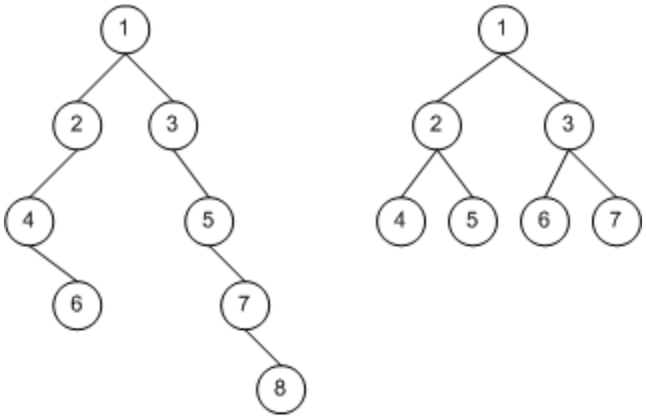
Q1



(a) (b)

CSSE 230 Day 11

Size vs height in a Binary Tree

After today, you should be able to…

… use the relationship between the size and height of a tree to find the maximum and minimum number of nodes a binary tree can have

…understand the idea of mathematical induction as a proof technique

http://i.msdn.microsoft.com/dynimg/IC71494.gif

Term project starts Day 13

Preferences for partners for the term project (groups of 3)

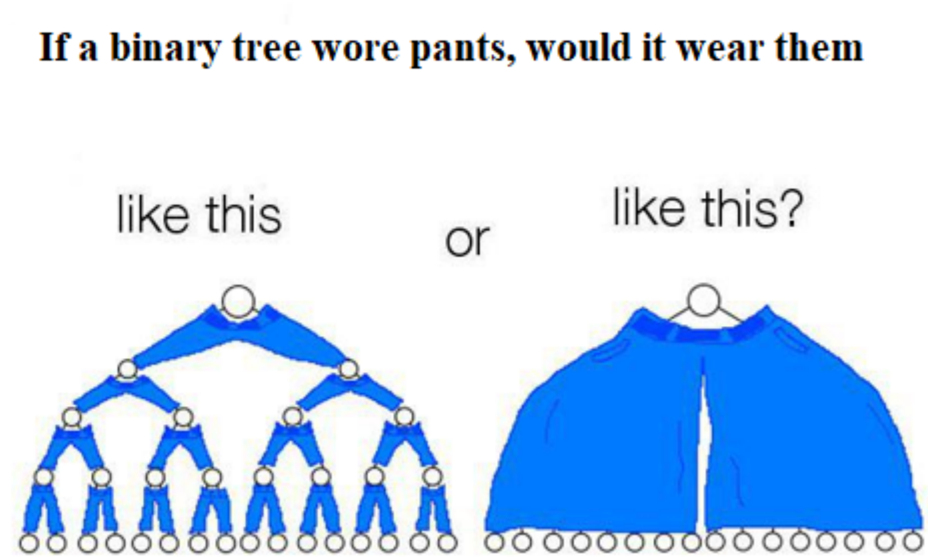
Partner preference survey on Moodle – Day 11

- Preferences balanced with experience level + work ethic
  - If course grades are close, I'll honor "prearranged teammate" preferences
  - If no "prearranged teammate", best to list several potential members
  - If you don't want to work with someone, that preference will be honored
  - Historical evidence indicates working with others in a similar current CSSE230 grade attainment level often pans out best

Some questions you might consider asking potential programming partners:

- What final grade range are you aiming for in CSSE230?
- Do you like to get it done early or to procrastinate?
- Do you prefer to work daytime, evening, late night?
- Do you normally get a lot of help on the homework?Survey is due 24 December – do it as soon as you can

Some meme humor



http://www.smosh.com/smosh-pit/memes/internets-best-reactions-if-dog-wore-pants
http://knowyourmeme.com/photos/1272773-if-a-dog-wore-pants

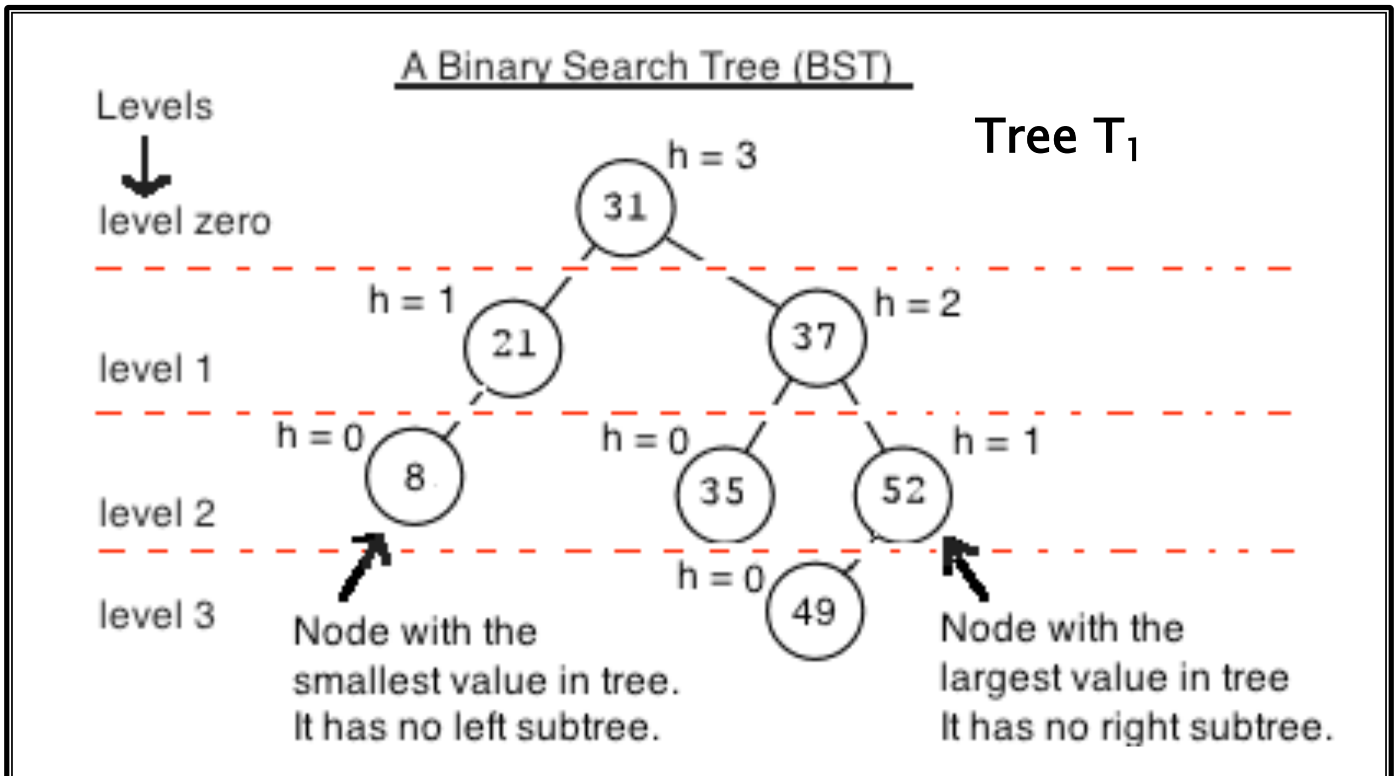
Other announcements

- Today:
  - Size vs height of trees: patterns and proofs
- Wrapping up the BST assignment, and worktime.

Tree Review

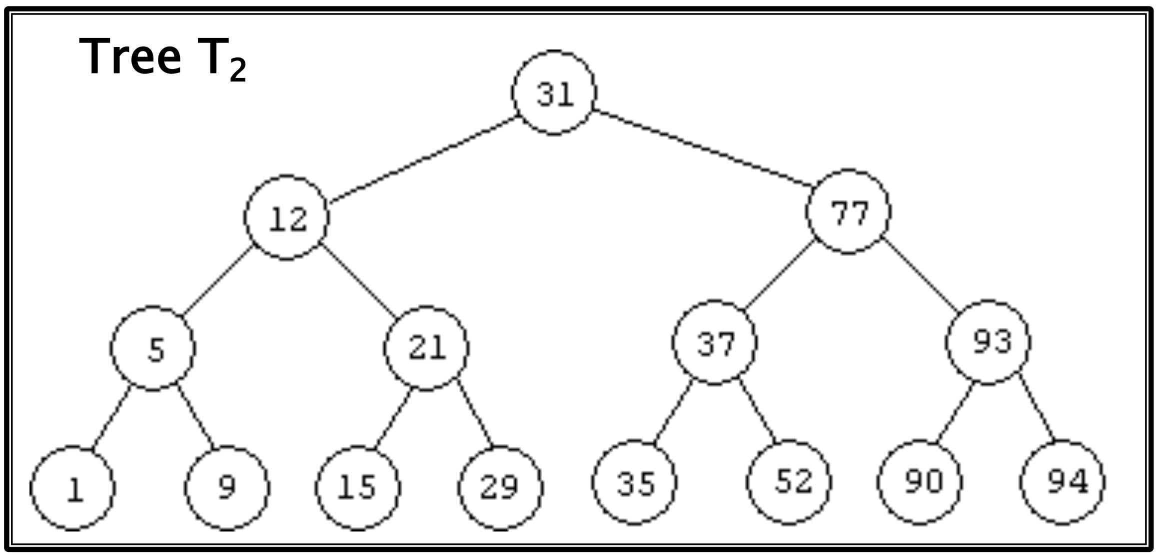
- $N(T)$ = # nodes in tree T
- $h(T)$ = height of tree T

- $N(T_1) = 7$
- $h(T_1) = 3$



Extreme Trees

- A tree with the maximum number of nodes for its height is a *full* binary tree.
- full binary tree – each node is either a leaf or has exactly two children

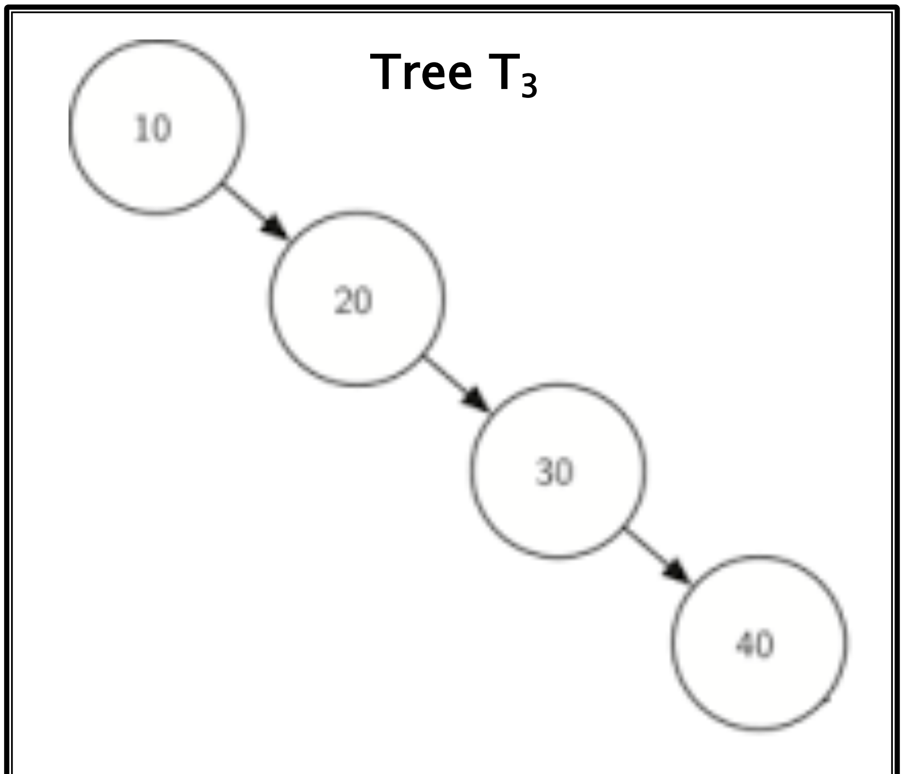


- $N(T_2) = 15$
- $h(T_2) = 3$

Now, write N(T) in terms of h(T): N(T) = ???

Extreme Trees

- A *degenerate* binary tree with the minimum number of nodes for its height
is essentially a ______ ______



- $N(T_3) = 4$
- $h(T_3) = 3$

BST Algorithms

- Height matters!
  - BST algorithms: *insertion*, *deletion*, and *search*
  - Performance of all 3 is height dependent: O(h(T))
  - What is Big-O for each when
    - Tree T is degenerate?
    - Tree T is full?

Mathematical Induction

To prove that P(n) is true for all $n \geq n_0$:

- *Basis step*: Prove that $P(n_0)$ is true (base case), and
- *Induction step*: Prove that **if P(k) is true** for any $k \geq n_0$, then P(k+1) is also true.

[This part of the proof must work for all such k!]

P(n) – propositional function, i.e., a declarative statement parameterized by *n* that is either true or false

$$(P(1) \wedge \forall k(P(k) \rightarrow P(k+1))) \rightarrow \forall n P(n)$$

DIRECT MATHEMATICAL PROOFS

def: A *direct proof* is a mathematical argument where one starts with the premises and reasons to the conclusion by using rules of inference.

Direct proofs and implication

- In this case we're trying to prove p -> q
- We need only show that *if* p is true, that q cannot be false
- The direct proof *assumes* p to be true and then shows that q cannot be false
- We don't have to show that p *is* true, because if p is false, then the implication is true no matter if q is true or false

p	**q**	**p -> q**
T	T	T
T	F	F
F	T	T
F	F	T

To prove recursive properties (on trees), we use a technique called mathematical induction

- Actually, we use a variant called *strong induction* :

The former governor of California

Q6

Strong Induction

- To prove that $p(n)$ is true for all $n \geq n_0$:
  - Prove that $p(n_0)$ is true (base case), and
  - For all $k > n_0$, prove that if we assume $p(j)$ is true for $n_0 \leq j < k$, then $p(k)$ is also true
- An analogy:
  - Regular induction uses the previous domino to knock down the next
  - Strong induction uses all the previous dominos to knock down the next!
- Warmup: prove the arithmetic series formula
- Actual: prove the formula for $N(T)$