

EXERCÍCIOS

1. Mostre que se $f(n)$ e $g(n)$ são funções monotonicamente crescentes, então também são funções monotonicamente crescentes: $f(n) + g(n)$ e $f(g(n))$. E se além de monotonicamente crescentes, $f(n)$ e $g(n)$ são também não negativas, então $f(n) \cdot g(n)$ é monotonicamente crescente.
2. Mostre que se $P(n)$ é um polinômio, então $O(\log(P(n))) = O(\log n)$
3. Compare as seguintes funções (cada par), dizendo se uma é O , o , Ω , ω , Θ da outra. Assuma que $k \geq 1$, $\epsilon > 0$ e $c > 1$ são constantes.
 - (a) $n^{\log m}$, $m^{\log n}$
 - (b) $3^{\log_2 n}$, $\log_2((n!)^k)$ $\sim O(kn \lg n)$
 - (c) $n2^n$, 3^n

4. Encontre um contra-exemplo para as seguintes afirmações:

- (a) $f(n) = O(s(n))$ e $g(n) = O(r(n))$ implicam que $f(n) - g(n) = O(s(n) - r(n))$.
- (b) $f(n) = O(s(n))$ e $g(n) = O(r(n))$ implicam que $f(n)/g(n) = O(s(n)/r(n))$.

5. A seguinte recorrência descreve o tempo de execução de um algoritmo recursivo para multiplicação de matrizes, descrito por Pan em 1978. Qual é a complexidade de tempo assintótico deste algoritmo?

$$T(n) = 143640 \cdot T(n/70) + O(n^2)$$

Este algoritmo tem complexidade de tempo melhor que o algoritmo direto, de n^3 ?

6. Resolva as seguintes recorrências, é suficiente achar o comportamento assintótico:

- (a) $T(n) = 4T(\lceil \sqrt{n} \rceil) + 1$; $T(2) = 1$
- (b) $T(n) = 2T(\lceil \sqrt{n} \rceil) + 2n$; $T(2) = 1$

7. Encontre o comportamento assintótico da função $T(n)$ definida pela recorrência:

$$T(n) = T(n/2) + T(\lfloor \sqrt{n} \rfloor) + n^2.$$

Você pode considerar n potência de 2.

8. Heap d -ário: Um heap d -ário é como um heap binário, mas em vez de 2 filhos, cada nó pode ter d filhos, $d \geq 2$.

- (a) Como você representaria um heap d -ário em um vetor?
- (b) Qual a altura do heap d -ário de n elementos em termos de n e d ?
- (c) Dê uma implementação eficiente de *Extrai-Max* e *Insert*. Analise suas complexidades de tempo em termos de n e d .
- (d) Dê uma implementação eficiente de *Atualiza-Chave*($A, i, Chave$), que dados um heap em um vetor A , uma posição i neste vetor, e uma nova *Chave*: consiste em atualizar o valor de $A[i]$ com o valor *Chave* e deve atualizar a estrutura do heap de forma apropriada. Analise sua complexidade de tempo em termos de n e d .

9. Considere um algoritmo A que tem como entrada uma lista L de n inteiros distintos, um inteiro positivo m , e uma fita F com m números distintos. Esse algoritmo deve ler os números de F um por um, e a cada leitura, deve decidir se o número lido pertence à lista L . Há as seguintes opções de funcionamento do algoritmo A .

- (a) A efetua uma busca seqüencial em L a cada leitura;
- (b) A ordena L . Depois, a cada leitura, A efetua busca binária em L .

Há duas possibilidades para o valor de m :

- (a) $\lfloor \log \log n \rfloor$
- (b) $\lfloor \sqrt[3]{n} \rfloor$

Para cada valor de m acima, diga qual a opção de funcionamento do algoritmo que é mais eficiente (na notação assintótica). Justifique.