B-trees

A *B-tree* of order m is a multi-way tree with the following properties.

1) The root has at least two children unless it is a leaf.
2) No node in the tree has more than m children.
3) Every node except for the root and the leaves has at least $\lceil m/2 \rceil$ children.

B-trees Definition

4) All leaves appear on the same level and contain no information.
5) An internal node with *k* children contains exactly (*k*-1) keys and they are kept sorted.

- The special case where *m* = 3 is called a 2-3 tree. Hence a **2-3 tree** is a B-tree of order 3.

B-trees…

- What can we say about search times in B-trees of order m? In the worst case the deepest B-tree on n keys will be constructed by taking the minimum number of children allowed for each node. Thus the root will have 2 children and each node on the levels below the root will have $\lceil m/2 \rceil$ children.

B-trees…

- So the number of nodes on levels 1,2,3… follow the geometric progression 2, $\lceil m/2 \rceil$, $2\lceil m/2 \rceil^2$, … If the tree has k levels and all the leaves appear on level k, then the number of leaves is just one more than the number of keys (why?). Since there are n keys, there are (n+1) leaves.

B-trees…

- We get for the deepest tree

$$\#of\ leaves = n+1 \geq 2\lceil m/2 \rceil^{k-1}$$

Solving for k yields

$$(n+1)/2 \geq \lceil m/2 \rceil^{k-1}$$

$$\log_{\lceil m/2 \rceil}((n+1)/2) \geq k-1$$

or

$$k \leq 1 + \log_{\lceil m/2 \rceil}((n+1)/2)$$

B-trees…

- Since we need to access at most k levels in the worst case, we see that the number of levels is quite modest. For instance, k is at most 3 if $n = 1{,}999{,}998$ and $m = 199$