- In the last lecture:

  1. we defined sets (other than the natural numbers) using induction. The set of SL (Simple-Logical) Expressions was one such set.

  2. we introduced structural induction

  3. we got an idea of how to prove properties of such sets

- In this lecture:

  1. we will define some additional sets inductively

  2. we will prove properties of these sets using structural induction

- Consider the following set defined inductively
  - Basis Step: $3 \in S$
  - Recursive Step: If $x \in S$ and $y \in S$ then $x + y \in S$

- List some elements of the set S

- Consider an alphabet denoted by $\Sigma = \{0, 1\}$

- The set Σ^* is the set of all strings over this alphabet, i.e. all combinations of 0 and 1.

- Σ^* can be defined recursively as:
  - Basis Step: $\gamma \in \Sigma^*$
  - Recursive step: If $\omega \in \Sigma^*$ and $x \in \Sigma$ then $\omega x \in \Sigma^*$

- List some elements of this set

- Define a function over the set Σ^*. Define $l(\omega)$, the length of the string ω
  - $l(\gamma) = 0$
  - $l(\omega x) = l(\omega) + 1$, where $\omega \in \Sigma^*$ and $x \in \Sigma$

- Graph: A Graph is a structure which consists of a finite nonempty set of nodes, V, together with a prescribed set of edges, E of unordered pairs of distinct nodes of V.

- Tree - A connected graph without cycles. (connected means in one piece and a cycle is an alternating series of nodes and edges starting and finishing at the same point.

- A graph (and hence a tree) can have a direction associated with the edges of the graph

- Rooted Tree - top node has no incoming edges

- If there is an edge from node i to node j then i is the parent of j and j is the child of i

- Nodes without children are called leaves, all other nodes are called branch nodes

- In a binary tree we distinguish between a left child and right child

- Recursive definition of a binary tree

  1. The empty tree is a binary tree

  2. If A and B are binary trees, and if c is a node then (A, c, B) is a binary tree. Here A is the left subtree and B is the right subtree

  3. Nothing else is a binary tree.

- Identify similarities between this definition and the inductive definition of the natural numbers

- Note: Rosen in the book Discrete Mathematics and its Applications(5th Edition), Section 3.4, pg 265 calls these binary trees 'extended binary trees'.

- Rosen uses a slightly different notation

- Describe a simple binary tree using the representation described in the recursive definition.

- All rules in our recursive definitions are lenghtening rules, i.e. the string or the tree that's generated gets longer or bigger.

- Structural Induction is applicable in domains containing only lengthening rules.

- Each node in a tree has a level. Suppose the root node is at level 1. The children of a node on level n have a level of $n + 1$.

- The height of a tree is the maximum level found in the tree. The empty tree has height 0.

- Prove properties of binary trees using structural induction.

- Prove: The maximum number of nodes in a binary tree of height n is $2^n - 1$ and there is a binary tree of height n with $2^n - 1$ nodes.

- Base case: A tree of height 0 is the empty tree which has $2^0 - 1$ nodes, i.e. 0 nodes

- Assume the inductive hypothesis: That is assume that trees of height $n = k$ have at most $2^k - 1$ nodes.

- Inductive step: Show that the property holds for a tree of height $n = k + 1$. Such a tree can have at most $2^{(k+1)} - 1$ nodes.

- Create a tree of height $k + 1$ from 2 subtrees each of height k.

- Introduce a new root node with the subtrees of height n as the left and right children

- We now have a tree of height $k+1$.

- How many nodes does this tree have?

- $2(2^k - 1) + 1 = 2^{(k+1)} - 2 + 1 = 2^{(k+1)} - 1$

- This is the maximum number of nodes a tree of height $(k+1)$ can have.

- Conclusion: All trees of height n have at most $2^k - 1$ nodes.

- Prove: A non-empty binary tree of n nodes contains exactly $n + 1$ empty subtrees

- Inductive Base: How many empty subtrees has a tree with 1 node (n=1).

- Assume the inductive hypothesis: Trees with $n = k$ nodes contain $k+1$ empth subtrees

- Again combine 2 trees, each with k nodes. The new tree has $2k+1$ nodes and $2(k+1)$ empty subtrees $= 2k + 2$ empty subtrees.