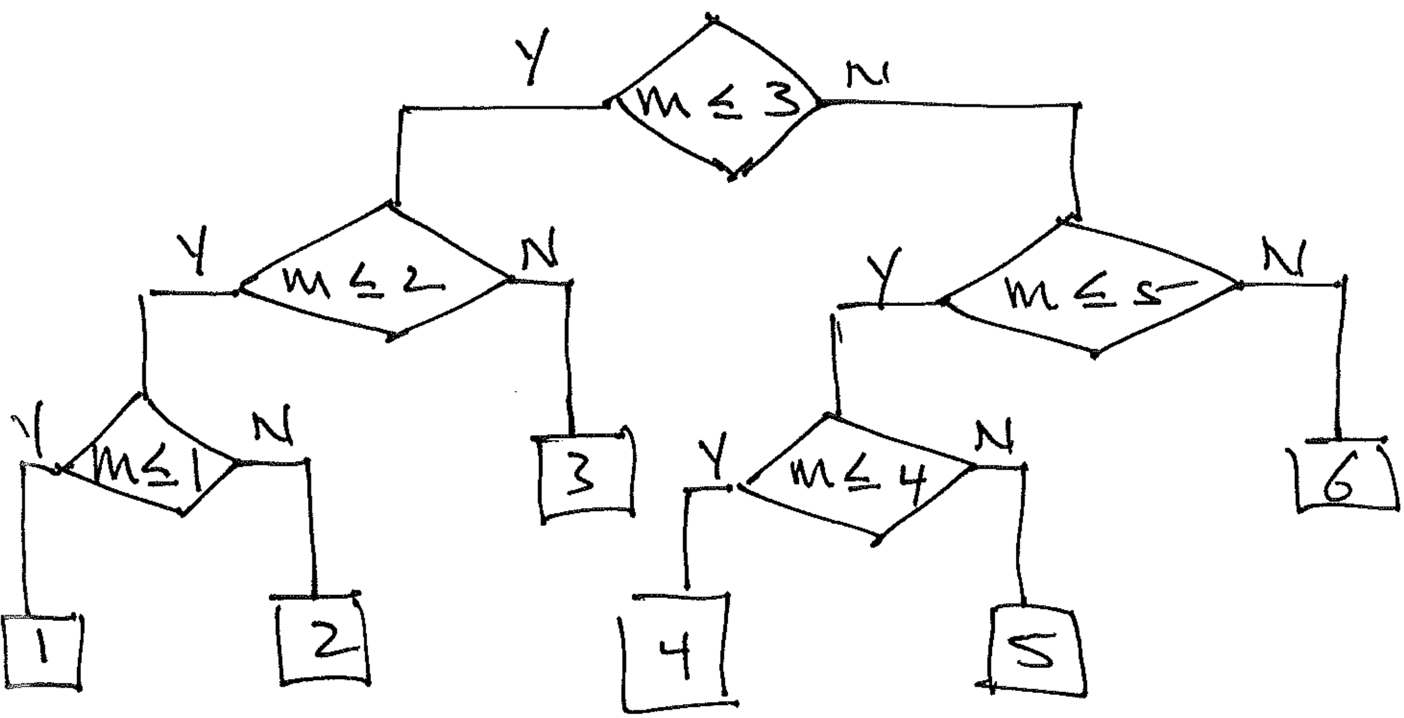
CMSC 201 5-12-10

Decision Trees:

Ex. Let m be an integer in range $1 \le m \le 6$. Problem: determine value of m by asking a seq. of yes/no questions.

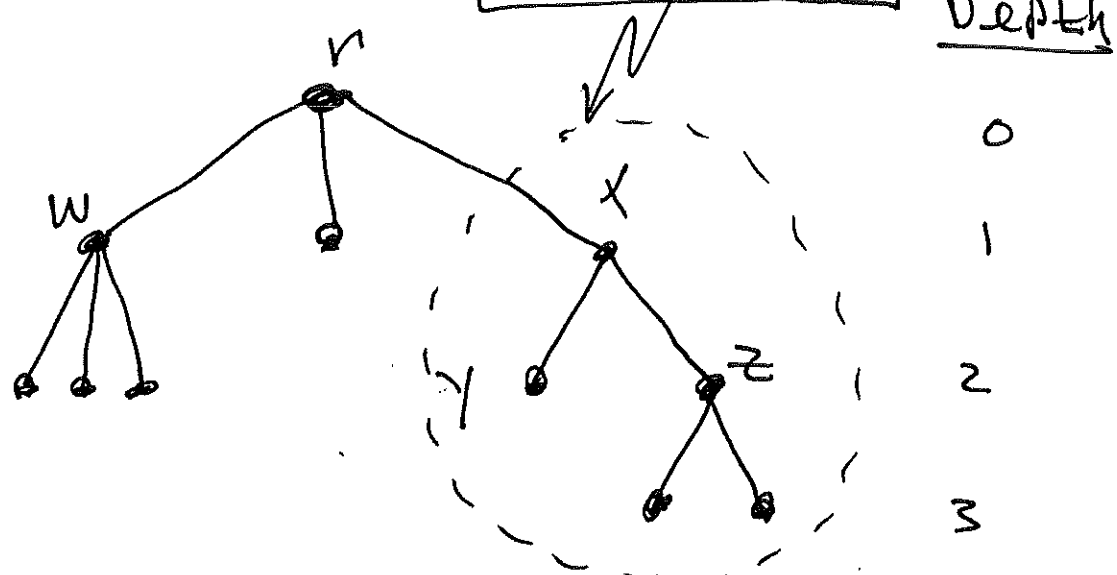


worst case # Questions = 3

avg. case # Questions $= \dfrac{3+3+2+3+3+2}{6} = 2.67\ldots$

Defn. A Rooted Tree is A Tree with a distinguished vertex (node) called root.

the depth of a node is its distance from the root.



the Parent of a node is the unique adjacent node whose depth is 1 less.

root has no Parent.

A leaf is a node with no Children.

An internal node is a non-leaf.

The height of a rooted tree is the maximum node depth in the Tree, i.e. depth of deepest leaf.

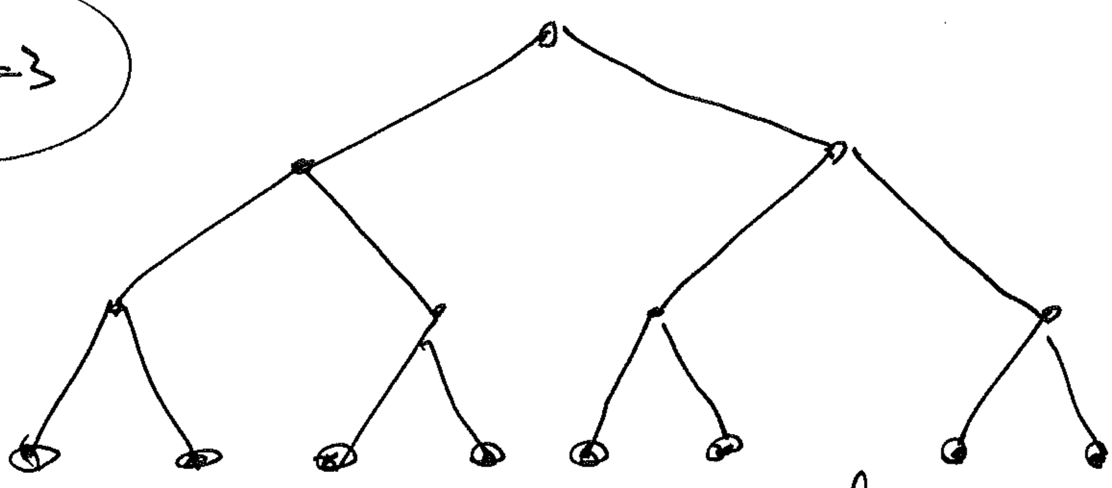
The height of a node x is the height of the subtree rooted at x, i.e. length of a longest descending path starting at x.

Defn: A Binary Tree is a rooted tree in which each node has at most 2 children.

A <u>k-ary tree</u> is a rooted tree in which each node has at most k children.

A <u>complete Binary Tree</u> (CBT) is a B.T. in which all leaves are at same depth, and internal nodes have 2 children

h=3



Depth	#nodes
0	1
1	2
2	4
3	8

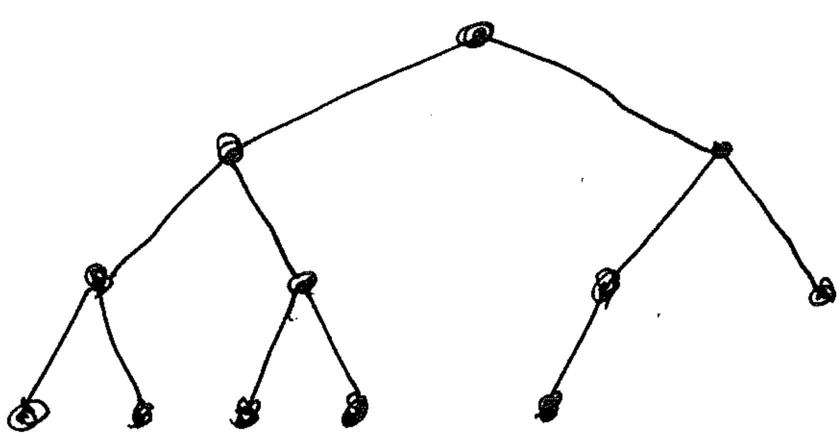
#nodes at depth d = 2^d

leaves = #nodes at depth h = 2^h

∴ the height h of a CBT with n leaves satisfies:

$$n = 2^h \quad \text{and} \quad h = \lg(n).$$

An <u>Almost Complete Binary Tree</u> (ACBT) is a Binary tree that is filled at each level, except possibly last, which is filled from Left to right.



<u>Exercise</u>: An ACBT with n leaves, height h satisfies $\lceil \lg n \rceil \le h \le \lceil \lg n \rceil + 1$

__Theorem__

The height h of __any__ Binary Tree with n leaves satisfies

$$h \geq \lceil \lg n \rceil$$

__Notation__ Let $L(T)$ and $H(T)$ denote #leaves & height of a B.T. T.

__Proof:__
Let T be a B.T. with $h = H(T)$ and $n = L(T)$. We show by induction on h that

$$h \geq \lceil \lg n \rceil .$$

I. base.

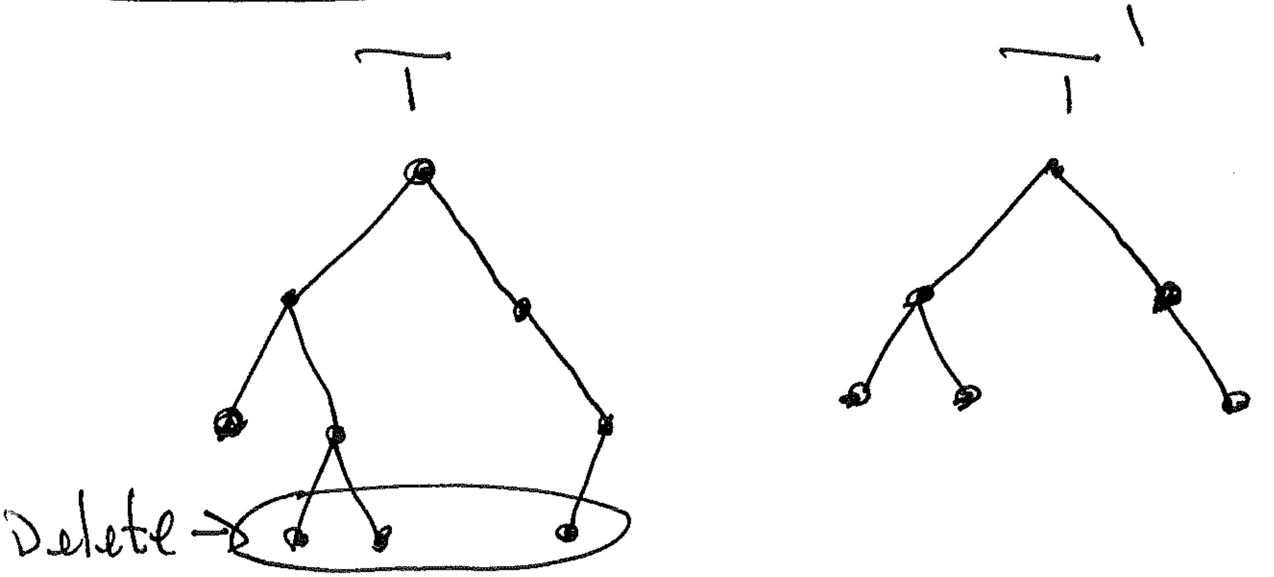
If $h=0$ then T contains just 1 node, the root, which is the sole leaf, so $n=1$. In this case $h \geq \lceil \lg n \rceil$ reduces to $0 \geq 0$, which is true,

II. b. Let $h>0$ and assume the result holds for all trees of height $h-1$. i.e. if S is a B.T. with $H(S)=h-1$, then

$$H(S) \geq \lceil \lg(L(S)) \rceil .$$

Let T' be the B.T. obtained from T by deleting all leaves at depth h.

Illustration:



Now $H(T') = h-1$, so by the Ind. hyp. we have

$$H(T') \geq \lceil \lg L(T') \rceil$$

Also since each node has at most 2 children, we have

$$L(T) \leq 2\,L(T')$$

So $L(T') \geq \frac{L(T)}{2}$. $\quad *$

Thus

$$
\begin{aligned}
h-1 &= H(T') \\
&\geq \lceil \lg L(T') \rceil && \text{ind. hyp.} \\
&\geq \left\lceil \lg \left(\frac{L(T)}{2} \right) \right\rceil && \text{by } * \\
&= \lceil \lg L(T) - 1 \rceil \\
&= \lceil \lg L(T) \rceil - 1 && \text{since } \lceil x-1 \rceil = \lceil x \rceil - 1 \\
&= \lceil \lg n \rceil - 1
\end{aligned}
$$

$\therefore \quad h \geq \lceil \lg n \rceil$.

Result follows for all h by Induction

///.

Thm

The height h of any k-ary tree with n leaves satisfies

$$h \geq \lceil \log_k n \rceil .$$

HW7: Prove by ind on h.

A Decision Tree is a Representation of operation of an algorithm on All possible inputs of a fixed size.

Each internal node represents a test (or probe) of the input data. Each leaf represents a possible output or <u>verdict</u>

Each downward path from root to a leaf represents a particular logical pathway, i.e. seq. of tests which lead to a conclusion.

k = max # of outcomes to each test.

n = # leaves $\geq$ # possible verdicts.

h = height = max # of tests necessary to reach a verdict.

so from last thm we conclude:

$$\text{max \# tests} = h \geq \lceil \log_K n \rceil \geq \lceil \log_K (\text{\#verdicts}) \rceil$$

<u>Previous Example</u>: guess m in range $1 \leq m \leq 6$ by asking only yes/no questions.

Is there an algorithm which determines m in at most 2 questions?

$\text{\#verdicts} = 6 \quad (m=1, m=2, \ldots, m=6)$

$\text{\# outcomes on each test} = 2.$

$\therefore (\text{max \# questions}) \geq \lceil \log_2(6) \rceil = 3.$

This argument does not prove the existence of an algorithm that solves the problem with 3 questions. Instead we've shown non-existence of any yes/no algorithm that solves it in 2 questions.