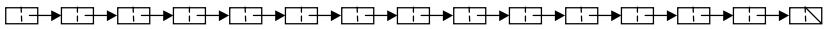
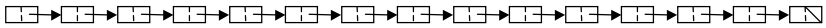


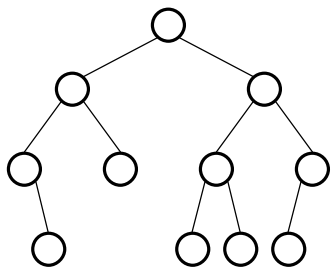
CSE 373: Binary Search Trees

Chapter 4



Binary Trees

Binary Tree: a Tree in which every node has two children or fewer



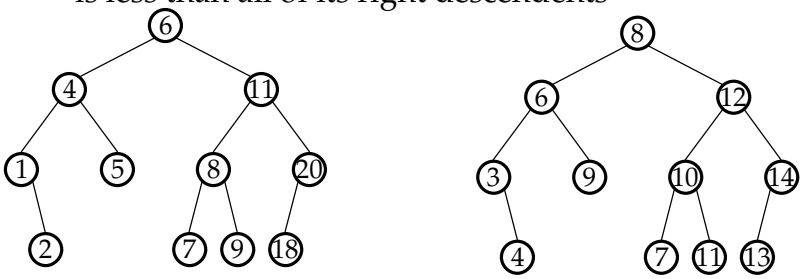
Numerical Trivia for Binary Trees

- Given a binary tree of depth *d*…
  - max number of nodes = min =
  - max number of leaves = min =
- Building a binary tree out of *n* nodes…
  - max depth of tree = 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



Binary Search Tree Operations

- Search Operations:

```
Position Find(SearchTree T, TType val);
Position FindMin(SearchTree T);
Position FindMax(SearchTree T);
```

- Collection Operations:

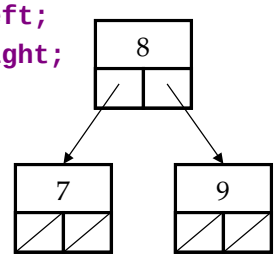
```
void Insert(SearchTree T, TType val);
void Delete(SearchTree T, TType val);
TType Retrieve(Position);
```

- 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:

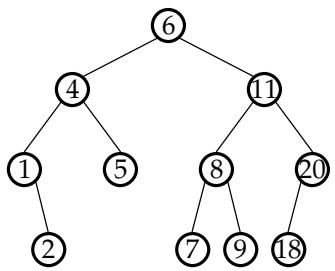
post-order:

in-order:

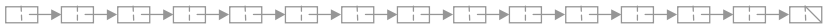
```
void InOrder(SearchTreeNode *T) {
  if (T == NULL) {
    return;
  } else {
    InOrder(T->Left);
    Process(T);
    InOrder(T->right);
  }
}
```


Searching

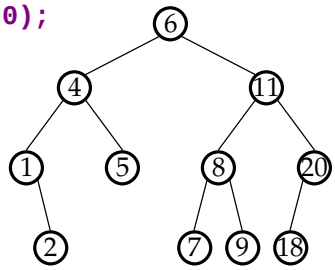
```
Find(T,11);
Find(T,9);
FindMin(T);
FindMax(T);
```



Insert()

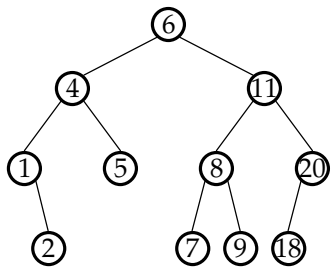


```
Insert(T,3);
Insert(T,19);
Insert(T,0);
```

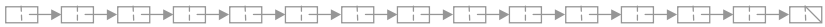


Delete()

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



Asymptotic Analysis



	SearchTree	List	Sorted List (Array)
problem size			
space			
`Find()`			
`FindMin()`			
`FindMax()`			
`Insert()`			
`Delete()`			
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?