

BST Iterators

Section ...

Ali Erkan
Ithaca College

Iterators in BSTs: A Few Questions...

- ▶ Why is the emphasis on iterators in the context of trees greater than the emphasis on iterators in the context of one dimensional data structures such as linked lists and arrays?

- 
- 

- ▶ What methods can we use to implement a binary tree iterator?

- 
- 
- 

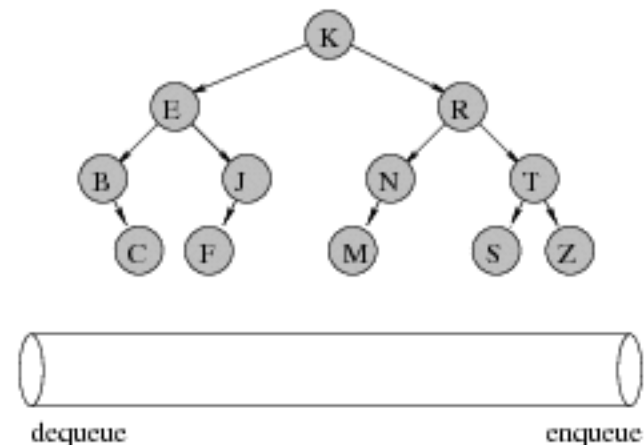
Iterators

- ▶ A mechanism that allows a **client** (what?) to process the contents of an arbitrary ADT one element at a time using a loop of the form

```
public static void process( Collection coll ) {  
    Iterator itr = coll.iterator();  
    while( itr.hasNext() ) {  
        Object theThing = itr.next();  
        ...Process theThing...  
    }  
}
```

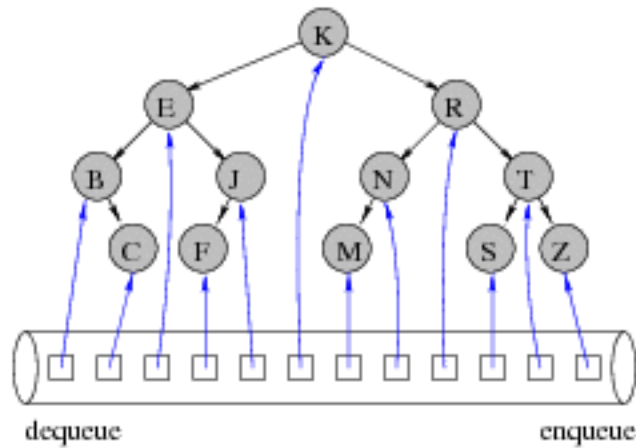
- ▶ Their utility comes from the fact that you can traverse an ADT without knowing (and caring to know) its internal arrangement

Queue Based Iterator Design: In Order

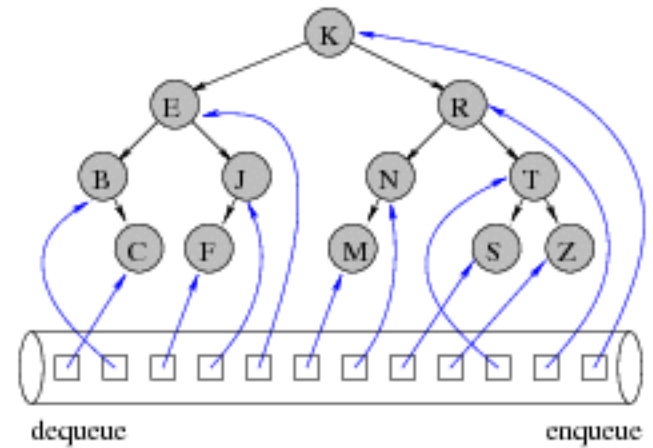


The idea is to start a **recursive in order traversal** method on the tree where the **processing** performed at each node is to **enqueue** a pointer to that node

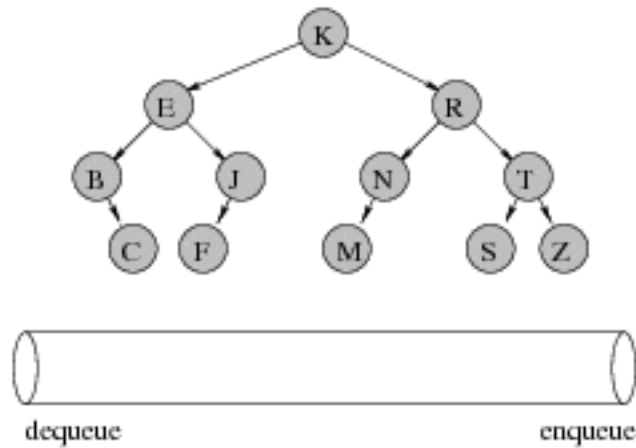
Queue Based Iterator Design: In Order



Queue Based Iterator Design: Post-Order

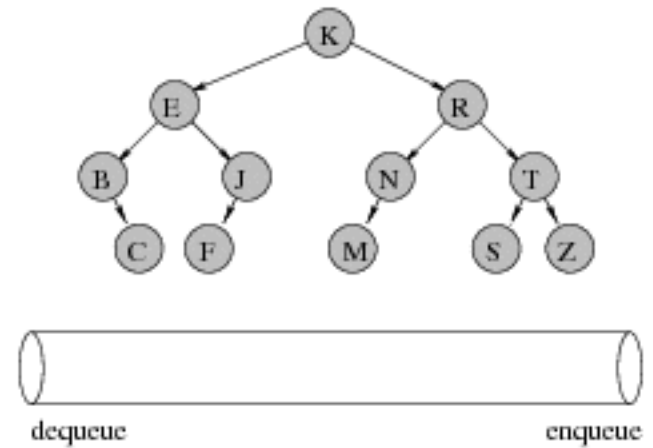


Queue Based Iterator Design: Post-Order



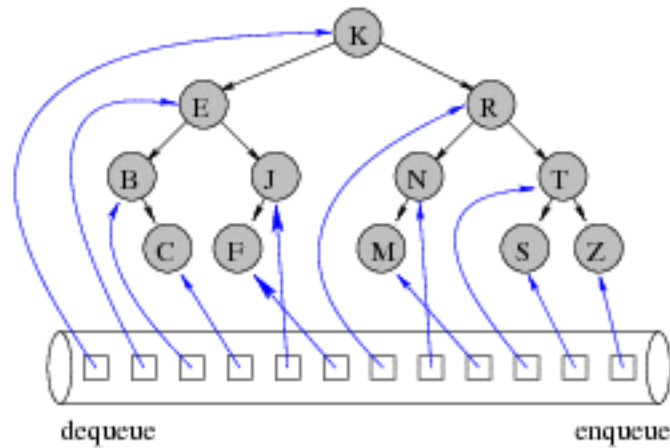
The idea is to start a **recursive post-order traversal** method on the tree where the **processing** performed at each node is to **enqueue** a pointer to that node

Queue Based Iterator Design: Pre-Order



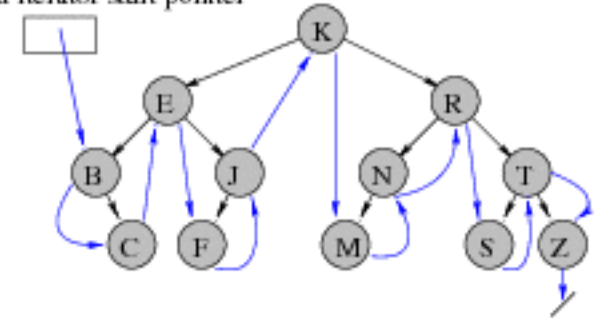
The idea is to start a **recursive pre-order traversal** method on the tree where the **processing** performed at each node is to **enqueue** a pointer to that node

Queue Based Iterator Design: Pre-Order



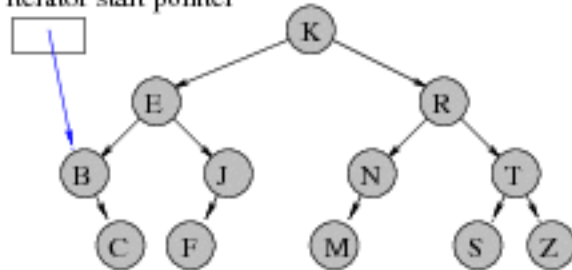
Thread Based Iterator

"Thread" based iterator start pointer



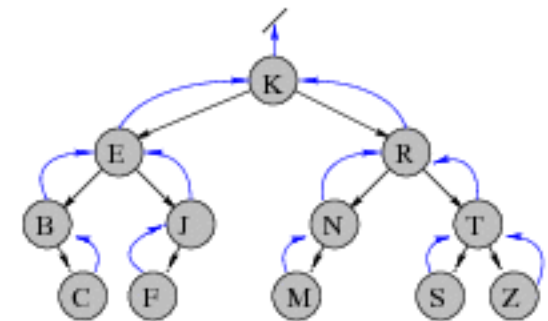
Thread Based Iterator

"Thread" based iterator start pointer



The idea is to start a **recursive in order traversal** method whose behavior is similar to the loops we set in linked lists where we had a 'current' pointer (pointing to the current node) and a 'last' pointer (pointing to the node 'current' most recently used to point to). At each point, the 'next' field of a binary node is set equal to the value of the 'current' pointer.

Parent Link Based Iterator



This scheme has no setup cost. All we do is to modify the node insertion method of the binary search tree class so that when a new node is added, the 'parent' pointer of the new node points to its parent.