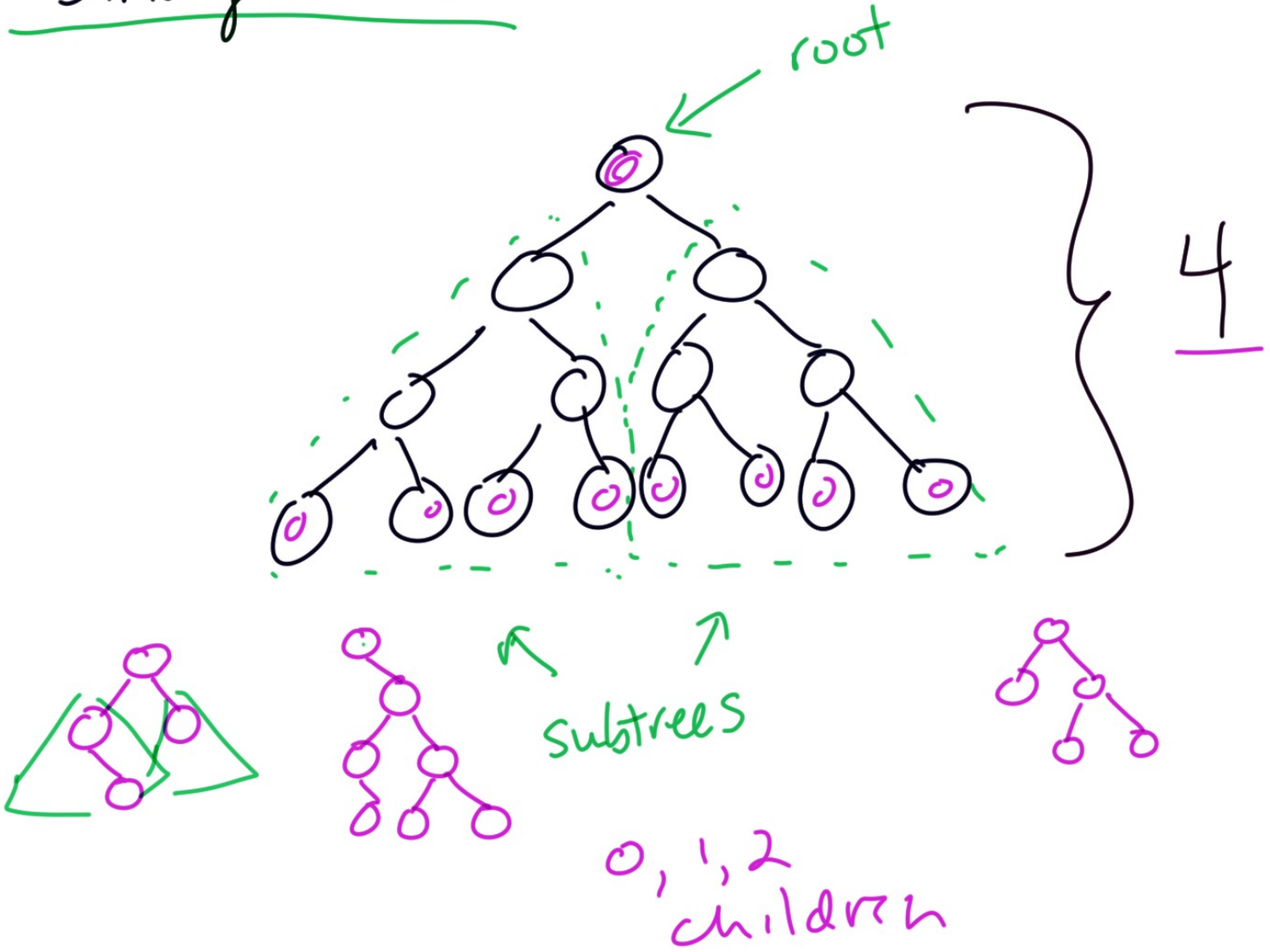
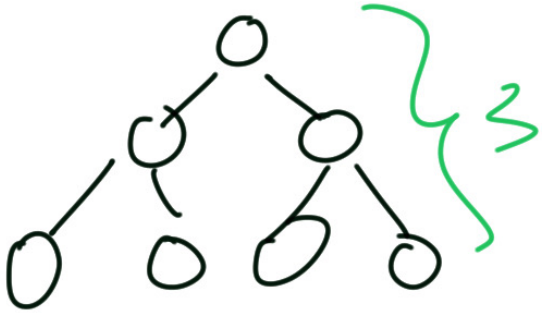
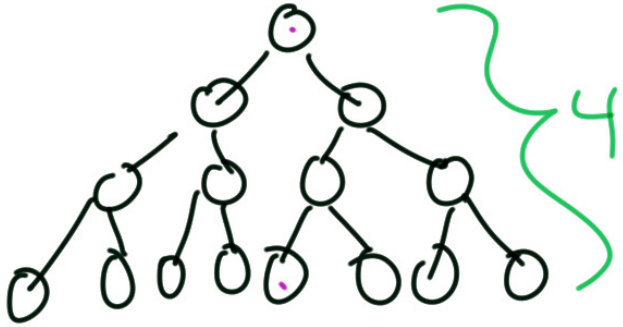
Binary Trees



Tree	Height	# of Nodes
	1	1
	2	3 = 4-1
	3	7 = 8-1
	4	$2^h - 1$ 15 = 16-1
	5	$2^h - 1$ 31 = 32-1

Height of Binary Tree and Number of Nodes

For a binary tree with nodes that either have 0 or 2 children and all leaves on the same level.

- The number of nodes, n, is a function of height, h, such that: $\mathbf{n = 2^h - 1}$
- Thus n elements can possibly be stored in a binary tree of height $\mathbf{\log_2(n+1)}$
- So the longest path from root to any leaf would be $\mathbf{\log_2(n+1)}$

Some real world applications of binary trees

- Search: objects such as *map* and *set*
- Videogames: determining – depending on the players viewpoint -- what objects to render
- Networking: managing router-tables