

Tablas hash

- 1) Direccionamiento directo
- 2) FUNCIÓN hash
- 3) Hashing uniforme $a = n/m$
- 4) Funcon hashing: División $k \bmod m$. Multiplicación piso($m \cdot (A \cdot k \bmod 1)$)

Arboles binarios de búsqueda

- 1) Recorrido inorden (Lista ordenada)
- 2) Hijos izquierdo son menores o iguales que el nodo
- 3) Hijos derechos son mayores que el nodo
- 4) Inserción X (Verifique si x es mayor, buscamos a la derecha, en otro caso a la izquierda)
- 5) Eliminación. Caso 1) x no tiene hijos, se borra. Caso 2) x solo tiene un hijo, reemplazamos x por el hijo. Caso 3) x tiene dos hijos, se reemplaza con el sucesor y se eliminar el sucesor

Las operaciones toman $O(h)$

Desventajas: Si no está nivelado las operaciones toman $O(n)$

Ventajas: Si está nivelado las operaciones toman $O(\log(n))$

Arboles rojinegros

- 1) Todo nodo es rojo o negro
- 2) La raíz es negra
- 3) El tamaño del camino negro de cualquier nodo debe ser igual
- 4) Si x es roja, sus hijos deben ser negros
- 5) Toda hoja nil es negra (Eliminación)

Se garantiza que la altura negra es $2\lg(n+1)$, por lo tanto las operaciones son $O(\log(n))$ a diferencia e los arboles binarios de búsqueda.

Para insertar, no es suficiente el algoritmo de los arboles binarios busqueda, por lo que agregamos dos operaciones. Rotación a la derecha y rotación a la izquierda.