# Math 135 - More on recurrences

Note Title 10/19/2012

## Announcements

- No class Monday
- HW due Wed.

# Recursion

Defining the $n^{th}$ term of a sequence in terms of previous terms.

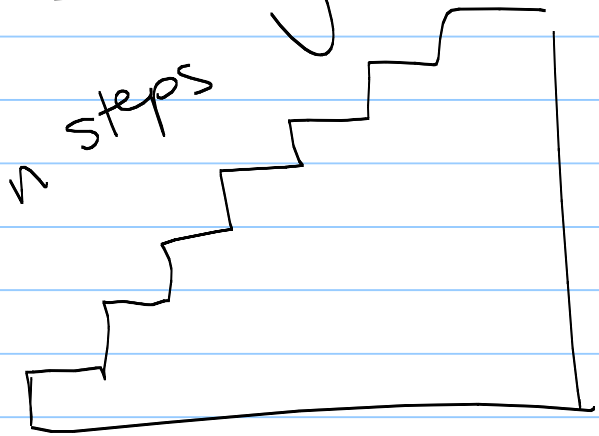
**Ex:** $F_n = F_{n-1} + F_{n-2}$

$A(n) = n \cdot A(n-1)$

$P(n) = (1.06)\, P(n-1)$

$B(n) = B\left(\frac{n}{2}\right) + 1$

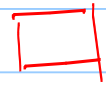
# Modeling recursion: stair climbing



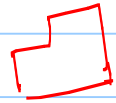

If I can take 1 or 2 stairs at a time, how many ways are there to climb $n$ steps?

Let $C_n$ = # ways to climb $n$ steps

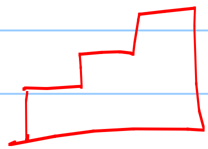
**Base cases:**



$C_1 = 1$



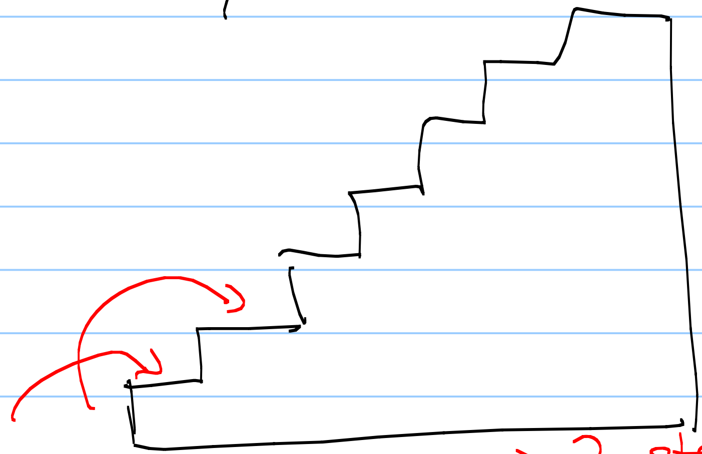
$C_2 = 2$



$C_3 = 3$

Think recursively:

What are my choices? (think about first step — what options?)



← go 2 steps ← go 1 step

$$C_n = C_{n-2} + C_{n-1}$$

Another example: bit strings of length $n$

How many bit strings are there with no 2 consecutive 0's?

Ex: 1101111

11010101011

~~10011111~~

Small cases:

1 bit: 1, 0 $\boxed{2}$

2 bits: 11, 10, 01, ~~00~~ $\boxed{3}$

3 bits: ~~000~~, ~~001~~, 011, 010, ~~100~~, 101, 110, 111 $\boxed{5}$

Let $b_n$ = # of bit strings of length $n$ with no 2 consecutive zeroes.

Consider the last bit: $\_\ \_\ \_\ \_\ \_\ \_\ \underline{1}$

What could it be?

<u>Case 1:</u> $1$ : $b_{n-1}$ $\quad \underbrace{\_\ \_\ \_\ \_\ \_\ \_\ \_}\ \underline{1}$

<u>Case 2:</u> $0$ : $b_{n-2}$ $\quad \underbrace{\_\ \_\ \_\ \_\ \_\ \_\ \_}\ \underline{1}\ \underline{0}$

$$b_n = b_{n-1} + b_{n-2}$$

# Recursively defined sets

Consider an ~~inductive~~ recursive definition for a set:

Base step: $3 \in S$

Recursive step: If $x \in S$ and $y \in S$, then $x + y \in S$.

So what are elements of S?

$\{3, 6, 9, 12, 15, \ldots\}$

Claim: $S$ = set of all positive integers divisible by 3. $= \{3n : n \in \mathbb{N}\} = A$

Pf: Show 2 sets are equal!
2 proofs:

$S \subseteq A$: Take $x \in S$. Show $x$ is divisible by 3.

induction on values in $S$:

Base step: if $x = 3$, then $x$ is div. by 3.

IH: Any value $< x$ is $S$ is div by 3.

(strong induction)

IS: Take $x$. Know $x = a + b$, where $a$ & $b$ are in $S$. By IH, $a$ & $b$ are div. by 3, so sum is div by 3.

$A \subseteq S$: take $x \in A$. Know $x = 3k$ for some $k \in \mathbb{N}$.

induction on $k$:

Base case $k = 1$, so $3k = 3 \in A$.
$3k = 3$ is also in $S$ (by base of dfn).

IH: Assume $3(k-1) \in S$

IS: Consider $3 \cdot k = \underbrace{3(k-1)}_{\in S \text{ by IH}} + \underbrace{3}_{\text{in } S \text{ by base}}$

→ rec dfn says
$3(k-1) + 3$ is also in $S$.

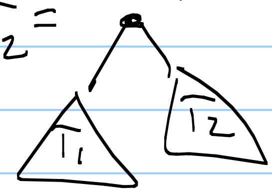
$\blacksquare$

Full binary trees: (each node has 0 or 2 children)

Defined recursively:

Base step: a single vertex is a full binary tree: •

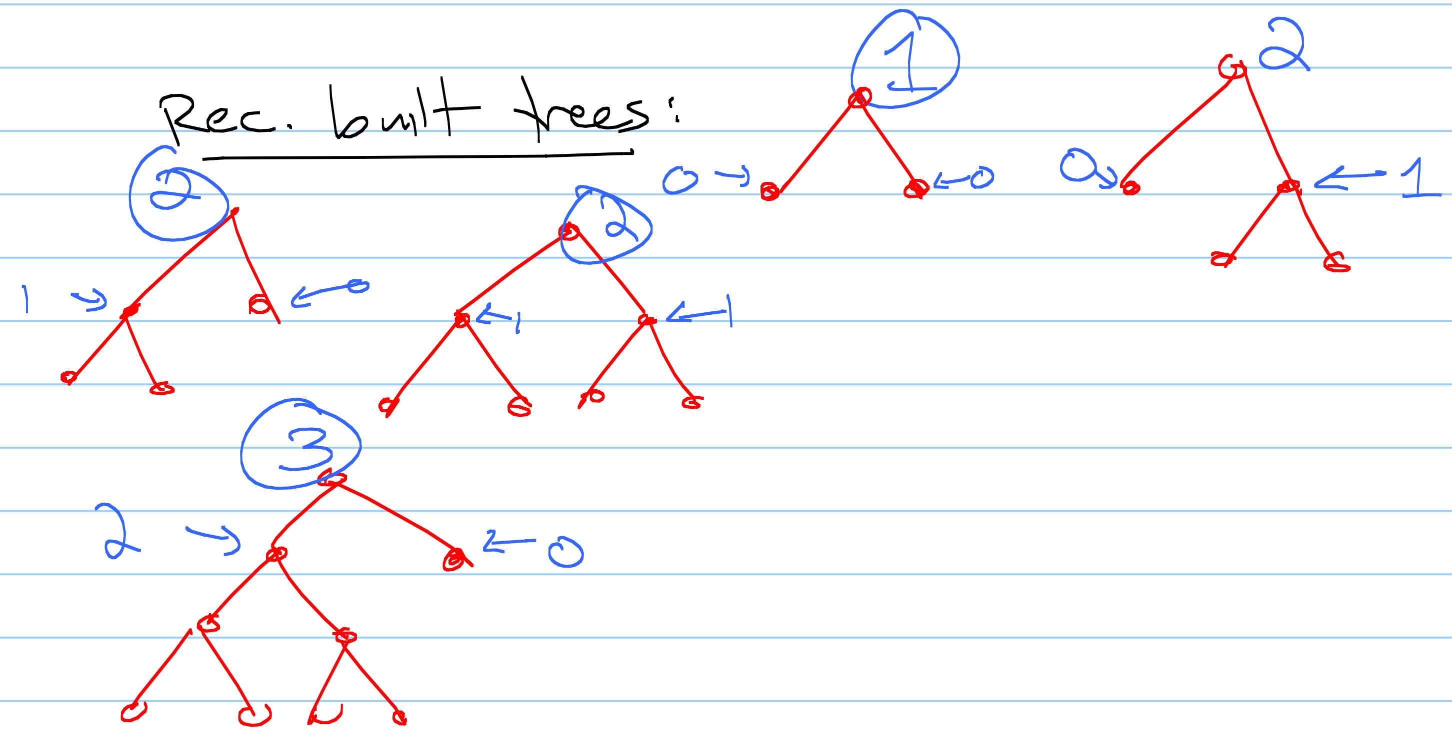
Recursive step: If $T_1$ & $T_2$ are full binary trees, then there is a full binary tree consisting of a new root with $T_1$ & $T_2$ as children: $T_1 \cdot T_2 =$



Height

Ex: Base • 0

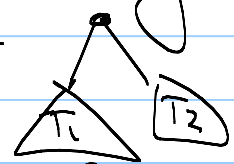
Rec. built trees:



Height of a tree is also defined recursively:

$h(T) = 0$ if $T$ has size 1 :

If $T_1$ & $T_2$ are full binary trees, then the tree $T_1 \cdot T_2$ =




has height $1 + \max\{h(T_1), h(T_2)\}$

$\Rightarrow h = O(\log n)$

Thm: If $T$ is a full binary tree, the number of vertices of $T$, written $n(T)$, is $\leq 2^{h(T)+1} - 1$

pf: induction on $h(T)$.

base case: single vertex

$n(T) = 1 \qquad h(T) = 0$

$n \leq 2^{h(T)+1} - 1$

$1 \quad \checkmark \quad 2^1 - 1 = 1$

IH : If $T'$ has height $< h(T)$,
inequality holds:
$$n(T') \leq 2^{h(T')+1} - 1$$

IS : Consider some tree $T$ with height $h(T)$.
Use rec dfn: know $T = T_1 \cdot T_2$
for some full binary trees $T_1$ + $T_2$
Know $h(T_1) < h(T)$
and $h(T_2) < h(T)$.