CSE 373: Binary Search Trees

Chapter 4

Binary Trees

*Binary Tree:* a Tree in which every node has two children or fewer
Numerical Trivia for Binary Trees

• Given a binary tree of depth \( d \)…
  - max number of nodes = \( \text{min} = \) \( \) 
  - max number of leaves = \( \text{min} = \)

• Building a binary tree out of \( n \) nodes…
  - max depth of tree = \( \text{min} = \)

Binary Search Tree

Binary Search Tree: a Binary Tree in which every node…
  - is greater than all of its left descendents
  - is less than all of its right descendents

UW, Spring 1999  CSE 373 – Data Structures and Algorithms  Bind Chamberlain
Binary Search Tree Operations

- Search Operations:
  - Position Find(SearchTree T, TType val);
  - Position FindMin(SearchTree T);
  - Position FindMax(SearchTree T);
- Collection Operations:
  - void Insert(SearchTree T, TType val);
  - void Delete(SearchTree T, TType val);
  - TType Retrieve(Position);
- Traversals...

Implementation

Similar to our naive fixed-degree Tree:

typedef struct _SearchTreeNode {
    TType data;
    struct _SearchTreeNode *left;
    struct _SearchTreeNode *right;
} SearchTreeNode;

(As with generic trees, may use a parent pointer)
**Traversals**

pre-order:

post-order:

in-order:

```c
void InOrder(SearchTreeNode *T) {
    if (T == NULL) {
        return;
    } else {
        InOrder(T->Left);
        Process(T);
        InOrder(T->right);
    }
}
```

Searching

```c
Find(T, 11);
Find(T, 9);
FindMin(T);
FindMax(T);
```
**Insert()**

```
Insert(T, 3);
Insert(T, 19);
Insert(T, 0);
```

```
4
\[\begin{array}{cc}
4 & 5 \\
5 & 6
\end{array}\]
```

**Delete()**

```
Delete(T, 2);
Delete(T, 20);
Delete(T, 11);
```

```
6
\[\begin{array}{cc}
4 & 1 \\
1 & 5
\end{array}\]
```

_UW, Spring 1999_  _CSE 373 – Data Structures and Algorithms_  _Brend Chamberlain_
Asymptotic Analysis

\[
\begin{array}{cccc}
\text{Search Tree} & \text{List} & \text{Sorted List (Array)} \\
\end{array}
\]

\text{problem size}
\text{space}

\text{Find()}
\text{FindMin()}
\text{FindMax()}

\text{Insert()}
\text{Delete()}
\text{traversals}

---

Food For Thought

If I read a list of integers from a file and insert them into a Binary Search Tree one by one, what’s an example of a worst-case file?