

SEGUNDO PARCIAL

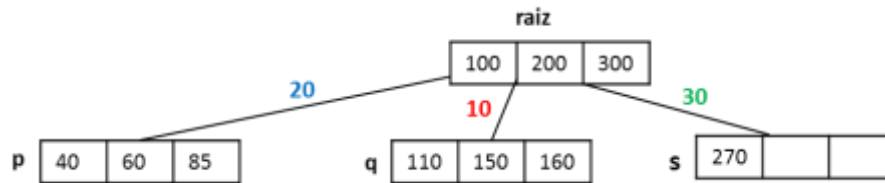
INF310 SX- Estructuras de Datos II. Gestión 1-2021.

Subgrupo: A-L

Árbol M-Vías pesado

1. Para que un árbol M-Vías se convierta en *pesado*, basta asignar un peso > 0 a cada uno de los punteros no-nulos. El peso del puntero que sostiene a un Nodo, se almacena en el mismo Nodo (La Raíz excepcionalmente tiene su peso=0, porque nadie la sostiene).

Por ejemplo



```
raiz.getPeso()=0    //La raíz siempre tiene costo=0
p.getPeso()=20
q.getPeso()=10
s.getPeso()=30
```

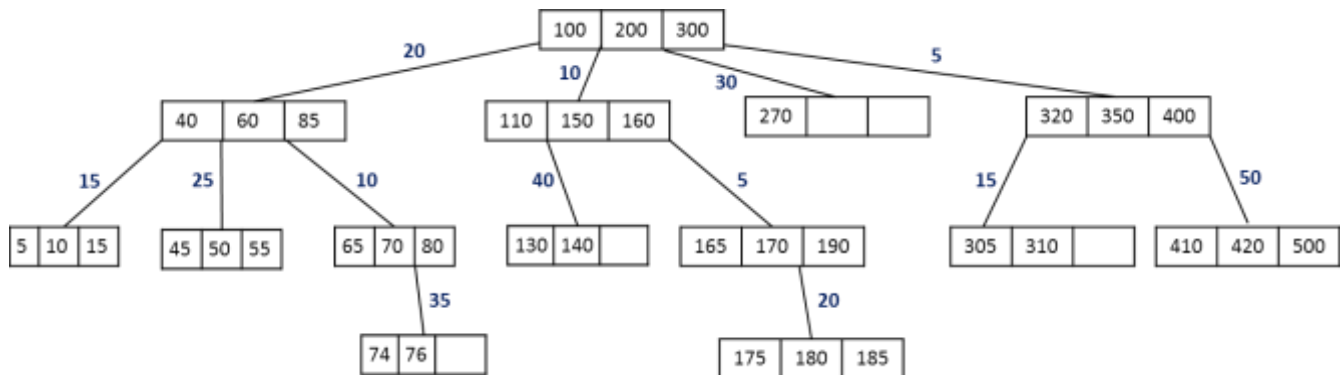
//En la consola un nodo se muestra con: el peso, un guión y luego los datos. Por ejemplo, el nodo p se verá 20- [40 | 60 | 85].

Se define el **costo** de un data x, como la suma de los pesos de los punteros que se deben recorrer desde la raíz hasta el nodo que contiene a x. Sabiendo esto, escriba la función

```
public boolean hasCost(int h, int cost)
```

la cual, **usando** una función **máscara RECURSIVA**, devuelve true si el data **h** está en una hoja y tiene el costo **cost**. Devolverá false si: h no existe o h no está en un nodo hoja o el costo del nodo hoja donde está h no es **cost**.

Por ejemplo: Dado el árbol A



```
A.hasCost(600, 100)= false    //Porque el 600 no existe.
A.hasCost(190, 15) = false    //El 190 existe, pero no está en un nodo hoja.
A.hasCost(140, 80) = false    //El 140 existe y está en una hoja, pero su costo no es 80 (el costo de 140 es 10+40=50)
A.hasCost(180, 35) = true     //El 180 existe y está en una hoja y su costo es 10+5+20=35
A.hasCost(74, 65) = true     //El 74 existe y está en una hoja y su costo es 20+10+35=65
```

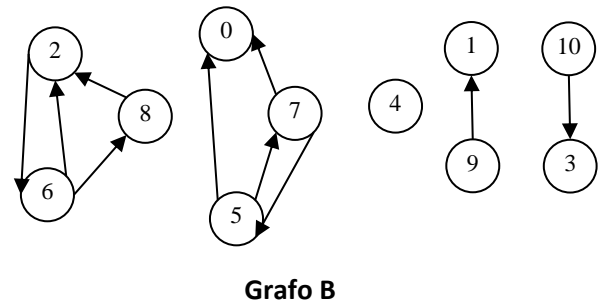
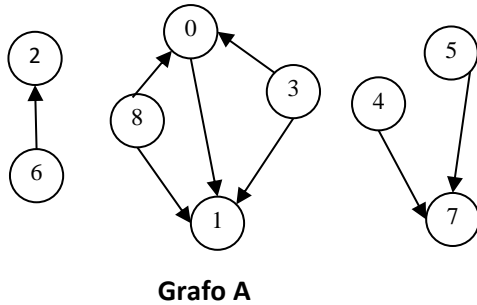
Grafos Dirigidos

2. En la class Grafo, escriba la función

```
public int edges(int u)
```

la cual devuelva la **cantidad de aristas** que tiene la isla que **contiene** al vértice u.

Por ejemplo:



- El Grafo A tiene tres islas: (2, 6), (0, 8, 3, 1) y (5, 4, 7). Así,

A.edges(2) = 1 //Porque el vértice 2 está en la primera isla (2,6) y ésta tiene una arista.

A.edges(0) = 5 //Porque el vértice 0 está en la isla(0, 8, 3, 1) y ésta tiene 5 aristas.

A.edges(5) = 2 //Porque el vértice 5 está en la isla y (5, 4, 7) y ésta isla tiene 2 aristas.

A.edges(7) = 2 //Porque el vértice 7 está en la isla y (5, 4, 7) y ésta isla tiene 2 aristas.

- El Grafo B tiene cinco islas:

B.edges(6) = 4 //Porque el vértice 6 está en la isla (2, 6, 8) y ésta isla tiene 4 aristas.

B.edges(4) = 0 //Porque el vértice 4 está en la isla (4) y ésta isla no tiene aristas.