



Teste de Conhecimento

avalie sua aprendizagem

ALGORITMOS E COMPLEXIDADE

DGT1348_202101110137_TEMAS

Lupa



Aluno: DOUGLAS MATOS DA SILVA

Matr.: 202101110137

Disc.: ALGORITMOS E COMPLEX

2023.1 EAD (G) / EX

Prezado(a) Aluno(a),

Você fará agora seu **TESTE DE CONHECIMENTO!** Lembre-se que este exercício é opcional, mas não valerá ponto para sua avaliação. O mesmo será composto de questões de múltipla escolha.

Após responder cada questão, você terá acesso ao gabarito comentado e/ou à explicação da mesma. Aproveite para se familiarizar com este modelo de questões que será usado na sua AV e AVS.

ANÁLISE DE ALGORITMO

1. Considere o algoritmo em pseudocódigo, descrito a seguir.

```

Para i=0 até n
    Início
        j = 1
        enquanto j < n
            Início
                j = 2 * j
                para k = 0 até j
                    Início
                        execute f
                    fim
                fim
            fim
        fim
    fim

```

Calcule a complexidade do algoritmo, sabendo que a função f tem complexidade igual a $O(n^2)$.

- $O(n^3)$
- $O(n^3)$
- $O(n^3 \log^2(n))$
- $O(n^4 \log(n))$
- $O(n^3 \log(n))$

Data Resp.: 19/04/2023 19:26:15

Explicação:

Vamos analisar o código simplificado abaixo:

J=1

Enquanto j < n

J = 2*j

Para k = 0 ate j

Operação elementar

Para facilitar, vamos fazer n = 2^k

J = 1 à j = 2, com 3 ($2^1 + 1$) iterações

J = 2 à j = 4 com 5 ($2^2 + 1$) iterações

J = 4 à j = 8 com 9 ($2^3 + 1$) iterações

J = 8 à j = 16 com 17 ($2^4 + 1$) iterações

J = 2^k à j = ($2^{k+1} + 1$) iterações

O total de iterações é a soma $\sum_{i=1}^{\log_2 n} (2^i + 1) < 2^k \sum_{j=1}^{\log_2 n} 1 = 2^k \log_2 n$, porém $2^k = n$, assim a complexidade do código é $n \log n$. Considerando OP com complexidade constante. Como OP é quadrática, temos que o código analisado é $n^3 \log n$.

O for mais externo se repete n vezes, assim a complexidade total do algoritmo é $n^4 \log n$

2. Registros são exemplos de tipos de dados heterogêneos. Assim, sobre tipos de dados elementares e estruturados, é correto afirmar que os elementos de um registro são de tamanhos potencialmente diferentes e residem em posições de memória:

- espalhadas
- aleatórias
- flexíveis
- adjacentes
- procedimentais

Data Resp.: 19/04/2023 19:27:02

Explicação:

Os elementos de um registro são alocados em posições adjacentes de memória.

RECURSIVIDADE

3. Ano: 2020 Banca: FAPEC Órgão: UFMS Prova: FAPEC - 2020 - UFMS - Técnico de Tecnologia da Informação

Considere a seguinte função recursiva: função recursiva(x : inteiro): inteiro inicio

se x = 1 então

retorna -x

senão

retorna -5 * recursiva(x - 1) + x

fimse

fimfuncão

Qual é o valor retornado pela função se ela for chamada com x = 4?

- 56
 143
 -56
 164
 -143

Data Resp: 19/04/2023 19:27:45

Explicação:

Resposta correta: 164

4. O código abaixo é uma implementação:

```
public class Misterio {
    public static long Misterio(long x) {
        if (x == 1)
            return 1;
        else
            return x * Misterio(x-1);
    }
}
```

Iterativa da exponenciação
 Recursiva da série de Fibonacci
 Recursiva da exponenciação
 Recursiva do fatorial
 Iterativa da série de Fibonacci

Data Resp: 19/04/2023 19:28:07

Explicação:

Resposta correta: Recursiva do fatorial

ALGORITMOS DE ORDENAÇÃO AVANÇADOS

5. A ordenação de elementos em um vetor pode ser executada a partir de diversos algoritmos conhecidos que são adequados para situações específicas. Sobre algoritmos de ordenação, analise as seguintes afirmativas:

- I. O algoritmo *bubble sort* é eficiente para ordenar poucos elementos, mas é lento para ordenar muitos itens.
 II. O algoritmo *selection sort* para ordenação crescente consiste em mover o menor valor do vetor para a primeira posição; depois, o segundo menor para a segunda posição; e assim sucessivamente, até os dois últimos valores.
 III. O algoritmo *quick sort* ordena os valores de um vetor por meio de sucessivas seleções do elemento correto a ser posicionado em um segmento ordenado.

Está(ão) correta(s) a(s) afirmativa(s):

- I e II
 I apenas
 I, II e III
 II apenas
 I e III

Data Resp: 19/04/2023 19:28:30

Explicação:

A resposta correta é: I e II

6. Considere que os percentuais foram inseridos no vetor vet de 5 posições, a partir da posição 1, na seguinte sequência: 25.33, 27.72, 27.10, 26.90 e 27.31, ou seja, com os dados de 2008 até 2012. Um técnico em processamento de dados do TCE-RS utilizou um método para ordenar os dados do vet. O método realizou os seguintes passos no processo de ordenação:

- Passo 1 - 25.33 27.72 27.10 26.90 27.31;
- Passo 2 - 25.33 27.10 27.72 26.90 27.31;
- Passo 3 - 25.33 26.90 27.10 27.72 27.31;
- Passo 4 - 25.33 26.90 27.10 27.31 27.72.

Trata-se do método de ordenação:

- Fast sort*
 Quick sort
 Bubble sort
 Insertion sort
 Selection sort

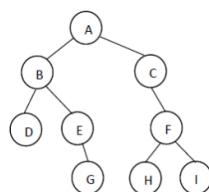
Data Resp: 19/04/2023 19:29:37

Explicação:

A resposta correta é: *Insertion sort*

ALGORITMOS EM ÁRVORES BINÁRIA E ÁRVORE AVL

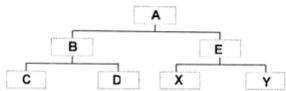
7. Analise a seguinte árvore binária e assinale a alternativa correta.



- "B" tem grau de saída 3 e "C" grau 2.
 "A" é filho de todos.
 Com exceção do nó "A", que é raiz, os demais nós são conhecido como folhas.
 "B" e "C" são caules da árvore.
 TA é a subárvore enraizada em "A"; portanto toda a árvore.

8. A estrutura abaixo representa uma célula de uma árvore em linguagem C:

```
typedef struct _no {
    int chave;
    struct _no *esq, *dir;
} no;
```



Assinale a alternativa correta sobre qual sequência será impressa ao executar um caminhamento na árvore abaixo, conforme o código escrito em linguagem C a seguir:

```
void ordem(no *arvore) {
    if(arvore != NULL) {
        printf("%d", arvore->chave);
        ordem(arvore->esq);
        ordem(arvore->dir);
    }
}

 YXEABC
 AEXYBCD
 ABCDEYX
 CBDAXEY
 ABCDEXY
```

Data Resp: 19/04/2023 19:31:18

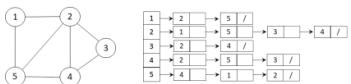
Explicação:

A resposta correta é: ABCDEXY

ALGORITMOS EM GRAFOS

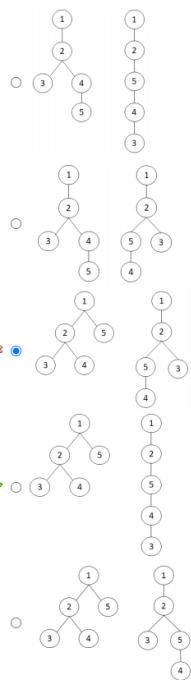
9. (FCM - IFN-MG - Ciências da Computação: Teoria da Computação - 2018)

Considere o grafo abaixo assim como sua representação por lista de adjacência:



Referência: CORMEN, Thomas H. et al. Algoritmos: teoria e prática. Rio de Janeiro: Cam-
pus, 2012.

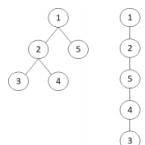
A Árvore em Largura e a Árvore em Profundidade, respectivamente, tendo como raiz o vértice 1, são:



Data Resp: 19/04/2023 19:31:29

Explicação:

Resposta correta:



10. (COMPERVE - UFRN - Engenheiro - Engenharia da Computação - 2019)

O código abaixo pode ser utilizado para atravessar um grafo:

Entrada: um gráfico G e um vértice v de G

```
Saída: todos os vértices alcançáveis de v marcados  
função DFS(G,v):  
    marque v  
    para todas as arestas adjacentes a v, faça  
        se vértice w não estiver marcado, então  
            Chame recursivamente DFS(G,w)  
        fim se  
    fim para  
fim função
```

Entre os diversos tipos de algoritmos utilizados para atravessar grafos, esse código implementa o algoritmo:

- ✓ Busca em profundidade ou depth first search.
 Busca exaustiva ou brute force search.
 Busca melhor-primeiro ou best first search.
 Busca em largura ou breadth first search.
 Busca pelo caminho mínimo (shortest path).

Data Resp: 19/04/2023 19:31:38

Explicação:

Resposta correta: Busca em profundidade ou depth first search.

 Não Resposta  Não Gravada  Gravada

Exercício iniciado em 19/04/2023 19:18:06.