

**Instituto Tecnológico de Tijuana**

**Subdirección Académica**

**Departamento de Sistemas y Computación**

**Semestre agosto-diciembre 2018**

**Carrera:** Ing. En Sistemas Computacionales Serie SC1A

**Materia:** Estructura de Datos

**Unidad 3 – Arboles Balanceados**

**Alumno:**15211883 - Angeles Valadez Jonathan

**Profesor:** Ray Brunett Parra Galaviz

**Árbol Balanceado**

Es un árbol binario de búsqueda en el cual se cumple: “Para todo nodo T del árbol, la altura de los sub-árboles izquierdo y derecho no debe diferir en más de una unidad”. Son llamados árboles AVL en honor a sus inventores G.M. Adelson Velskii y E.M. Landis. La idea central de éstos es la de realizar reacomodos o balanceos, después de inserciones o eliminaciones de elementos. Los árboles balanceados se parecen mucho, en su mecanismo de formación, a los números de Fibonacci. El árbol de altura 0n es vacío, el árbol de altura 1 tiene un único nodo y en general el número de nodos del árbol con altura h>1 se calcula aplicando la siguiente fórmula recursiva:

arbolesbalanceados.png

Donde k = número mínimo de nodos de árbol y h = altura

**Inserción en árboles balanceados**

1. Las ramas izquierdas (RI) y derecha (RD) del árbol tienen la misma altura (HRI=HRD), por lo tanto:

1. Si se inserta un elemento en RI entonces HRI será mayor a HRD
2. Si se inserta un elemento en RD entonces HRD será mayor a HRI No se altera el equilibrio del árbol

2. Las ramas izquierda (RI) y derecha (RD) del árbol tienen altura diferente (HRI!=HRD):

1. Supóngase que HRIHRD: Si se inserta un elemento en RI, entonces se rompe el criterio de equilibrio del árbol y es necesario reestructurarlo.
2. Si se inserta un elemento en RD entonces HRD será igual HRI. Se mejora el equilibrio del árbol.

**Borrado en árboles balanceados**

1. Localizar la posición del nodo a eliminar en el árbol.

2. Se elimina siguiendo los siguientes criterios:

1. Si el elemento a borrar es terminal u hoja, simplemente se suprime.
2. Si el elemento a borrar tiene un solo descendiente entonces, tiene que sustituirse por ese descendiente.
3. Si el elemento a borrar tiene los dos descendientes, entonces tiene que sustituirse por el nodo que se encuentra más a la izquierda en el subárbol derecho o por el nodo que se encuentra más a la derecha en el subárbol izquierdo.

3. Se regresa por el camino de búsqueda calculando el FE de los nodos visitados. Si en alguno de los nodos se viola el criterio de equilibrio, entonces debe reestructurarse el árbol. El proceso termina cuando se llega hasta la raíz del árbol.

**Nota:** Cabe aclarar que mientras que en el algoritmo de inserción una vez que era efectuada una rotación podía detenerse el proceso, en este algoritmo debe continuarse puesto que se puede producir más de una rotación en el camino hacia atrás.

**Bibliografía**

[En Línea]:<http://mapaches.itz.edu.mx/~mbarajas/edinf/avl.pdf>