



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina: Projeto e Desenvolvimento de Algoritmos**  
**AP3 1º semestre de 2008.**

**Nome –**

**Assinatura –**

---

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
  2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
  3. Você pode usar lápis para responder as questões.
  4. Ao final da prova devolva as folhas de questões e as de respostas.
  5. Todas as respostas devem ser transcritas no local apropriado, ao final do caderno de questões.
- 

**1ª questão (valor 1.0)**

Considere que os valores mostrados abaixo foram fornecidos na ordem indicada ao algoritmo a seguir:

2 4 3 2  
3  
1 1  
2 2  
3 3  
0 0 0 0

**início**

```
    leia tix, tiy, tfx, tfy
    enquanto (tix<>0) e (tiy<>0) e (tfx<>0) e (tfy<>0) faça
        dentro ← 0
        leia meteoros
        para i ← 1 até meteoros faça
            leia mx, my
            se (tix<=mx)e(mx<=tfx)e(tfy<=my)e(my<=tiy) então
                dentro ← dentro + 1
            fim se
        próximo i
        imprime dentro
        leia tix, tiy, tfx, tfy
    fim enquanto
```

**fim**

O valor impresso pelo algoritmo é:

- A) 0
- B) 1
- C) 2
- D) 3
- E) 4

## 2ª questão (valor 1.0)

Considere que os valores mostrados abaixo foram fornecidos na ordem indicada ao algoritmo abaixo.

```
4
0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1
1 1 0 1 1 1 0 1
1 0 1 0 1 0 1 0
```

```
início
  leia casos
  para i ← 1 até casos faça
    m ← 0
    s ← 0
    para j ← 1 até 8 faça
      leia bit
      se bit = 1 então
        s ← s + 1
      senão
        se m < s então
          m ← s
        fim se
      s ← 0
    fim se
    próximo j
    se m < s então
      m ← s
    fim se
    imprime m
  próximo i
fim
```

Considere que o comando **imprime** não passa para a linha seguinte ao terminar de imprimir. O valor impresso pelo algoritmo é:

- A) 0 7 3 4
- B) 0 8 3 1
- C) 0 8 6 4
- D) 0 4 3 2
- E) 8 0 1 1

### 3ª questão (valor 1.0)

Considere que os valores mostrados abaixo foram fornecidos na ordem indicada ao algoritmo a seguir:

1 2 3 4 5 6 7 8 9  
1 2 3 4 5 6 7 8 11

```
função leiaV (saídas: v[])
início
    i ← 1
    leia v[i]
    enquanto (i < 9) e (v[i] > 0) faça
        i ← i + 1
        leia v[i]
    fim enquanto
    v[i+1] ← -1
fim

função tam (entradas: v[])
início
    i ← 1
    enquanto v[i] > 0 faça
        i ← i + 1
    fim enquanto
    resultado ← i - 1
fim

procedimento processa (entradas: v1[], v2[])
início
    e ← 0
    t1 ← tam(v1)
    t2 ← tam(v2)

    imprime t1, t2

    se t1 <> t2 então
        e ← 1
    senão
        para i ← 1 até t1 faça
            se (v1[i] mod 2) <> (v2[i] mod 2) então
                e ← 1
            fim se
        próximo i
    fim se
    se e = 1 então
        imprime 'um'
    senão
        imprime 'zero'
    fim se
fim

início
    leiaV (v1)
```

```

        leiaV (v2)
        processa(v1,v2)
fim

```

Considere que o comando **imprime** não passa para a linha seguinte ao terminar de imprimir. O valor impresso pelo algoritmo é:

- A) 10 10 zero
- B) 10 10 um
- C) 9 10 zero
- D) 9 9 um
- E) 9 9 zero

#### 4ª questão (valor 1.0)

Considere que os valores mostrados abaixo foram fornecidos na ordem indicada ao algoritmo a seguir:

```

4
3 3
1 2
2 1
0 -1

```

```

função f (entradas: p, c)
início
    a ← 1
    se p > c então
        a ← 3
    fim se
    se p < c então
        a ← 0
    fim se
    resultado ← a
fim

início
    leia j
    p ← 0
    para i ← 1 até j faça
        leia gp, gc
        p ← p + f(gp, gc)
    próximo i
    imprime p
fim

```

O valor impresso pelo algoritmo é:

- A) 12
- B) 9
- C) 6
- D) 7
- E) -1

### 5ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Considere que o comando **imprime** não muda para a linha seguinte ao terminar de imprimir.

**Variáveis públicas** *i*

**procedimento** f1 (**entradas:** *i*)

**início**

*i* ← *i* + 1

**imprime** *i*

**fim**

**procedimento** f2 (**saídas:** *i*)

**início**

*i* ← *i* + 1

**imprime** *i*

**fim**

**procedimento** f3()

**início**

*i* ← *i* + 1

**imprime** *i*

**fim**

**início**

*i* ← 0

f3()

f2(*i*)

f1(*i*)

**imprime** *i*

**fim**

A) 1 2 2 2

B) 1 2 3 2

C) 1 2 3 4

D) 1 2 3 3

E) 1 2 3 1

### 6ª questão (valor 1.0)

Um bombeiro hidráulico dispõe de uma quantidade **N** de canos inteiros de cobre de comprimento variável dos quais ele precisa cortar **P** pedaços iguais de cano com o maior tamanho possível. O algoritmo a seguir lê da entrada padrão (teclado) o número de pedaços de cano necessários (**P**), o número de canos inteiros disponíveis para corte (**N**) e o tamanho de cada cano inteiro.

**variáveis públicas**

tamanhoCano[], N, P

**função** verifica(**entradas:** t)

```

início
    total  $\leftarrow$  0
    para i  $\leftarrow$  1 até N faça
        total  $\leftarrow$  total + tamanhoCano[i] / t
    próximo i
    resultado  $\leftarrow$  (total  $\geq$  P)
fim

```

```

início
    MAX  $\leftarrow$  0
    leia P
    leia N
    para i  $\leftarrow$  1 até N faça
        leia tamanhoCano[i]
        se tamanhoCano[i] > MAX então
            MAX  $\leftarrow$  tamanhoCano[i]
        fim se
    próximo i

    pedacoMaximo  $\leftarrow$  MAX
    encontrou  $\leftarrow$  falso
    enquanto não encontrou faça
        se verifica(pedacoMaximo) então
            encontrou  $\leftarrow$  verdadeiro
        senão
            pedacoMaximo  $\leftarrow$  pedacoMaximo - 1
        fim se
    fim enquanto
    imprima pedacoMaximo
fim

```

Considere que as entradas foram fornecidas ao algoritmo na ordem:

3  
2  
45  
85

A saída gerada pelo algoritmo será:

- A) 40
- B) 42
- C) 44
- D) 46
- E) 82

### 7ª questão (valor 1.0)

Observe o algoritmo a seguir:

```

função max(entradas: p1, p2)
início
    se p1 > p2 então
        resultado  $\leftarrow$  p1
    senão
        resultado  $\leftarrow$  p2
    fim se

```

```

fim

início
  leia N
  para i ← 1 até N faça
    leia val[i]
  próximo i
  sum[1] ← 0
  best[1] ← 0
  para i ← 1 até N faça
    sum[i+1] ← sum[i] + val[i]
    best[i+1] ← max(best[i], sum[i+1])
  próximo i
  ans ← 0
  para i ← 1 até (N+1) faça
    ans ← max(ans, best[i] + (sum[N] - sum[i]))
  próximo i
  tmp ← 0
  para i ← 1 até N faça
    tmp ← max(0, tmp + val[i])
    ans ← max(ans, tmp)
  próximo i
  imprime ans
fim

```

Entradas:

2  
-3  
-10

Se as entradas acima forem fornecidas ao algoritmo, o valor impresso será:

- A) 0
- B) 2
- C) 4
- D) 6
- E) 9

### 8ª questão (valor 1.0)

O algoritmo a seguir que determina corretamente o maior de dez números inteiros fornecidos pelo usuário é:

i.

```

início
  para i ← 1 até 10 faça
    leia array[i]
  próximo i
  max ← 0
  para i ← 1 até 10 faça
    se array[i] > max então
      max ← array[i]
    fim se
  próximo i
  imprime max
fim

```

ii.

```
início
i ← 1
enquanto i <= 10 faça
    leia array[i]
    i ← i + 1
fim enquanto
max ← array[1]
i ← 2
enquanto i <= 10 faça
    se array[i] > max então
        max ← array[i]
    fim se
    i ← i + 1
fim enquanto
imprime max
fim
```

- A) somente i
- B) somente ii
- C) i e ii
- D) nenhum dos dois algoritmos resolve o problema
- E) nenhuma das respostas anteriores

### 9ª questão (valor 1.0)

Observe a expressão a seguir:

$$\frac{\frac{A^2}{X} + \frac{B^2}{Y}}{\frac{A \cdot C}{X \cdot Y}} + \frac{A}{B}$$

A alternativa abaixo que representa corretamente em PETEQS essa expressão, com o menor número possível de parênteses é:

- A) A\*A/X+B\*B/Y/A\*C/X\*Y+A/B
- B) (A\*A)/X+(B\*B)/Y/(A\*C)/(X\*Y)+(A/B)
- C) ((A\*A)/X+(B\*B)/Y)/((A\*C)/(X\*Y))+A/B
- D) (((A\*A+B\*B)/(X+Y))+A)/((A\*C)/(X\*Y)+B)
- E) (A\*A/X+B\*B/Y)/(A\*C/(X\*Y))+A/B

### 10ª questão (valor 1.0)

Cinco amigos costumam freqüentemente viajar juntos pelo Brasil. Eles combinaram que todas as despesas de viagem seriam rateadas por igual. Durante a viagem cada um deles anota suas despesas e o acerto é feito no regresso para casa. O algoritmo a seguir calcula quanto cada um dos amigos tem a pagar ou a receber.

```
início
soma ← 0
para i ← 1 até 5 faça
```



```
    leia array[i]
    soma ← soma + array[i]
próximo i
media ← soma/5.0
para i ← 1 até 5 faça
    imprime array[i]-media, ' ', ' '
próximo i
fim
```

Se as entradas fornecidas ao algoritmo forem 100, 120, 80, 150 e 50, nessa ordem, a saída gerada será:

- A) 20.0, 40.0, 0.0, 70.0, -30.0
- B) -20.0, 0.0, -40.0, 30.0, -70.0,
- C) 20.0, 0.0, 40.0, -30.0, 70.0,
- D) 0.0, 20.0, -20.0, 50.0, -50.0,
- E) 0.0, -20.0, 20.0, -50.0, 50.0,

Questão					
1	A	B	C	D	E
2	A	B	C	D	E
3	A	B	C	D	E
4	A	B	C	D	E
5	A	B	C	D	E
6	A	B	C	D	E
7	A	B	C	D	E
8	A	B	C	D	E
9	A	B	C	D	E
10	A	B	C	D	E