Discrete Mathematical Structures
CS 3233 Lecture 27

Prof. William Winsborough

November 7, 2005

Business

- Assignment 9, due to Humayun in class on Wednesday 11/9
- Assignment 10, due my office Thursday 11/17:
  - Section 3.4: 4, 6, 8, 12, 48, 56
- I am out until Monday 11/14
  - Review conducted by Humayun that day
  - Midterm II Friday 11/11
- Today’s lecture is covered by Midterm II
- Mock exam and solutions are on course website

Recursive Definition

- A *recursive definition* (also called an *inductive definition*) of a function f over N is given by defining the function
  - The base case: f(0) is defined; more generally f(i) may be defined for all i less or equal to some $k \in N$
  - The recursive case: f(n+1) is defined in terms of f(n), f(n-1), …, f(0)
- Observe that f is a sequence

Examples

- Factorial F(n) = n!
  - F(0) = 1
  - F(n+1) = (n+1) F(n)
  - Defines the sequence {0!, 1!, 2!, …}
- Exponentiation
  - $a^0 = 1$
  - $a^{n+1} = a \cdot a^n$
- Σ: Sum of first n elements of a sequence $\{a_k\}$
  - $\sum_{0 \le i \le 0} a_i = 0$
  - $\sum_{0 \le i \le n+1} a_i = \sum_{0 \le i \le n} a_i + a_{n+1}$

Fibonacci Numbers

- The *Fibonacci numbers*, f_0, f_1, f_2, …, are defined by:
  - $f_0 = 0$
  - $f_1 = 1$
  - $f_n = f_{n-1} + f_{n-2}$ for $n > 1$
- {1, 2, 3, 5, 8, …}

Recursively Defined Sets

- Can define a set by giving some elements (basis) and giving a rule for constructing other elements from ones in the set (step)
  - Exclusion rule: defines the *smallest* set satisfying the basis and the step
- Example: Define S as follows
  - Basis: $3 \in S$
  - Step: if $x \in S$ and $y \in S$, then $x+y \in S$

Set of Strings

- Given an alphabet Σ the set of *strings* Σ^*
  - Basis: $\lambda \in \Sigma^*$ (λ denotes the empty string)
  - Step: if $w \in \Sigma^*$ and $x \in \Sigma$, then $wx \in \Sigma^*$ (concatenation)
- Examples:
  - Bit strings
  - words