Prepared by Linan Qiu <lq2137@columbia.edu>

Trees vs Search Trees

Trees on their own are kind of clunky. Sure they can be useful for modeling family trees (and generations of amoeba as we have shown earlier). They can even be used for expressing mathematical expressions (and you surely have seen in homework 3). However, if you're trying to find a specific element in a tree, you will have to traverse the entire tree. That operation is $O(N)$, and this means that the tree is no better than a linked list in searching for an element.

We can create a kind of tree that imposes special comparable meaning on the `left` and `right` elements. For example, we can say that the `left` child of a node is always smaller than the node itself, and the `right` child is always greater than the node itself. Then, we can order the elements in a tree. These kind of "ordered" trees are called **search trees**.

Search trees can be categorized based on the number of children each node has. For example, a search tree where each node has two children is called a **binary search tree**, which will be the topic of the rest of this chapter. More complex trees exist, but we will only cover them briefly.

An example of a binary search tree of integers is shown below.

```
          +--+
          |10|
     +------------+
     |            |
   +-++          ++-+
   | 5|          |16|
  +-----+       +-----+
  |     |       |     |
+-++  ++-+    +-++  ++-+
| 2|  | 7|    |14|  |19|
+--+  +--+    +--+  +--+
```

We stipulate that

- The left child of a node is less than $<$ the node itself
- The right child of a node is greater than $>$ the node itself

The tree above shows this relationship. 5 is less than 10 while 16 is greater than 10. 7 is greater than 5, and since it is the grandchild of 10, it is also less than 10.

This makes searching for a specific element rather easy. For example, let's say I want to search for the number 14. I start at 10. I know that $14 > 10$, so I look towards 10's right. I do not need to care about anything to the left of 10. I have eliminated half the number of nodes in the tree from my search! Now that I'm at 16, I know that $14 < 16$, so I look towards 16's left. Again, I have eliminated all nodes on 16's right from my search. I arrive at 14, which is the element I'm looking for.

We engineer such a tree in the next chapter.