



Gabarito da Prova de Conhecimento

Resolva as questões 1 e 2 apresentando códigos em uma das linguagens permitidas (C, C++ e JAVA), sempre que solicitada a apresentação de funções.

- 1) Escreva um algoritmo que ao receber um vetor de inteiros (não negativos) de 1001 elementos não repetidos, encontre e imprima a mediana dos valores encontrados no vetor. **(1,5 pontos)**

OBS: a mediana é o valor que a divide uma amostra ao meio, isto é, 50% dos elementos da amostra são menores ou iguais à mediana e os outros 50% são maiores ou iguais à mediana

Restrições: não é permitido usar vetores ou estruturas auxiliares;
não é permitido alterar os valores do vetor recebido.

```
publico vazio imprimeMediana(Vetor vet){
    inteiro tamanhoVet,contMediana,contMaior,indiceMed,contCompara;
    booleano naoAchou;

    tamanhoVet = vet.pegarTamanho(); //ele tem tamanho 1001
    contMediana = 0;
    naoAchou = verdadeiro;

    enquanto ((contMediana < tamanhoVet) && ~naoAchou){
        contMaior = 0;
        para(contCompara = 0;contCompara < tamanhoVet; contCompara++){
            se (vet.pegarElement(contMediana) > vet.pegarElemento(contCompara)){
                contMaior++;
            }
        }
        se (contMaior == 500){ //se encontrar alguém maior a somente 500, eh a mediana!!
            naoAchou = falso;
            indiceMed = contMediana;
        }
        contMediana++;
    }
    imprima("A mediana eh",vet.pegarElemento(indiceMed));
}
```

- 2) Em matrizes, quando utilizado o método de La Place é empregado para o cálculo do determinante, uma otimização conhecida para o algoritmo é escolher a linha ou coluna com mais zeros para realizar o cálculo com os cofatores. Dessa forma, implemente um método que ao receber uma matriz quadrada de inteiros como entrada, descubra quem tem mais zeros na matriz recebida se uma linha ou uma coluna. Ao final, imprima se a resposta foi uma linha ou coluna, o índice e a quantidade de zeros encontrados nela. **(2,0 pontos)**



Universidade Federal do Amazonas
Instituto de Computação
Programa de Pós-graduação em Informática



```
publico vazio linhaOuColunaComMaisZeros(Matriz mat, int numValue){
    inteiro contColuna,contLinha,ordem,indiceCMaior,indiceLMaior,contNum,lMaior,cMaior;
    ordem = mat.pegarOrdem();

    lMaior = 0;
    indiceLMaior = 0;
    // primeiro laço encontra a linha com mais zeros e a quantidade
    para (contLinha = 0; contLinha < ordem; contLinha++){
        contNum = 0;
        para (contColuna = 0; contColuna < ordem; contColuna++){
            se (mat.pegarElemento(contLinha,contColuna) == numValue){
                contNum++;
            }
        }
        se (contNum > lMaior){
            lMaior = contNum;
            indiceLMaior = contLinha;
        }
    }
    cMaior = 0;
    indiceCMaior = 0;
    // segundo laço encontra a coluna com mais zeros e a quantidade
    para (contColuna = 0; contColuna < ordem; contColuna++){
        contNum = 0;
        para (contLinha = 0; contLinha < ordem; contLinha++){
            se (mat.pegarElemento(contLinha,contColuna) == numValue){
                contNum++;
            }
        }
        se (contNum > cMaior){
            cMaior = contNum;
            indiceCMaior = contColuna;
        }
    }
    // imprime a resposta pedida no enunciado da questao
    if (cMaior > lMaior){
        imprima("Coluna");
        imprima(indiceCMaior);
        imprima(cMaior);
    }
    else{
        imprima("Linha");
        imprima(indiceLMaior);
        imprima(lMaior);
    }
}
```



Universidade Federal do Amazonas
Instituto de Computação
Programa de Pós-graduação em Informática



- 3) Com relação ao algoritmo de ordenação Quicksort, sabe-se que a escolha do pivô de partição é o fator decisivo no desempenho deste algoritmo. Explique: **(2,0 pontos)**

- Qual é o melhor caso para essa escolha do pivô? Explique sua resposta.

O melhor caso para a escolha do pivô é sempre quando o valor apontado por este divide o array em 2 partes iguais. A maior frequência deste caso maximiza a eficiência do algoritmo. Isso acontece devido a quantidade de divisões e chamadas recursivas do procedimento de particionamento ser a mínima possível (log de n na base 2 vezes).

- Qual é o pior caso para essa escolha do pivô? Explique sua resposta.

O pior caso para a escolha do pivô será quando o valor selecionado for o menor ou o maior elemento do vetor. Uma grande frequência deste caso incorre em degradação do desempenho do algoritmo, sendo que uma chamada do particionamento será feita para somente um elemento e a outra chamada, para um array de n-1. Este caso acaba maximizando o número de chamadas do procedimento de particionamento (n vezes).

- 4) Apresente a função de custo para determinar o número de vezes que a operação “faz_algo” é executada no algoritmo abaixo em função da variável *n*. **(1,5 pontos)**

```
VERIFICA_ALGO(int n) {  
    INT x,y,z;  
  
    FOR(x=0; x<n; x++) {  
        FOR(y=x; y<x+10; y++) {  
            FOR(z=0; z<n; z++) { faz_algo; }  
        }  
    }  
}
```

10n² (Dez vezes *n* elevado ao quadrado)



- 5) Considere uma arquitetura de computador dotada de memória virtual paginada, *Translation Lookaside Buffer* (TLB) e memória cache. A respeito dessa arquitetura, analise as seguintes afirmações: **(1,5 pontos)**

- I – O número de entradas da tabela de páginas corresponde ao número de páginas comportadas pela memória virtual.
- II – O número de entradas da tabela de páginas corresponde ao número de páginas comportadas pela memória física.
- III – Uma falta de página nem sempre implica em falha na memória cache.
- IV – Uma falha na TLB nem sempre implica em falha na memória cache.

Está correto o que se afirma em:

- A – I e III, apenas
- B – I e IV, apenas**
- C – II e III, apenas
- D – II e IV, apenas
- E – I, II, III e IV

- 6) Dada a linguagem $L = \{0^m 1^n \mid m \neq n\} \cup \{0^n 1^n \mid n \geq 0\}$, analise as seguintes afirmações: **(1,5 pontos)**

I - Ou existe um autômato finito não-determinístico que reconhece L , ou existe um autômato finito determinístico que reconhece L , mas não ambos;

II - A linguagem L pode ser representada pela expressão regular $0^*(1^*)^*$;

III - A linguagem L pode ser representada pela expressão regular 0^*1^* ;

IV - A linguagem L pode ser gerada pela gramática $G = (\{P, A\}, \{0, 1\}, R, P)$, em que R contém as regras (onde λ é a palavra vazia);

$$P \rightarrow 0P \mid 1A \mid \lambda$$

$$A \rightarrow 1A \mid 1$$

V - A linguagem L pode ser gerada pela gramática $G = (\{P, A\}, \{0, 1\}, R, P)$, em que R contém as regras (onde λ é a palavra vazia);

$$P \rightarrow 0P \mid 1A \mid \lambda$$

$$A \rightarrow 1A \mid \lambda$$

Marque a alternativa correta:

- a) Estão corretas apenas as afirmativas I e II.
- b) Estão corretas apenas as afirmativas II e III.
- c) Estão corretas apenas as afirmativas II, III e V.**
- d) Estão corretas apenas as afirmativas I, III e V.
- e) Estão corretas apenas as afirmativas II, III e IV.