Program 2

As a continuation of Project 1, your program will still solve Jeff Horn's robots exactly as specified as before. However, now you will clean up the program a bit; it should no longer be a COMPLETE hack job.

You will store positions that need to be checked in a queue. The queue should be implemented via a linked list. DO NOT use any built-in Java queues or linked lists. Build your own instead.

Use the following classes in your code. You do NOT have to use these names exactly, but it must have this structure. Put each class in its own file.

```java
class Position {
    should have a row and column variable to indicate exactly where in the maze you are.
}
class LinkedListNode {
    should have a Position variable to store position and a next pointer to get to the next element in the linked list.
}
class LinkedList {
    should control the entire linked list structure. Should control a head and tail pointer to the appropriate linked list nodes. Should include methods return the head and tail pointers, to add and remove nodes to the head and to add (but not remove) nodes to the tail. This class should manipulate LinkedListNodes only; it should NOT deal directly with Positions.
}
class Queue {
    should control a queue by means of a linked list. You should write an enqueue method that adds a Position to the queue and a dequeue method that removes and returns a Position from the queue. You do not have to worry about what happens when you try to remove an element from an empty queue. In this project, that will never happen.
}
```

Create a directory called “PG2” directly off your shared folder. Place all files pertaining to this program in this directory. Your main method should be in “PG2.java”.


Your classes should have appropriate constructors and all class variables should be private (so that they cannot be modified outside the class). The methods should be public or private as appropriate. Use this queue structure to control the queue in Jeff Horn's robots. The rest of the program is still allowed to be a hack job, however.