



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 2014

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) **Todas as respostas devem ser transcritas no local apropriado, no cartão de respostas a seguir.**

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

Uma prática comum no mercado financeiro é comprar um ativo no dia de uma forte queda e vendê-lo alguns dias depois, supostamente com lucro. O algoritmo a seguir calcula o retorno que um investidor teria se comprasse um ativo no dia em que o preço cai abaixo de R\$ 5,00 e o vendesse cinco dias depois.

início

```
    fechamento ← [10, 9, 8, 2, 3, 5, 6.5, 8, 8.5, 8, 7, 5.5, 3]
```

```
    para i ← 2 até tamanho(fechamento) faça
```

```
        se (fechamento[i] <= 5) E (fechamento[i - 1]) > 5 então
```

```
            se (i + 5) > tamanho(fechamento) então
```

```
                imprima 'Erro'
```

```
            senão
```

```
                imprima fechamento[i + 5]/fechamento[i]
```

```
            fim se
```

```
        fim se
```

```
    próximo i
```

fim

O valor impresso pelo algoritmo é:

- A) 2.67
- B) 1.40
- C) 4.25
- D) 3.50
- E) Erro

2ª questão (valor 1.0)

O algoritmo a seguir determina o retorno obtido por um investidor com uma estratégia de comprar um ativo e vendê-lo três dias depois.

início

```
    fechamento ← [100.0, 112.0, 109.0, 113.0,  
                  100.0, 114.0, 115.0, 114.0,  
                  100.0, 111.0, 112.0, 115.0]
```

```
    compras ← [1, 5, 9]
```

```
    ganho ← 1
```

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

```
        d ← compras[i]
```

```
        ganho ← ganho * fechamento[d + 3]/fechamento[d]
```

```
    próximo i
```

```
    imprima ganho
```

fim

O valor impresso pelo algoritmo está dentro da faixa:

- A) 1.0 a 1.1
- B) 1.1 a 1.2
- C) 1.2 a 1.3
- D) 1.3 a 1.4
- E) 1.4 a 1.5

3ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

```

função quemSabe(entradas: num1, num2)
início
    num3 ← num1 + num2
    num2 ← num2 + num3 * 2
    resultado ← num2
fim

função misterio(entradas: num1, num2)
início
    num1 ← quemSabe(num1, num2)
    num2 ← quemSabe(num2, num1)
    resultado ← num2
fim

início
    num1 ← 2
    num2 ← 13
    imprima misterio(num2 MOD 5, 1 + num1*2)
fim

```

- A) 47
- B) 58
- C) 66
- D) 73
- E) 81

4ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

```

procedimento geraVetor(saídas: v1[], v2[])
início
    para i ← 1 to tamanho(v1) faça
        se v1[i] < v2[i] então
            v1[i] ← v2[i]
        fim se
    próximo i
fim

início
    v1 ← [12, 3, 5, -7, 8, 0, 4, 14, 67, -10]
    v2 ← [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

    geraVetor(v1, v2)
    soma ← 0
    para i ← 1 até tamanho(v2) faça
        soma ← soma + v1[i]
    próximo i
    imprima soma
fim

```

- A) 55
- B) 96
- C) 136
- D) 151
- E) Nenhuma das respostas anteriores

5ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir? Nesta questão considere que o comando **imprima** não muda de linha ao final do comando. Considere também a existência de um comando **imprimaln** que força uma mudança de linha antes da próxima impressão.

início

```
para i ← 1 até 5 faça
    matrix[i] ← i
    imprima matrix[i], " "
próximo i
imprimaln
```

```
matrix1 ← matrix
```

```
para j ← 1 até 3 faça
    para i ← 1 até 5 faça
        matrix1[i] ← matrix1[i] + 1
        imprima matrix1[i], " "
    próximo i
próximo j
imprimaln
```

fim

A)

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

B)

1	2	3	4	5
2	3	4	5	6
3	4	5	6	0
4	5	6	0	0

C)

1	2	3	4	5
1	0	0	0	0
2	5	0	0	0
3	5	6	0	0

D)

1	2	3	4	5
1	0	0	0	0
5	2	0	0	0
5	5	3	0	0

E)

1	2	3	4	5
1	0	0	0	0
5	2	0	0	0
3	4	5	0	0

6ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir, se o valor fornecido pelo usuário for **24**?

Considere que o comando **imprima** não muda de linha ao imprimir.

```
início
    menor ← 0
    maior ← 128
    passos ← 0
    leia num
    achou ← 0
    enquanto achou = 0 faça
        meio ← (maior + menor) / 2
        se num > meio então
            menor ← meio
        senão
            se num < meio então
                maior ← meio
            senão
                imprima num
                achou ← 1
        fim se
    fim se
    passos ← passos + 1
fim enquanto
imprima passos
fim
```

- A) 24 0
- B) 24 2
- C) 10 4
- D) 24 4
- 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 imprimir. Considere também a existência de um comando **imprimaln** que força uma mudança de linha antes da próxima impressão.

```
início
    c[1] ← 3
    c[2] ← 5
    c[3] ← 1
    c[4] ← 4
    c[5] ← 6
    para i ← 1 até 5 faça
        para j ← 1 até c[i] faça
            imprima '*'
        próximo j
        imprimaln
    próximo i
fim
```

- A) ***

*

- B) *
**

- C) *****

**
*
- D) *****

*

- E) Nenhuma das respostas anteriores

8ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir, se os valores fornecidos pelo usuário forem 9, 8, 7, 6, 5, 4 e 6?

```
procedimento leVetor (entradas: t, saídas: v[])  
início
```

```
    para i ← 1 até t faça  
        leia v[i]  
    próximo i  
fim
```

```
função func (entradas: t, v, val)  
início
```

```
    res ← -1  
    para i ← 1 até t faça  
        se v[i] = val então  
            res ← i  
        fim se  
    próximo i  
    resultado ← res  
fim
```

```
início
```

```
    leVetor(6, v)  
    leia valor  
    f ← func(6, v, valor)  
    se f < 0 então  
        imprime 'Nao existe'  
    senão  
        imprime 'Existe ', f  
    fim se  
fim
```

- A) Nao existe
B) Existe 6
C) Existe -1
D) Existe 4
E) Nenhuma das respostas anteriores

9ª questão (valor 1.0)

Matemáticos gostam de usar sequências de números inteiros que seguem várias leis de formação interessantes. Por exemplo, 0, 1, 4, 9, 16, ... é a sequência dos quadrados dos números consecutivos 0, 1, 2, 3, 4, ...

O algoritmo a seguir calcula os 6 primeiros termos de uma sequência.

```

início
    array[1] ← 6
    array[2] ← 10

    i ← 3
    enquanto i <= 6 faça
        array[i] ← array[i-1] + array[i-2]/2
        i ← i + 1
    fim enquanto
    imprima array[6]
fim

```

O sexto termo da sequência vale:

- A) 24
- B) 33
- C) 45
- D) 18
- E) Nenhuma das respostas anteriores

10ª questão (valor 1.0)

O que será impresso pelo algoritmo a seguir?

```

função fm(entradas: f[], t)
início
    m ← 0.0
    para i ← 1 até t faça
        m ← m + f[i]
    próximo i
    resultado ← m/t
fim

```

```

função fd(entradas: f[], t)
início
    m ← fm(f, t)
    d ← 0.0
    para i ← 1 até t faça
        d ← d + f[i] - m
    próximo i
    resultado ← d/t
fim

```

```

início
    f[1] ← 10
    f[2] ← 20
    f[3] ← 30
    f[4] ← 40
    f[5] ← 50

    m ← fd(f, 5)
    imprima m
fim

```


- A) 0.0
- B) 20.0
- C) 20.0
- D) 30.0
- E) Nenhuma das respostas anteriores