

Cap 1

2. O que significa dizer que uma função  $g(n)$  é  $O(f(n))$ ?

5. Qual algoritmo você prefere: um algoritmo que requer  $n^5$  passos ou um que requer  $2^n$  passos?

7. Indique se as afirmativas a seguir são verdadeiras ou falsas e justifique a sua resposta.

a)  $2^{n+1} = O(2^n)$

b)  $2^{2n} = O(2^n)$

c)  $f(n) = O(u(n))$  e  $g(n) = O(v(n)) \Rightarrow f(n) + g(n) = O(u(n) + v(n))$

d)  $f(n) = O(u(n))$  e  $g(n) = O(v(n)) \Rightarrow f(n) - g(n) = O(u(n) - v(n))$

17. Resolva as seguintes equações de recorrência:

a) 
$$\begin{cases} T(n) = T(n-1) + c & c \text{ constante}, n > 1 \\ T(1) = 0 \end{cases}$$

b) 
$$\begin{cases} T(n) = T(n-1) + 2^n & n \geq 1 \\ T(0) = 1 \end{cases}$$

c) 
$$\begin{cases} T(n) = cT(n-1) & c, k \text{ constantes}, n > 0 \\ T(0) = k \end{cases}$$

d) 
$$\begin{cases} T(n) = 3T(n/2) + n & n > 1 \\ T(1) = 1 \end{cases}$$

e) 
$$\begin{cases} T(n) = 3T(n-1) - 2T(n-2) & n > 1 \\ T(0) = 0 \\ T(1) = 1 \end{cases}$$

f) 
$$\begin{cases} T(n) = \sum_{i=1}^{n-1} 2T(i) + 1 & n > 1 \\ T(1) = 1 \end{cases}$$

g) 
$$\begin{cases} T(n) = 2T(\lfloor n/2 \rfloor) + 2n \log_2 n \\ T(2) = 4 \end{cases}$$

cuja solução satisfaz  $T(n) = O(n \log^2 n)$ .

Prove usando indução matemática em  $n$  (Manber, 1989, p. 56)

18. Considere o algoritmo a seguir. Suponha que a operação crucial é o fato de inspecionar um elemento. O algoritmo inspeciona os  $n$  elementos de um conjunto e, de alguma forma, isso permite descartar  $2/5$  dos elementos e então fazer uma chamada recursiva sobre os  $3n/5$  elementos restantes.

```
procedure Pesquisa (n: integer);  
if n <= 1  
then 'inspecione elemento' e termine  
else begin  
    para cada um dos n elementos 'inspecione elemento';  
    Pesquisa(3n/5);  
end;  
end;
```

## Cap 2

5. Responda as seguintes questões sobre **recursividade**.

a) Quando se deve e quando não se deve utilizar a recursividade para resolver problemas utilizando o computador?

b) Porque é preferível usar a versão iterativa em vez da versão recursiva quando a estrutura do programa é do tipo  $P = \text{if } B \text{ then } (S;P)$ ?

6. Determine o que faz a função recursiva a seguir:

```
function Recursiva (n: integer) : integer;  
begin  
    if n <= 0  
    then Recursiva := 1;  
    else Recursiva := Recursiva (n-1) + Recursiva (n-1);  
end
```