Discrete Mathematics

Recursion

Shin Hong

Recursively Defined Functions

Example: Suppose f is defined by:

$$f(0) = 3,$$

$$f(n + 1) = 2f(n) + 3$$

Find $f(1), f(2), f(3), f(4)$

Solution:

- $f(1) = 2f(0) + 3 = 2 \cdot 3 + 3 = 9$
- $f(2) = 2f(1) + 3 = 2 \cdot 9 + 3 = 21$
- $f(3) = 2f(2) + 3 = 2 \cdot 21 + 3 = 45$
- $f(4) = 2f(3) + 3 = 2 \cdot 45 + 3 = 93$

Example: Give a recursive definition of the factorial function $n!$:

Solution:

$f(0) = 1$
$f(n + 1) = (n + 1) \cdot f(n)$

Recursively Defined Functions

Example: Give a recursive definition of:

$$\sum_{k=0}^{n} a_k.$$

Solution:

The first part of the definition is $\displaystyle\sum_{k=0}^{0} a_k = a_0.$

The second part is $\displaystyle\sum_{k=0}^{n+1} a_k = \left(\sum_{k=0}^{n} a_k\right) + a_{n+1}.$

Fibonacci Numbers

Fibonacci
(1170- 1250)

Example : The Fibonacci numbers are defined as follows:

$f_0 = 0$
$f_1 = 1$
$f_n = f_{n-1} + f_{n-2}$

Find f_2, f_3, f_4, f_5 .

- $f_2 = f_1 + f_0 = 1 + 0 = 1$
- $f_3 = f_2 + f_1 = 1 + 1 = 2$
- $f_4 = f_3 + f_2 = 2 + 1 = 3$
- $f_5 = f_4 + f_3 = 3 + 2 = 5$

Fibonacci Numbers

- **Example** Show that whenever $n \geq 3$, $f_n > \alpha^{n-2}$, where $\alpha = (1 + \sqrt{5})/2$.
- **Solution**: Let $P(n)$ be the statement $f_n > \alpha^{n-2}$.
  Use strong induction to show that $P(n)$ is true whenever $n \geq 3$.
  - BASIS STEP: $P(3)$ holds since $\alpha < 2 = f_3$
    $P(4)$ holds since $\alpha^2 = (3 + \sqrt{5})/2 < 3 = f_4$
  - INDUCTIVE STEP: Assume that $P(j)$ holds, i.e., $f_j > \alpha^{j-2}$ for all integers j with $3 \leq j \leq k$, where $k \geq 4$. Show that $P(k + 1)$ holds, i.e., $f_{k+1} > \alpha^{k-1}$.
    - Since $\alpha^2 = \alpha + 1$ (because α is a solution of $x^2 - x - 1 = 0$),
      $\alpha^{k-1} = \alpha^2 \cdot \alpha^{k-3} = (\alpha + 1) \cdot \alpha^{k-3} = \alpha \cdot \alpha^{k-3} + 1 \cdot \alpha^{k-3} = \alpha^{k-2} + \alpha^{k-3}$
    - By the inductive hypothesis, because $k \geq 4$ we have
      $f_{k-1} > \alpha^{k-3}$, $\quad f_k > \alpha^{k-2}$.
    - Therefore, it follows that
      $f_{k+1} = f_k + f_{k-1} > \alpha^{k-2} + \alpha^{k-3} = \alpha^{k-1}$.
    - Hence, $P(k + 1)$ is true.

Recursively Defined Sets and Structures

Recursive definitions of sets have two parts:

- The *basis step* specifies an initial collection of elements.
- The *recursive step* gives the rules for forming new elements in the set from those already known to be in the set.

- Sometimes the recursive definition has an *exclusion rule*, which specifies that the set contains nothing other than those elements specified in the basis step and generated by applications of the rules in the recursive step.
- We will always assume that the exclusion rule holds, even if it is not explicitly mentioned.
- We will later develop a form of induction, called *structural induction*, to prove results about recursively defined sets.

Recursively Defined Sets and Structures

Example: The natural numbers **N**.

BASIS STEP: $0 \in$ **N.**

RECURSIVE STEP: If *n* is in **N**, then $n + 1$ is in **N**.

- Initially 0 is in *S*, then $0 + 1 = 1$, then $1 + 1 = 2$, etc.

Example : Subset of Integers *S*:

BASIS STEP: $3 \in$ S.

RECURSIVE STEP: If $x \in S$ and $y \in S$, then $x + y$ is in *S*.

- Initially 3 is in *S*, then $3 + 3 = 6$, then $3 + 6 = 9$, etc.

Strings

- **Definition**: The set Σ^* of *strings* over the alphabet Σ:
  - BASIS STEP: $\lambda \in \Sigma^*$ (λ is the empty string)
  - RECURSIVE STEP: If w is in Σ^* and x is in Σ, then $wx \in \Sigma^*$

- **Example**: If $\Sigma = \{0,1\}$, the strings in in Σ^* are the set of all bit strings, λ,$0,1, 00,01,10, 11,$ etc.

- **Example**: If $\Sigma = \{a,b\}$, show that aab is in Σ^*.
  - Since $\lambda \in \Sigma^*$ and $a \in \Sigma$, $a \in \Sigma^*$
  - Since $a \in \Sigma^*$ and $a \in \Sigma$, $aa \in \Sigma^*$
  - Since $aa \in \Sigma^*$ and $b \in \Sigma$, $aab \in \Sigma^*$

Length of a String

- **Example**: Give a recursive definition of $l(w)$, the length of the string w.
- **Solution**: The length of a string can be recursively defined by:

  $l(\lambda) = 0$;

  $l(wx) = l(w) + 1$ if $w \in \Sigma^*$ and $x \in \Sigma$.

Balanced Parentheses

Example: Give a recursive definition of the set of balanced parentheses *P*.

Solution:

BASIS STEP: () ∈ *P*

RECURSIVE STEP:

- (w) ∈ *P* for w ∈ *P*
- w w' ∈ *P* for w ∈ *P* and w' ∈ *P*

- Show that (() ()) is in *P*.
- Why is))(() not in *P*?

String Concatenation

- **Definition**: Two strings can be combined via the operation of *concatenation*. Let Σ be a set of symbols and Σ* be the set of strings formed from the symbols in Σ. We can define the concatenation of two strings, denoted by ·, recursively as follows.
  - BASIS STEP: If $w \in \Sigma^*$, then $w \cdot \lambda = w$.
  - RECURSIVE STEP: If $w_1 \in \Sigma^*$ and $w_2 \in \Sigma^*$ and $x \in \Sigma$, then $w \cdot (w_2 x) = (w_1 \cdot w_2)x$.
- Often $w_1 \cdot w_2$ is written as $w_1 w_2$.
- If w_1 = *abra* and w_2 = *cadabra*, the concatenation $w_1 w_2$ = *abracadabra.*

Well-Formed Formulae in Propositional Logic

- **Definition**: The set of *well-formed formulae* in propositional logic involving **T**, **F**, propositional variables, and operators from the set $\{\neg,\wedge,\vee,\rightarrow,\leftrightarrow\}$
  - BASIS STEP: **T**,**F**, and *s*, where *s* is a propositional variable, are well-formed formulae.
  - RECURSIVE STEP: If *E* and *F* are well formed formulae, $(\neg E)$, $(E \wedge F)$, $(E \vee F)$, $(E \rightarrow F)$, $(E \leftrightarrow F)$, are well-formed formulae.

Examples: $((p \vee q) \rightarrow (q \wedge \mathbf{F}))$ is a well-formed formula.

$pq \wedge$ is not a well formed formula.