**Ejercicio 6:**

Responder V o F y justificar su respuesta:

* 1. \_\_\_ En un AVL el penúltimo nivel tiene que estar completo
  2. \_\_\_ Un AVL donde todos los nodos tengan factor de balance 0 es completo
  3. \_\_\_ En la inserción en un AVL, si al actualizarle el factor de balance al padre del nodo insertado éste no se desbalanceó, entonces no hay que seguir verificando hacia arriba porque no hay cambios en los factores de balance.
  4. \_\_\_ En todo AVL existe al menos un nodo con factor de balance 0.

**Ejercicio 7:**

Sean *A* y *B* dos *AVL* de *m* y *n* nodos respectivamente y sea *x* un key cualquiera de forma tal que para todo key *a* ∈ *A* y para todo key *b* ∈ *B* se cumple que *a* < *x* < *b*. Plantear un algoritmo *O*(log *n* + log *m*) que devuelva un *AVL* que contenga los key de *A*, el key *x* y los key de *B*.

1. Se busca el nodo en A con la llave más grande (mayor) que x y se guarda en una variable llamada maxA.
2. Se busca el nodo en B con la llave más pequeña (menor) que x y se guarda en una variable llamada minB.
3. Se crea un nuevo AVL llamado C con raíz en x.
4. Si maxA existe, se agrega el subárbol izquierdo de maxA como subárbol izquierdo de la raíz de C.
5. Si minB existe, se agrega el subárbol derecho de minB como subárbol derecho de la raíz de C.
6. Se balancea el AVL C y se devuelve como resultado.

El algoritmo recorre cada árbol solo una vez, y la búsqueda del nodo máximo y mínimo se puede hacer en tiempo logarítmico en cada árbol, por lo que el tiempo de ejecución es O(log n + log m).

Principio del formulario