MC448 – Análise de Algoritmos
Lista de Exercícios 3

Orlando Lee

1. Mostre que a solução da recorrência $T(n) = T(\lceil n/2 \rceil) + 1$ pertence a $O(\lg n)$.

2. Mostre que a solução de $T(n) = 2T(\lceil n/2 \rceil) + n$ é $\Theta(n \lg n)$.

3. Mostre que a solução de $T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + n$ é $\Omega(n \lg n)$ (vimos em aula que é $O(n \lg n)$).

4. Resolva a recorrência $T(n) = T(\lceil n/3 \rceil) + T(\lfloor 2n/3 \rfloor) + 1$ pelo método de substituição.

5. Determine um bom limite assintótico (O) da recorrência $T(n) = 3T(\lfloor n/2 \rfloor) + n$ usando o método da iteração. Use o método da substituição para resolver a recorrência.

6. Argumente que a solução da recorrência $T(n) = T(\lceil n/3 \rceil) + T(\lfloor 2n/3 \rfloor) + n$ é $\Omega(n \lg n)$ usando o método da árvore de recorrência.

7. Use a árvore de recorrência para $T(n) = 4T(\lfloor n/2 \rfloor) + n$ e obtenha a classe Θ a qual a solução pertence.

8. Use o método da árvore de recorrência para resolver a recorrência $T(n) = T(\alpha n) + T((1 - \alpha)n) + n$ onde $0 < \alpha < 1$ é uma constante.

9. Use o Teorema Master para resolver as recorrências abaixo.
   (a) $T(n) = 4T(n/2) + n$
   (b) $T(n) = 4T(n/2) + n^2$
   (c) $T(n) = 4T(n/2) + n^3$

10. O tempo de execução de um algoritmo A é descrito pela recorrência $T(n) = 7T(n/2) + n^2$. Outro algoritmo A' tem complexidade de tempo descrita por $T'(n) = aT'(n/4) + n^2$. Qual é o maior inteiro a tal A' é assintoticamente mais rápido que A? **Observação:** use o Teorema Master neste.