

```
Stack<String> s = new Stack<String>();  
Queue<String> q = new Queue<String>();  
String item;  
While(!s.empty()) {  
    Item = s.pop();  
    q.offer(item);  
}  
While(!q.isEmpty()) {  
    Item = q.remove();  
    s.push(item);  
}
```

- a. First loop: s loses the topmost string and is sent to the queue.  
After loop: will continue this timeline until the queue is empty.
- b. Stack s will contain what it had before but in the reverse order since its grabbing the topmost string and pushing it to the top of s. And q will be empty once again.