Lógica para ciencias de la computación Periodo: 2020-1
Taller Nash Primer Parcial Profesores: E. Andrade y D. Bojacá

EJERCICIO 1: Sea n un número natural y considere el siguiente código en Python que define la función recursiva $F(n)$:

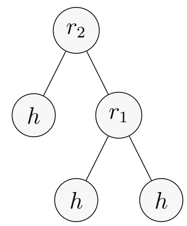
```
Def F(n):
    if n == 0:
        return 1
    else:
        return 2^n + F(n-1)
```

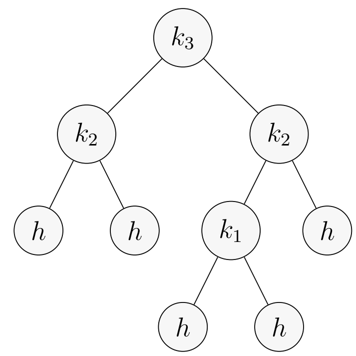
a) Escriba el paso a paso de $F(3)$.

b) Demuestre que $F(n) = 2^{n+1} - 1$

EJERCICIO 2: Sea A un árbol binario.

a) Defina de manera recursiva la función $Corta[A]$, la cual cuenta el número de aristas de la rama más corta de A. Observe que, por ejemplo, $Corta[r_2] = 1$ y $Corta[k_3] = 2$.





b) Presente el paso a paso de esta función sobre el árbol r_2.

c) Asuma la definición recursiva de la función $Num\_Aristas(A)$, la cual cuenta el número total de aristas de A. Demuestre por inducción estructural que

$$Corta(A) \leq \frac{Num\_Aristas(A)}{2}$$

EJERCICIO 3: Sea A una fórmula representada como un árbol y asuma la definición de las siguientes funciones:

- $C\_Bin(A)$: Número de ocurrencias de conectivos binarios en A.
- $Negs(A)$: Número de ocurrencias de negaciones en A.
- $Atom(A)$: Número de ocurrencias de letras proposicionales en A.
- $Inorder(A)$: Cadena que representa la notación inorder de A.
- $Len(c)$: Cantidad de símbolos en la cadena c.

Demuestre por inducción estructural que:

$$Len(Inorder(A)) = 3 * C\_Bin(A) + Negs(A) + Atom(A)$$

EJERCICIO 4: Sea $I = \{‘p’ \colon 1, ‘q’ \colon 0\}$. Escriba el paso a paso de $V_I(A_4)$ donde:

$$A_0 = \textsc{Tree}(p, \textsc{null}, \textsc{null})$$
$$A_1 = \textsc{Tree}(q, \textsc{null}, \textsc{null})$$
$$A_2 = \textsc{Tree}(\neg, \textsc{null}, A_1)$$
$$A_3 = \textsc{Tree}(\wedge, A_2, A_0)$$
$$A_4 = \textsc{Tree}(\rightarrow, A_0, A_3)$$