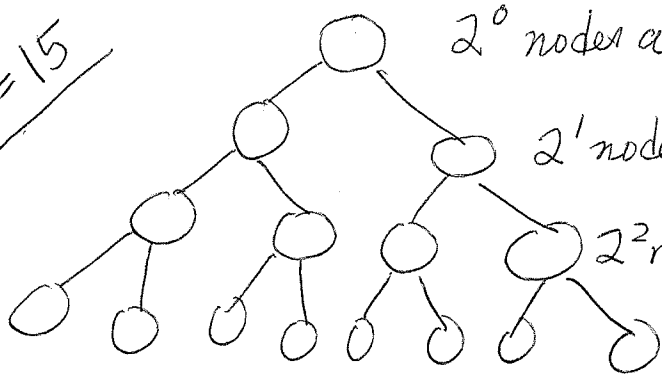
Binary Tree (Perfect Triangle)



B = 3

2^k nodes at level k.

$$N = \text{total \# of nodes} = \sum_{k=0}^{B} 2^k = 2^{(B+1)} - 1$$

For the above example $2^{(3+1)} - 1 = 15$

How many levels? B+1

$$2^{(B+1)} - 1 = N$$

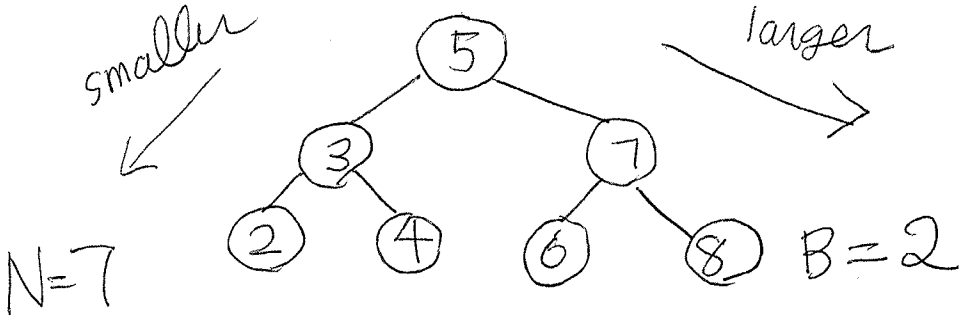
$$2^{(B+1)} = N+1$$

$$\boxed{B+1 = \log_2(N+1) \text{ levels}}$$

= # comparisons

(2)

Binary Decision Tree



To look for 2, 3 comparisons
To look for 6, 3 comparisons
To look for 5, 1 comparison
To look for 10, 3 comparisons & off the tree

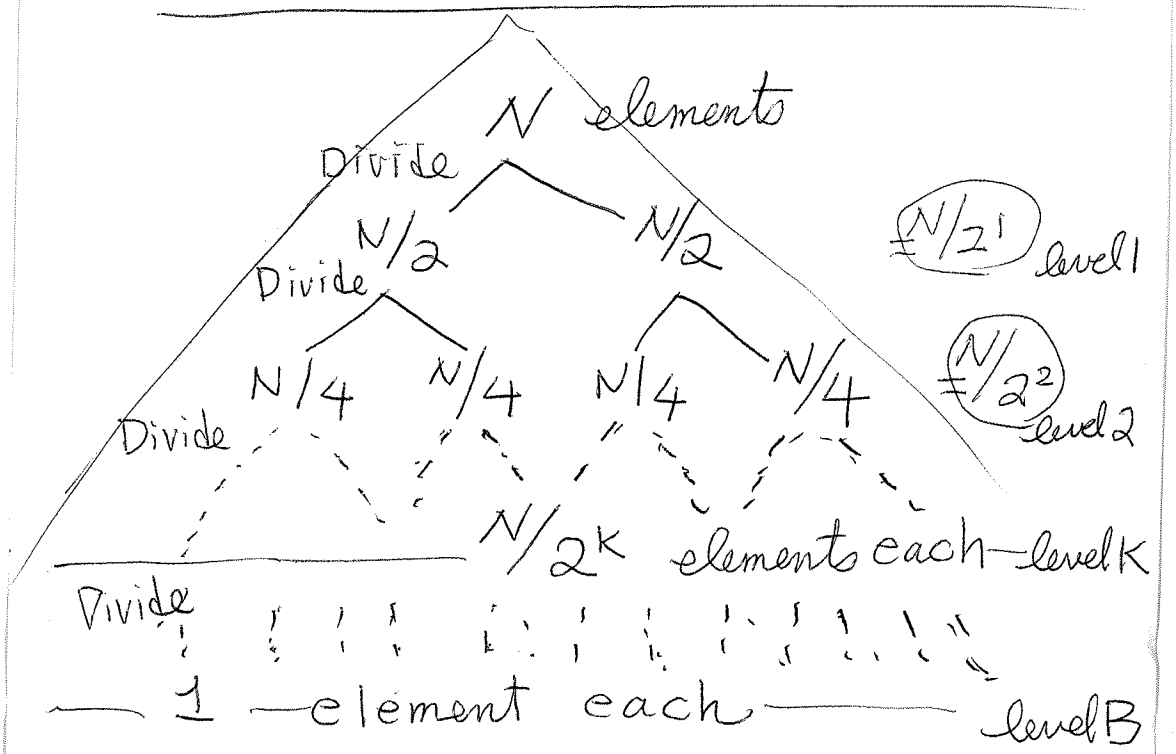
$$\boxed{B+1 = \log_2(N+1)}$$

$$= \log_2 8 = 3$$

= max number of comparisons

The best possible way to search through an ordered list.

(3) Recursion Tree



At level B $\frac{N}{2^B} = 1$ element each

$N = 2^B$

$\boxed{B = \log_2 N}$

① If $N = 8$, $B = 3$.

$\Rightarrow$ 4 levels in the tree $(B+1)$

$\Rightarrow$ List was divided in halves via 3 levels (B) of recursion.

② If $N = 1024$, $B = 10$. 11 levels in the tree