

CIS-280
Lab 6: Operations on Trees
Monday, April 4, 2005

File `~carberrry/Lab6-treecode.scm` contains the procedures `entry`, `left-subtree`, and `right-subtree` discussed in class. You will want to copy this file into your own directory. This file also contains a procedure `form-tree` that takes as argument a set represented as an unordered list and returns the set represented as an ordered binary tree.

1. Create objects `x` and `y` that are ordered binary trees by executing the commands

```
(define x (form-tree '(10 26 -3 4 12 8)))  
(define y (form-tree '(15 0 -12 7 10 26 -3 4 12 8)))
```

2. At the prompt, type the character `x` to display the internal form of the first tree structure. Make sure that you understand the representation. Execute the following commands to see the contents of the root node and the left and right subtrees. Make sure that you understand what is displayed.

```
(entry x)  
(left-subtree x)  
(right-subtree x)
```

3. Write a procedure `Smallest` that takes an ordered binary tree as argument and returns its smallest element. (Remember that the smallest element of an ordered binary tree will be the leftmost element in the tree structure.) Test your procedure on trees `x` and `y`.
4. Write a procedure `Count` that takes as argument an ordered binary tree and returns a count of the number of elements in the tree. Test your procedure on trees `x` and `y`.
5. Submit your procedures `Smallest` and `Count` on the lab submission system for Lab-6.