

Análise de Algoritmos – Lista I

Ex. 1 — Resolva as recorrências:

$$T(n) = 7T(n/7) + n$$

$$T(n) = 9T(n/3) + n^2$$

$$T(n) = 8T(n/2) + n^3$$

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

$$T(n) = T(\sqrt{n}) + 1$$

Ex. 2 — O número de linhas na saída do programa a seguir é dado por uma relação de recorrência.

funcao f(n):

 if n>1:

 print ("uma linha")

 f(n/2)

 f(n/2)

Identifique a recorrência e a resolva, dizendo quantas linhas o programa mostrará. Não use notação “O” – aqui estamos interessados na quantidade exata de linhas, e não em raciocínio assintótico.

Ex. 3 — O algoritmo de Euclides para calcular o MDC de dois números é mostrado a seguir.

mdc(m,n):

 repita:

 r = n mod m

 n = m

 m = r

 eté r=0

 retorne n

- Prove que o algoritmo de fato calcula o mínimo divisor comum de m e n.
- Determine a relação de recorrência que dá o tempo de execução deste algoritmo, e a resolva. Presuma que o tempo é dado em função de n, e que $n > m$.

Ex. 4 — O algoritmo a seguir calcula uma potência de x:

pot(x,n):

 res = 1

 para i de 1 a n-1:

 res = res * x

 retorne res

- Qual é a complexidade deste algoritmo, em função de x e n?
- Proponha um algoritmo com complexidade melhor (e calcule a complexidade dele, claro).