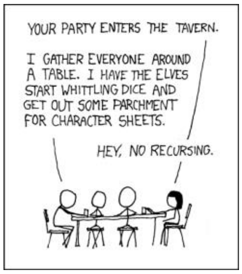
cse 311: foundations of computing

Spring 2015
Lecture 17: Recursively defined sets



administrative

Midterm review session tonight @ 6pm (EEB 105)

MIDTERM FRIDAY (IN THIS ROOM, USUAL TIME)

Closed book.
One page (front and back) of hand-written notes allowed.
Exam includes induction and strong induction!
Homework #5 is up now, but due on Friday, May 15th.

review: strong induction

$P(0)$
$\forall k\ \big((P(0) \wedge P(1) \wedge P(2) \wedge \cdots \wedge P(k)) \rightarrow P(k+1)\big)$

$\therefore \forall n\ P(n)$

Follows from ordinary induction applied to $Q(n) = P(0) \wedge P(1) \wedge P(2) \wedge \cdots \wedge P(n)$

review: strong induction English proof

1. **By induction we will show that $P(n)$ is true for every $n \geq 0$**
2. **Base Case: Prove $P(0)$**
3. **Inductive Hypothesis:**
   Assume that for some arbitrary integer $k \geq 0$, $P(j)$ is true for every j from 0 to k
4. **Inductive Step:**
   Prove that $P(k+1)$ is true using the Inductive Hypothesis (that $P(j)$ is true for all values $\leq k$)
5. **Conclusion: Result follows by induction**

review: every integer at least 2 is the product of primes

We argue by strong induction.
P(n) = "n can be expressed as a product of primes" for n ≥ 2.

Base Case:
Note that 2 is prime; so, we can express it as "2" which is a product of primes.

Induction Hypothesis:
Suppose P(2) ∧ P(3) ∧ · · · ∧ P(k) is true for some k ≥ 2.

Induction Step:
We go by cases.
Suppose k+1 is prime. Then, "k+1" is a product of primes.
Suppose k+1 is composite. Then, k+1 = ab for some a and b such that 1 < a, b < k+1.
By our IH, we know $a = p_1p_2 \cdots p_m$ and $b = q_1q_2 \cdots q_n$.
So, k+1 = ab = "$p_1p_2 \cdots p_mq_1q_2 \cdots q_n$", which is a product of primes.

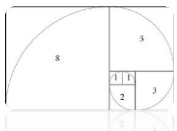
Thus, our claim is true for n ≥ 2 by strong induction.

review: recursive definition of functions

- $F(0) = 0;\ F(n+1) = F(n) + 1$ **for all** $n \geq 0$
- $G(0) = 1;\ G(n+1) = 2 \times G(n)$ **for all** $n \geq 0$
- $0! = 1;\ (n+1)! = (n+1) \times n!$ **for all** $n \geq 0$
- $H(0) = 1;\ H(n+1) = 2^{H(n)}$ **for all** $n \geq 0$

review: Fibonacci numbers

$f_0 = 0$
$f_1 = 1$
$f_n = f_{n-1} + f_{n-2}$ for all $n \geq 2$



review: bounding the Fibonacci numbers

Theorem: $f_n < 2^n$ for all $n \geq 2$.

bounding the Fibonacci numbers

Theorem: $2^{\frac{n}{2}-1} \leq f_n < 2^n$ for all $n \geq 2$

running time of Euclid's algorithm

running time of Euclid's algorithm

Theorem: Suppose that Euclid's algorithm takes n steps for $\gcd(a, b)$ with $a > b$, then $a \geq f_{n+1}$.

Proof:

Set $r_{n+1} = a, r_n = b$ then Euclid's algorithm computes

$r_{n+1} = q_n r_n + r_{n-1}$
$r_n = q_{n-1} r_{n-1} + r_{n-2}$
$\vdots$
$r_3 = q_2 r_2 + r_1$
$r_2 = q_1 r_1$

each quotient $q_i \geq 1$
$r_1 \geq 1$

recursive definition of sets

Recursive definition

- **Basis step:** $0 \in S$
- **Recursive step:** if $x \in S$, then $x + 2 \in S$
- **Exclusion rule:** Every element in S follows from basis steps and a finite number of recursive steps

recursive definition of sets

Basis: $6 \in S$; $15 \in S$;
Recursive: if $x, y \in S$, then $x + y \in S$;

Basis: $[1, 1, 0] \in S, [0, 1, 1] \in S$;
Recursive:
if $[x, y, z] \in S,\ \alpha \in \mathbb{R}$, then $[\alpha x, \alpha y, \alpha z] \in S$
if $[x_1, y_1, z_1], [x_2, y_2, z_2] \in S$
then $[x_1 + x_2,\ y_1 + y_2,\ z_1 + z_2] \in S$

Powers of 3:

recursive definitions of sets: general form

Recursive definition
- *Basis step:* Some specific elements are in S
- *Recursive step:* Given some existing named elements in S some new objects constructed from these named elements are also in S.
- *Exclusion rule*: Every element in S follows from basis steps and a finite number of recursive steps

strings

- An *alphabet* Σ is any finite set of characters.
- The set Σ^* of *strings* over the alphabet Σ is defined by
  - **Basis:** $\varepsilon \in \Sigma^*$ (ε is the empty string)
  - **Recursive:** if $w \in \Sigma^*, a \in \Sigma$, then $wa \in \Sigma^*$

palindromes

Palindromes are strings that are the same backwards and forwards.

Basis:
ε is a palindrome and any $a \in \Sigma$ is a palindrome

Recursive step:
If p is a palindrome then apa is a palindrome for every $a \in \Sigma$.

all binary strings with no 1's before 0's

function definitions on recursively defined sets

Length:
len (ε) = 0;
len (wa) = 1 + len(w); for $w \in \Sigma^*, a \in \Sigma$

Reversal:
$\varepsilon^{R} = \varepsilon$
$(wa)^{R} = aw^{R}$ for $w \in \Sigma^*, a \in \Sigma$

Concatenation:

function definitions on recursively defined sets

Length:

len (ε) = 0;

len (wa) = 1 + len(w); for $w \in \Sigma^*, a \in \Sigma$

Reversal:

$\varepsilon^R = \varepsilon$

$(wa)^R = aw^R$ for $w \in \Sigma^*, a \in \Sigma$

Concatenation:

$x \bullet \varepsilon = x$ for $x \in \Sigma^*$

$x \bullet wa = (x \bullet w)a$ for $x, w \in \Sigma^*, a \in \Sigma$