As you arrive, snarf the code for today. Then check your solution to the prep work against mine:

```java
public void reverseTopK(Stack<String> stack, int k) {
    Queue<String> queue = new LinkedList<String>();
    for(int i = 0; i < k; i++) {
        queue.add(stack.pop());
    }
    for(int i = 0; i < k; i++) {
        stack.push(queue.remove());
    }
}
```

Stacks and Queues

Why do we care about stacks and queues?

- Can’t do anything a linked list can’t do
- Represent the idea of deferring work
- Often have the form:

```java
while(!s.isEmpty()) { //we still have work to do
    Item current = s.pop() //or q.remove()
    doSomethingToItem(current);
    for(Item subproblem : current.subproblems()) {
        s.push(subproblem) //or q.add()
    }
}
```

Dumping Eclipse Directories

We want to display all the directories (not files – just directories) in the Eclipse workspace, followed by all their children. Note that children come *before* other directories at the same level.

BTW, if you know about recursion DON’T use it to solve this.

Example output:

```
C:\Users\Mike\Dropbox\workspace
C:\Users\Mike\Dropbox\workspace\MikeStatusExampleSolution
C:\Users\Mike\Dropbox\workspace\MikeStatusExampleSolution\src
C:\Users\Mike\Dropbox\workspace\MikeStatusExampleSolution\bin
C:\Users\Mike\Dropbox\workspace\MikeStatusExampleSolution\settings
C:\Users\Mike\Dropbox\workspace\MikeStatusExample
C:\Users\Mike\Dropbox\workspace\MikeStatusExample\src
C:\Users\Mike\Dropbox\workspace\MikeStatusExample\bin
C:\Users\Mike\Dropbox\workspace\MikeStatusExample\settings
```
EvilMazeSolver

• We have a maze, we’re trying to find the exit
• Start at a particular maze node
• You can always get the nodes that follow a node by saying getPaths(). These all paths to new nodes – there’s no call to return to your parent (turns out, you don’t need to do that)
• Our goal is to find the distance to the exit
• Beware! The maze is infinitely large!

Paren Matching

• Two kinds of parens “[“ “[”
• Given a particular string of parens, we want to know if all the parens match
  – “[“ or “[“ matches, but “[“ or “[“ doesn’t
  – Can be nested, or in sequence: “{[([[]])]}”
  – Every paren must be matched so “[“ or “[“ don’t match
• Stack or queue?