



Instituto Federal de Educação, Ciência e Tecnologia do Ceará - IFCE
Programa de Pós-Graduação em Ciência da Computação - PPGCC
Curso de Mestrado Acadêmico em Ciência da Computação

Seleção PPGCC 2018.1 - Prova Escrita

Observações:

- Esta avaliação possui 12 páginas, 20 questões objetivas (**parte objetiva**) e 1 questão dissertativa (**parte dissertativa**). Verifique se há algum erro na impressão da avaliação.
- Você não pode usar livros, notas de aulas, ou dispositivos eletrônicos que possam conter assuntos relacionados à avaliação. Além disso, a avaliação é estritamente pessoal. Qualquer infração relacionada implicará na anulação da avaliação.
- O tempo limite para realização da prova é de 2 horas.
- Você deve preencher suas respostas da **parte objetiva** deste exame no **cartão resposta** (página 2). Quaisquer anotações/respostas fora do cartão resposta serão desconsideradas pela comissão de avaliação do PPGCC.

Fortaleza, 7 de Dezembro de 2017.

PARTE OBJETIVA

Questão 1

```
int misterio(int a, int b){
    // assumo que há aqui verificação dos
    // valores a e b (a e b devem ser positivos)
    while(a >= b){
        a = a-b;
    }
    return a;
}
```

Considerando o trecho de código acima, e que os argumentos a e b passados para a função *misterio* são inteiros positivos. O que a função *misterio* computa?

- a) o maior valor entre a e b
- b) a diferença entre a e b
- c) o resto da divisão de a por b
- d) a potência a^b

Questão 2 Qual será a saída do seguinte programa?

```
// Atenção, apesar do código abaixo ser escrito em linguagem C,
// o programa não possui diferenças relevantes com relação às
// linguagens Java, Python ou PHP.
```

```
#include <stdio.h>
int main( )
{
    int i=2,j=3,k,l;
    float a,b;
    k = i/j * j;
    l = j/i * j;
    a = i/j * j;
    b = j/i * i;
    printf("%d,%d,%f,%f",k,l,a,b);
    return 0;
}
```

- a) 3,0,0,0
- b) 0,3,0.000000,2.000000
- c) 0,0,0.000000,0.000000
- d) Error

Questão 3 Uma máquina leva 200seg. para fazer a ordenação pelo método da bolha (Bubble Sort) de 200 nomes. Em 800seg. quantos nomes aproximadamente podem ser ordenados?

- a) 800 nomes.
- b) 750 nomes.
- c) 600 nomes.
- d) 400 nomes.

Questão 4 Considere o algoritmo a seguir.

```
mist(n)
  if n <= 2
    f = 1
  else
    f = mist(n-1) + mist(n-2)

  return f
```

O que o procedimento *mist* acima, em pseudocódigo, computa?

- a) O n -ésimo termo da sequência de Fibonacci.
- b) O fatorial de n .
- c) O valor $n - 1 + n - 2$.
- d) Retorna sempre o valor 1.

Questão 5 Em qual dos tipos de dados abaixo, o operador % não pode ser usado?

- a) Variável inteira.
- b) Variável real.
- c) Constante inteira.
- d) Todas as opções anteriores.

Questão 6 O fator tempo, relacionado à complexidade, quando usado para determinar a eficiência de algoritmos é medido em:

- a) Número de Microsegundos.
- b) Número de Operações Chave.
- c) Número de Rotinas.
- d) Tamanho em kB (kilobytes).

Questão 7 A operação de busca binária em um vetor ordenado com n elementos possui complexidade temporal dada por:

- a) $O(n)$
- b) $O(n^2)$
- c) $O(n \log n)$
- d) $O(\log n)$

Questão 8 Qual das seguintes relações binárias **não** representa uma função?

- a) $R_1 = \{(3, 6), (4, 15)\}$
- b) $R_2 = \{(4, 5), (7, 2), (7, 9)\}$
- c) $R_3 = \{(-2, 5), (-1, 5), (0, 5), (1, 5), (2, 5)\}$
- d) $R_4 = \{(x, y) \mid y = 3x^2 + 2\}$

Questão 9 Que recursos as linguagens oferecem para evitar falhas de acesso a um endereço de memória inexistente?

- a) Ponteiros.
- b) Tratamento de Exceções.
- c) DMA (Acesso Direto à Memória).
- d) SWAP de Memória em Disco.

Questão 10 Seja o conjunto $A = \{0, 1\}$. Quantas relações binárias distintas podem ser definidas sobre o conjunto A ?

- a) 2
- b) 4
- c) 8
- d) 16

Questão 11 Sobre o algoritmo QuickSort, e considerando n o número de elementos do vetor a ser submetido para ordenação, assinale a alternativa FALSA.

- a) A complexidade temporal do pior caso é $O(n^2)$.
- b) A complexidade temporal do melhor caso é $O(n \log n)$.
- c) A complexidade do caso médio é $O(n^2 \log n)$.
- d) É um algoritmo de ordenação que funciona *inplace*, ou seja, cuja ordenação é efetuada com a própria sequência, sem utilizar sequências (memória relevante) adicionais.

Questão 12 Sobre o algoritmo *InsertionSort* (ordenação por inserção), e considerando n o número de elementos do vetor a ser submetido para ordenação, assinale a alternativa FALSA.

- a) Possui complexidade temporal do melhor caso de $O(n)$.
- b) Possui complexidade temporal do pior caso de $O(n^2)$.
- c) Possui complexidade espacial de $O(n)$.
- d) É um algoritmo de ordenação não estável, ou seja, não preserva a ordem relativa de chaves iguais.

Questão 13 Considere um heap binário de mínimo (*MinHeap*) com n elementos. Assinale a alternativa FALSA.

- a) Pode ser implementado como uma árvore binária balanceada usando vetor..
- b) A operação básica de extrair/remover o menor elemento (*extract-min()*) executa em tempo $O(\log n)$.
- c) Permite encontrar o menor elemento em tempo constante. Ou seja, a operação *find-min()* executa em tempo constante.
- d) Podemos construir um heap a partir de um vetor qualquer não ordenado de n elementos em tempo $O(\log n)$.

Questão 14 Sobre o algoritmo de ordenação MergeSort, quando aplicado a um vetor qualquer de n elementos, assinale a alternativa FALSA.

- a) É um exemplo de algoritmo de divisão e conquista.
- b) Permite ordenar em tempo $O(n \log n)$.
- c) A implementação clássica do algoritmo MergeSort não consiste de um algoritmo de ordenação *in-place*.
- d) Tem custo de ordenação no melhor caso de $O(n)$.

Questão 15 Considere uma tabela de espalhamento (*hash table*) de comprimento igual a 11, na qual a técnica de resolução de colisões utilizada é a de encadeamento. Nessa tabela, as posições são numeradas (indexadas) com os valores $0, 1, 2, \dots, 10$, o mapeamento de chaves para posições usa a função hash definida por $h(k) = k \bmod 11$, onde k é o valor da chave, e \bmod é o operador de módulo, e os números 1, 5, 18, 20, 4, 12, 10, 34, 15, 28 e 17 foram as chaves inseridas, nessa ordem, nessa tabela de espalhamento que estava inicialmente vazia. Qual a quantidade de posições em que houve colisão durante as inserções das chaves?

- a) 1.
- b) 2.
- c) 3.
- d) 4.

Questão 16 Sobre o tipo de dados abstrato Pilha, assinale a alternativa INCORRETA.

- a) É possível implementar uma Pilha utilizando uma lista simplesmente encadeada.
- b) Possui complexidade temporal $\Omega(\log n)$ para a operação *push(element)* — inserir na Pilha.
- c) Possui complexidade temporal constante $\Theta(1)$ para a operação *pop()* — remover da Pilha.
- d) A inserção e remoção devem ocorrer no mesmo extremo da estrutura, chamada de topo da Pilha.

Questão 17 Considere o pseudocódigo abaixo.

```
mist2 (a, b)
  a = a + b;
  b = a - b;
  a = a - b;
  print(a);
  print(" e ");
  print(b);
```

Considerando ainda que as variáveis a e b são inteiras, o que será exibido após a chamada da função *mist2(10, 5)*?

- a) 5 e 10.
- b) 10 e 5.
- c) 15 e 5.
- d) Nenhuma das respostas anteriores.

Questão 18 Sobre árvores binárias de busca, assinale o item INCORRETO.

- a) Possui complexidade temporal, no pior caso, de $O(n)$ para a operação de remoção em uma árvore com n nós.
- b) Um caminhamento inOrder (em ordem ou interfixado) na árvore gera uma sequência ordenada das chaves armazenadas na árvore.
- c) Permite efetuar buscas e inserções em tempo $O(h)$, em que h denota a altura da árvore.
- d) Consiste em uma árvore balanceada, ou seja, $h = O(\log n)$, em que h é a altura da árvore e n é o número de nós.

Questão 19 Considere uma estrutura de dados em que cada elemento armazenado apresenta ligações de apontamento com seu sucessor e com o seu predecessor, o que possibilita que ela seja percorrida em qualquer sentido. Trata-se de

- a) uma árvore binária.
- b) uma lista duplamente encadeada.
- c) uma fila.
- d) um grafo.

Questão 20 Considere o algoritmo a seguir.

```
int mist (int x, int v[], int e, int d){
    int m = (e + d)/2;
    if (v[m] == x)
        return m;
    if (e >= d)
        return -1;
    else
        if (v[m] < x)
            return mist(x, v, m+1, d);
        else
            return mist(x, v, e, m-1);
}
```

A que corresponde o procedimento *mist* acima?

- a) O algoritmo de ordenação MergeSort, onde x é uma variável auxiliar, v é um vetor não ordenado, a variável e denota o limite inferior e d representa o limite superior.
- b) O algoritmo de busca binária, onde x é a chave a ser buscada, v é um vetor ordenado, a variável e denota o limite inferior e d representa o limite superior.
- c) O procedimento de inserção em uma árvore binária de busca, onde x é o elemento a ser inserido, v é o vetor representando a árvore binária de busca, e as variáveis e e d são auxiliares.
- d) Nenhuma das respostas anteriores.

PARTE DISSERTATIVA

Você irá começar a parte dissertativa desta avaliação. Disserte sobre **o passado, o presente e o futuro** de **UM** dos seguintes temas de pesquisa:

- a) Inteligência Artificial Aplicada
- b) Processamento Digital de Imagens e Sinais
- c) Redes de Computadores
- d) Sistemas Distribuídos
- e) Aplicações da Internet
- f) Informática na Saúde
- g) Aplicações Embarcadas

Obs: a escolha **não** está necessariamente vinculada à linha de pesquisa escolhida no ato da inscrição.

A sua dissertação deverá ser feita utilizando o espaço destinado na página a seguir (página 12). Textos presentes fora do espaço destinado (25 linhas) e na folhas rascunho serão desconsiderados para avaliação.

RASCUNHO

RASCUNHO

RASCUNHO

CARTÃO RESPOSTA

Nome do candidato: _____

Questão	Resposta
1	
2	
3	
4	
5	
6	
7	
8	
9	
10	
11	
12	
13	
14	
15	
16	
17	
18	
19	
20	

Tema do texto: _____

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25
