Práctica 1

1. Probar utilizando el método de sustitución que $T(n) \in O(lg(n))$.

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

2. Sean $a, b \in \mathbb{R}^+$, utilizar el método de sustitución para encontrar funciones $f(n)$ tales que $T(n) \in \Theta(f(n))$ para las siguientes recurrencias.

a)

$$T(n) = \begin{cases} a & \text{si } n = 1 \\ 2T(\lfloor n/2 \rfloor) + b & \text{si } n > 1 \end{cases}$$

b)

$$T(n) = \begin{cases} a & \text{si } n = 1 \\ 2T(\lfloor n/2 \rfloor) + n & \text{si } n > 1 \end{cases}$$

Ayuda: Recuerde las propiedades:

- $\forall n, a, b \in \mathbb{Z}, a \neq 0 \land b \neq 0 \Rightarrow \lfloor \lfloor n/a \rfloor / b \rfloor = \lfloor n/(ab) \rfloor$
- $\forall n, a, b \in \mathbb{R}^+, a^{\log_b(n)} = n^{\log_b(a)}$

3. Utilice un árbol de recurrencia para encontrar una cota asintótica Θ para la recurrencia

$$T(n) = \begin{cases} a & \text{si } n = 1 \\ 4T(\lceil n/2 \rceil) + bn & \text{si } n > 1 \end{cases}$$

donde a y b son constantes positivas. Verifique que la cota encontrada es correcta.

4. Utilizar un árbol de recurrencia para obtener una cota asintótica para

$$T(n) = \begin{cases} a & \text{si } n \leq \alpha \\ T(n-1) + T(\alpha) + bn & \text{si } n > \alpha \end{cases}$$

donde $\alpha \geq 1$, $a, b > 0$ son constantes.

5. Utilizar el teorema Maestro para encontrar cotas asintóticas Θ para las siguientes recurrencias (asumir que $T(1) > 0$):

a) $T(n) = 4T(n/2) + n$

b) $T(n) = 4T(n/2) + n^2$

c) $T(n) = 4T(n/2) + n^3$

6. Para cada una de las siguientes funciones, determinar si son suaves o no. Demostrar.

a) $ln(n)$

b) n^2

c) n^n

7. Encontrar cotas asintóticas Θ para cada una de las siguientes recurrencias (Asumir que $T(1) > 0$):

a) $T(n) = T(n/2) + 1$

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

c) $T(n) = T(\lfloor\sqrt{n}\rfloor) + 1$. Ayuda: use "renombre de variable" con $n = 2^k$. En otras palabras, calcule primero una cota Θ para $T \circ 2^k$, usando $T(2^k) = T(\lfloor 2^{k/2} \rfloor) + 1$

8. Utilizar la técnica de la ecuación característica para encontrar cotas asintóticas Θ para las siguientes recurrencias:

a)

$$T(n) = \begin{cases} n & \text{si } n = 0 \vee n = 1 \\ 5T(n-1) - 6T(n-2) & \text{si } n > 1 \end{cases}$$

b)

$$T(n) = \begin{cases} n+1 & \text{si } n = 0 \vee n = 1 \\ 3T(n-1) - 2T(n-2) + 3 \times 2^{n-2} & \text{si } n > 1 \end{cases}$$

9. Resolver las siguientes recurrencias para el caso en que n es potencia de dos, utilizando la técnica de la ecuación característica. Expresar la solución utilizando la notación Θ.

a)

$$T(n) = \begin{cases} 1 & \text{si } n = 1 \\ 2T(n/2) + lg(n) & \text{si } n > 1 \end{cases}$$

b)

$$T(n) = \begin{cases} 1 & \text{si } n = 1 \\ 2T(n/2) + nlg(n) & \text{si } n > 1 \end{cases}$$

10. * La siguiente recurrencia es muy utilizada en los análisis de algoritmos divide y vencerás:

$$T(n) = aT(n/b) + cn^k \text{ para } n > n_0$$

donde $n_0 \geq 1$, $a \geq 1$, $b \geq 2$ y $k \geq 0$ son enteros y $c \in \mathbb{R}^+$.
Probar utilizando la técnica de ecuación característica que:

$$T(n) \in \begin{cases} \Theta(n^k) & \text{si } a < b^k \\ \Theta(n^k lg(n)) & \text{si } a = b^k \\ \Theta(n^{log_b(a)}) & \text{si } a > b^k \end{cases}$$

Ayuda: Renombrar la variable con $n = b^i n_0$

11. Dadas las siguientes definiciones en seudocódigo de exp_1 y exp_2, calcular el trabajo de cada una de ellas y determinar qué función es más eficiente.

$$\begin{array}{lcl} exp_1\ 0 & = & 1 \\ exp_1\ (n+1) & = & 2\ (exp_1\ n) \end{array}$$

$$\begin{array}{lcl} exp_2\ 0 & = & 1 \\ exp_2\ n & = & case\ par\ n\ of \\ & & \quad true \to (exp_2\ (n \div 2))^2 \\ & & \quad false \to 2\ (exp_2\ (n-1)) \end{array}$$

12. Dados los siguientes seudocódigos que implementan distintos algoritmos para invertir los elementos de una lista, calcular el trabajo de $reverse_1$ y $reverse_2$ y determinar qué función es más eficiente.

$$\begin{array}{lcl} reverse_1 & : & [a] \to [a] \\ reverse_1\ [] & = & [] \\ reverse_1\ (x \triangleleft xs) & = & (reverse_1\ xs)\ @\ [x] \end{array}$$

$$\begin{array}{lcl} []\ @\ ys & = & ys \\ (x \triangleleft xs)\ @\ ys & = & x \triangleleft (xs\ @\ ys) \end{array}$$

$$\begin{array}{lcl} revStack & : & [a] \to [a] \\ revStack\ []\ ys & = & ys \\ revStack\ (x \triangleleft xs)\ ys & = & revStack\ xs\ (x \triangleleft ys) \end{array}$$

$$\begin{array}{lcl} reverse_2 & : & [a] \to [a] \\ reverse_2\ xs & = & revStack\ xs\ [] \end{array}$$

13. Dado el siguiente seudocódigo de un algoritmo que construye un árbol binario a partir de una lista:

$$data\ Tree\ a\ =\ Empty\ \mid\ Leaf\ a\ \mid\ Node\ (Tree\ a)\ (Tree\ a)$$

$$\begin{array}{lcl} split & : & [a] \to [a] \times [a] \\ split\ [] & = & ([], []) \\ split\ [x] & = & ([x], []) \\ split\ (x \triangleleft y \triangleleft xs) & = & let\ \ (ys, zs)\ =\ split\ xs \\ & & in\ \ (x \triangleleft ys, y \triangleleft zs) \end{array}$$

$$\begin{array}{lcl} toTree & : & [a] \to Tree\ a \\ toTree\ [] & = & Empty \\ toTree\ [x] & = & Leaf\ x \\ toTree\ (x \triangleleft y \triangleleft xs) & = & let\ \ (ys, zs)\ = split\ (x \triangleleft y \triangleleft xs) \\ & & \qquad (t1, t2)\ = toTree\ ys\ ||\ toTree\ zs \\ & & in\ \ Node\ t1\ t2 \end{array}$$

a) Expresar las recurrencias correspondientes al trabajo y a la profundidad de la función $toTree$, asumiendo que $W_{split}(n) = S_{split}(n) = n$, siendo n la longitud de la lista que recibe.

b) Resolver las recurrencias encontradas en el apartado anterior usando la técnica de ecuación característica. Expresar las soluciónes utilizando la notación Θ.