

Exercício Algoritmo Estruturado II - valor (2 pontos)

Aluno(a): _____ Data: 06/04/2021

Aluno(a): _____

Professor.: Ronilson R. Pinho

- 1) Quantas unidades de tempo são necessárias para rodar o algoritmo abaixo? Qual a ordem de complexidade de tempo?

início

i, j: inteiro

A: vetor inteiro de n posições

i = 1 $\rightarrow 1$

enquanto (i < n) faça

A[i] = 0 $\rightarrow n-1$

i = i + 1 $\rightarrow n-1$

fimenquanto

para i = 1 até n faça

para j = 1 até n faça

A[i] = A[i] + (i*j) $\rightarrow n * n$

Fimpara

fimpara

Fim

$$n^2 + 1 + (n-1) + n - 1 = n^2 + n - 1$$

Ordem = n^2

- 2) Quantas unidades de tempo são necessárias para rodar o algoritmo abaixo? Qual a ordem de complexidade de tempo?

Algoritmo

Var lista: vetor [1..n] de inteiro

chave, n, i, fim, pos: inteiro

troca: booleano

Início

troca=verdadeiro $\rightarrow 1$

fim=n-1 $\rightarrow 1$

pos=1 $\rightarrow 1$

enquanto troca=verdadeiro faça

troca=falso $\rightarrow n-1$

para i de 1 até fim faça

se v[i] > v[i+1] então $\rightarrow n-1$

chave = v[i]; $\rightarrow n-1$

v[i] = v[i+1]; $\rightarrow n-1$

v[i+1] = chave; $\rightarrow n-1$

pos=i; $\rightarrow n-1$

troca=verdadeiro $\rightarrow n-1$

fimse

fimpara

fim=pos-1 $\rightarrow n-1$

fim enquanto

fim

$$(5(n-1))(2(n-1)) + 3 = (5n-5)(2n-2) + 3 =$$

$$10n^2 - 10n + 10n^2 - 10 + 3 =$$

$$20n^2 - 10n - 7$$

Ordem = n^2

3) Escreva um algoritmo recursivo capaz de gerar todos os elementos do conjunto potência dado um conjunto formado por letras.

- Caso base? $2^{\{\}} = \{\}$

- Passo da recursão:

$$2\{a,b,c\} = 2\{b,c\} \cup \{a\} \times 2\{b,c\}$$

Exemplo: $X=\{a,b,c\}$ então $2^X = \{\{\}, \{a\}, \{b\}, \{c\}, \{a,b\}, \{a,c\}, \{b,c\}, \{a,b,c\}\}$

4) [Análise de Algoritmo] Analise o algoritmo abaixo, escrito em C, que recebe dois arrays, a e b , de tamanhos iguais n . Determine:

```
float f(float* a, float* b, int n) {  
    int i, j;  
    float s = 0.0;  
    for (i=1; i<n; i++) {  
        if (a[i]>600) {  
            for (j=n-1; j>=0; j--) {  
                s += a[i]*b[j];  
            }  
        } else if (a[i]<300) {  
            for (j=n; j<n*n; j+=5) {  
                s += a[i]*b[j];  
            }  
        } else {  
            for (j=1; j<n; j=3*j) {  
                s += a[i]*b[j];  
            }  
        }  
    }  
    return s;  
}
```

a) O maior limite assintótico inferior para o melhor caso em função do parâmetro n .

b) O menor limite assintótico superior para o pior caso em função do parâmetro n .

c) As condições que o array a deve satisfazer para caracterizar o melhor caso