

```

1 /*
2  * Written by JJ Shepherd
3  */
4 public class LLQueue <T> implements QueueI<T>
5 {
6     private class ListNode
7     {
8         T data;
9         ListNode link;
10        public ListNode(T aData, ListNode aLink)
11        {
12            data = aData;
13            link = aLink;
14        }
15    }
16    private ListNode head;
17    private ListNode tail;
18    private int size;
19    public LLQueue()
20    {
21        head = tail = null;
22    }
23    public void enqueue(T aData)
24    {
25        ListNode newNode = new ListNode(aData,null);
26        if(head == null)
27        {
28            head = tail = newNode;
29            size = 1;
30            return;
31        }
32        tail.link = newNode;
33        tail = tail.link;
34        size++;
35    }
36    public T dequeue()
37    {
38        if(head == null)
39            return null;
40        T ret = head.data;
41        head = head.link;
42        size--;
43        return ret;
44    }
45    public T peek()
46    {
47        if(head == null)
48            return null;
49        return head.data;
50    }
51    public void print()
52    {
53        for(ListNode temp = head;temp!=null;temp = temp.link)
54            System.out.println(temp.data);
55    }
56    public int size()
57    {

```

LLQueue.java

```
58     return this.size;
59 }
60 }
61
```