2 2º semestre de 2017**

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

Para resolver algumas das questões abaixo considere que, em PETEQS, existe uma tabela do tipo ASCII que atribui um número inteiro para cada caractere. Considere também que, da mesma forma que na tabela ASCII, as letras do alfabeto receberam números inteiros consecutivos e em ordem crescente. Considere ainda a existência das funções `ordem()`, `tamanho()`, e `charAt()` e `concat()` cuja documentação é mostrada a seguir:

**função `ordem(entradas: car)`**

Retorna a posição do caractere `car` na tabela de caracteres do computador

Exemplo:

```
imprima ordem('C') - ordem('A')  
# imprimiria 2
```

**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
```

**função `tamanho(entradas: str)`**

Retorna o número de caracteres na string `str` passada como parâmetro.

Exemplos:

```
imprima tamanho('Dilma')  
# imprimiria 5
```

**função `charAt(entradas: str, pos)`**

Retorna uma string contendo o caractere na posição `pos` da cadeia de caracteres `str` passada como parâmetro.

Exemplo:

```
imprima charAt('CEDERJ', 3)  
# imprimiria 'D'
```

**função `concat(entradas: str1, str2)`**

Retorna uma cadeia de caracteres formada pela concatenação de `str1` e `str2`.

Exemplo:

```
imprima concat('Alo ', 'mundo! ')  
# imprimiria 'Alo mundo!'
```

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? Suponha que as entradas fornecidas pelo usuário ao algoritmo foram, nessa ordem, 10, 1, 10, 2, 10, 3 e 0.

**início**

```
total ← 0  
voucher ← 0  
quantidade ← 1  
enquanto quantidade > 0 faça  
  leia quantidade  
  se quantidade > 0 então  
    leia valor  
    se voucher < valor*quantidade então  
      total ← total - voucher +  
        quantidade*valor  
    fim se  
    voucher ← 0.1 * quantidade * valor  
  fim se  
fim enquanto  
imprima total
```

**fim**

- A) 34
- B) 52
- C) 57
- D) 60
- E) Nenhuma das respostas anteriores

## 2ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

**função `substitui(entradas: str)`**

**início**

```
resultado ← ''  
para i ← 1 até tamanho(str) faça  
  ch ← charAt(str, i)  
  se ordem(ch) >= ordem('A') E  
    ordem(ch) <= ordem('Z') então  
    resultado ← concat(resultado,  
      concat(ch, concat('.', ' ')))  
  fim se  
próximo i  
fim
```

**início**

```
imprima substitui('LucasMacielPereiraLima')  
fim
```

- A) LucasMacielPereiraLima
- B) Lucas M. P. Lima
- C) LMPL
- D) L. M. P. L.
- E) Nenhuma das respostas anteriores

## 3ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

#### variáveis públicas

```
notasMinimas[], conceitos[]
```

**função** classifica(**entradas:** nota)

**início**

```
resultado ← 'F'
```

```
para i ← 1 até tamanho(notasMinimas) faça
```

```
se nota >= notasMinimas[i] então
```

```
resultado ← conceitos[i]
```

```
fim se
```

```
próximo i
```

**fim**

**início**

```
notasMinimas ← [52, 70, 73, 76, 80, 84, 87,  
89, 91, 98]
```

```
conceitos ← ['D', 'C-', 'C', 'C+', 'B-',  
'B', 'B+', 'A-', 'A', 'A+']
```

```
imprima classifica(81)
```

**fim**

A) F

B) C+

C) B-

D) B+

E) Nenhuma das respostas anteriores

#### 4ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

**função** misterio(**entradas:** lista[])

**início**

```
resultado ← 0
```

```
para i ← 1 até tamanho(lista) faça
```

```
aux[tamanho(lista) - i + 1] = lista[i]
```

```
próximo i
```

```
se lista = aux então
```

```
resultado ← 1
```

```
fim se
```

**fim**

**início**

```
imprima misterio(['A', 'N', 'I', 'L', 'I',  
'N', 'A'])
```

```
imprima misterio(['R', 'E', 'V', 'I', 'V',  
'E', 'R'])
```

**fim**

A) 1 1

B) 1 0

C) 0 1

D) 0 0

E) Nenhuma das respostas anteriores

#### 5ª questão (valor 1.0)

Observe o algoritmo a seguir:

**início**

```
DAT ← ['Ana', 'Edu', 'Carolina', 'David']
```

```
P ← 4
```

```
enquanto P > 2 faça
```

```
DAT[P] ← DAT[P - 1]
```

```
P ← P - 1
```

```
fim enquanto
```

**fim**

Após a execução do algoritmo, qual elemento de **DAT** conterá a string 'Carolina'?

A) **DAT**[1]

B) **DAT**[2]

C) **DAT**[3]

D) **DAT**[4]

E) Nenhuma das respostas anteriores

#### 6ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Em sua resposta, considere que o comando **imprima** não muda de linha após a impressão.

**início**

```
i ← 10
```

```
a ← verdadeiro
```

```
enquanto a = verdadeiro faça
```

```
se (i mod 3) = 0 então
```

```
imprima ' casa'
```

```
i ← i - 1
```

```
fim se
```

```
se i = 10 então
```

```
imprima 'Quem'
```

```
i ← i - 1
```

```
fim se
```

```
se i = 8 então
```

```
imprima ' quer'
```

```
i ← i - 2
```

```
fim se
```

```
se i = 5 então
```

```
i ← i - 1
```

```
imprima ' longe de'
```

```
i ← i - 1
```

```
fim se
```

```
se i = 2 então
```

```
imprima '!' 
```

```
a ← falso
```

```
fim se
```

```
fim enquanto
```

**fim**

A) Quem casa quer casa longe de casa!

B) Quem quer casa longe de casa casa!

C) Quem casa longe de casa quer casa!

D) Quem longe de casa casa quer casa!

E) Nenhuma das respostas anteriores

#### 7ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Em sua resposta, considere que o comando **imprima** não muda de linha após a impressão.

```

procedimento imp(entradas: tam, v[])
início
    para i ← 1 até tam faça
        imprima v[i], ' '
    próximo i
fim

```

```

procedimento troca(saídas: x, y)
início
    temp ← x
    x ← y
    y ← temp
fim

```

```

início
    v[1] ← 10
    v[2] ← 12
    v[3] ← 5
    para i ← 1 até 2 faça
        min ← i
        para j ← i+1 até 3 faça
            se v[j] < v[min] então
                min ← j
            fim se
        próximo j
        se i <> min então
            troca(v[i], v[min])
        fim se
    próximo i
    imp(3, v)
fim

```

- A) 10 12 5
- B) 12 10 5
- C) 12 10 5
- D) 5 10 12
- E) Nenhuma das respostas anteriores

### 8ª questão (valor 1.0)

Observe o algoritmo a seguir:

```

início
    leia items
    total ← 0.0
    para i ← 1 até items faça
        leia quant, preco
        total ← total + quant * preco
    próximo i
    imprima total
fim

```

Considere que os valores fornecidos pelo usuário ao algoritmo são, nessa ordem:

```

2
3 5.5
2 7.5

```

A saída impressa pelo algoritmo será:

- A) 30.0
- B) 15.3
- C) 20.0
- D) 31.5
- E) Nenhuma das respostas anteriores

### 9ª questão (valor 1.0)

Determine a saída do algoritmo a seguir:

```

procedimento valor(entradas: x, saídas: y)
início
    x ← x + 1
    y ← y - 1
fim

```

```

início
    a ← -1
    b ← 1
    valor(a, b)
    imprima a, b
fim

```

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

### 10ª questão (valor 1.0)

Determine a saída do algoritmo a seguir.

```

início
    tS ← 556
    h ← tS / 3600
    m ← (tS mod 3600) / 60
    s ← (tS mod 3600) mod 60
    imprima h, ' ', m, ' ', s
fim

```

- A) 5 55 56
- B) 0 55 6
- C) 0 9 56
- D) 0 9 16
- E) Nenhuma das respostas anteriores