Lógica para ciencias de la computación Periodo: 2020-1
Taller: Recursión sobre números y árboles Profesor: E. Andrade

---

Sean n, m números naturales. Defina las funciones $2\times[n]$, Pred$[n]$ y $m-[n]$ de la siguiente manera:

DEF $2\times[n]$:
  SI $n == 0$:
    RETORNAR 0
  SI NO:
    RETORNAR $2 + 2\times[n-1]$

DEF PRED$[n]$:
  SI $n == 0$:
    RETORNAR 0
  SI NO:
    RETORNAR $n-1$

DEF $m-[n]$:
  SI $n == 0$:
    RETORNAR m
  SI NO:
    RETORNAR PRED$[m-[\text{PRED}[n]]]$

EJERCICIO 1.

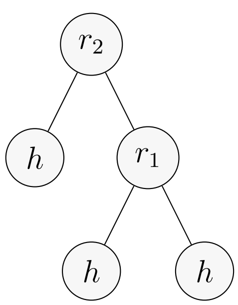
a) Escriba el paso a paso de $2\times[3]$.

b) Escriba el paso a paso de $3-[2]$ y de $2-[3]$.

c) Demuestre por inducción sobre n que $2\times[n]$ devuelve el número $2n$.

d) Suponga que m es un número natural arbitrario. Demuestre por inducción sobre n que $m-[n]$ devuelve el número $\max\{0, m-n\}$.

Una manera sencilla de escribir un árbol mediante la estructura TREE(LEFT, RIGHT) es primero escribir sus subárboles y luego usarlos para construir el árbol de interés. Considere el siguiente árbol y su representación mediante la estructura TREE(LEFT, RIGHT):



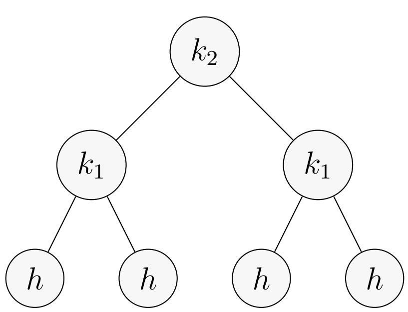
h = TREE(NULL, NULL)

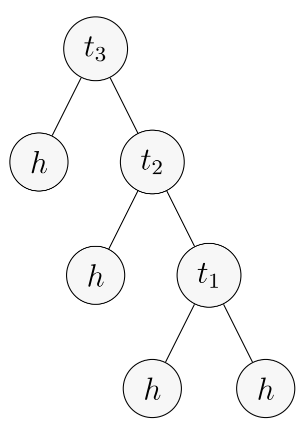
r_1 = TREE(h, h)

r_2 = TREE(h, r_1)

Observe que, por ejemplo, r_2.RIGHT=r_1 y que r_1.LEFT=h

EJERCICIO 2: Utilice la estructura TREE(LEFT, RIGHT) para representar los siguientes árboles según el modelo dado en el ejemplo anterior:





Una función recursiva para contar el número de aristas de un árbol es la siguiente:

DEF NUM_ARISTAS(A):
    SI A.RIGHT == NULL:
        RETORNAR 0
    SI NO:
        RETORNAR 2 + NUM_ARISTAS(A.LEFT) + NUM_ARISTAS(A.RIGHT)

El paso a paso de aplicar esta función al árbol r_2 es:

$$
\begin{aligned}
\text{NUM\_ARISTAS}(r_2) &= 2 + \text{NUM\_ARISTAS}(r_2.\text{LEFT}) + \text{NUM\_ARISTAS}(r_2.\text{RIGHT}) \\
&= 2 + \text{NUM\_ARISTAS}(h) + \text{NUM\_ARISTAS}(r_1) \\
&= 2 + 0 + \text{NUM\_ARISTAS}(r_1) \\
&= 2 + 0 + \big(2 + \text{NUM\_ARISTAS}(r_1.\text{LEFT}) + \text{NUM\_ARISTAS}(r_1.\text{RIGHT})\big) \\
&= 2 + 0 + \big(2 + \text{NUM\_ARISTAS}(h) + \text{NUM\_ARISTAS}(h)\big) \\
&= 2 + 0 + (2 + 0 + 0) = 4
\end{aligned}
$$

EJERCICIO 3: Presente el paso a paso de NUM_ARISTAS para los árboles k_2 y t_3.

Una función recursiva para determinar la altura de un árbol es la siguiente:

DEF ALTURA(A):
    SI A.RIGHT == NULL:
        RETORNAR 0
    SI NO:
        RETORNAR 1 + MAX{ALTURA(A.LEFT), ALTURA(A.RIGHT)}

EJERCICIO 4: Presente el paso a paso de ALTURA para los árboles r_2, k_2 y t_3.

EJERCICIO 5: Defina una función recursiva NUM_NODOS que encuentre el número de nodos de un árbol y presente el paso a paso de NUM_NODOS para cada uno de los árboles del ejercicio 2.

EJERCICIO 6: Demuestre mediante inducción estructural que, para cualquier árbol binario A:

$$\text{NUM\_NODOS}(A) = \text{NUM\_ARISTAS}(A) + 1$$

EJERCICIO 7: Sea A un árbol binario y a=ALTURA(A). Demuestre por inducción estructural que:

$$\text{NUM\_NODOS}(A) \leq 2^{a+1} - 1$$