Homework Turnin

Account:	6G_06 (rgalanos@fcps.edu)
Section:	6G
Course:	TJHSST APCS 2016-17
Assignment:	05-06
Receipt ID:	32417558ff5bec79bb69695f09058285

Execution failed with return code 1 (general error). (Expected for JUnit when any test fails.)

Warning: Your program failed to compile :

```
CollectionsSpeed_shell.java:4: error: class CollectionsSpeed is public, should be declared in a file named CollectionsSpeed.java
    public class CollectionsSpeed
           ^
1 error
```

Please correct your file(s), go back, and try to submit again. If you do not correct this problem, you are likely to lose a large number of points on the assignment. Please contact your TA if you are not sure why your code is not compiling successfully.

Turnin Failed! (See above)

There were some problems with your turnin. Please look at the messages above, fix the problems, then Go Back and try your turnin again.

GradeIt has a copy of your submission, but we believe that you will want to fix the problems with your submission by resubmitting a fixed version of your code by the due date.

We have received the following file(s):

CollectionsSpeed.java (2399 bytes)

```
1.  //  name:       date:
2.  import java.util.*;
3.
4.  public class CollectionsSpeed
5.  {
6.     public static int N = 10000;
7.
8.     public static void main(String[] args)
9.     {
10.       ArrayList<Integer> al = new ArrayList<Integer>();
11.       LinkedList<Integer> ll = new LinkedList<Integer>();
12.       makeValues(al, ll);
13.
14.       System.out.println("get each for ArrayList = " + timeGetEach(al));
15.       System.out.println("get each for LinkedList = " + timeGetEach(ll));
16.       System.out.println("\nadd at 0 for ArrayList = " + timeAddFirst(al));
17.       System.out.println("add at 0 for LinkedList = " + timeAddFirst(ll));
18.       System.out.println("\nadd at list.size() for ArrayList = " + timeAddLast(al));
19.       System.out.println("add at list.size() for LinkedList = " + timeAddLast(ll));
20.       System.out.println("addLast for LinkedList = " + timeAddLastLL(ll));
21.    }
22.    public static void makeValues(ArrayList<Integer> al, LinkedList<Integer> ll)
23.    {
24.       for(int i=0; i<N; i++)
25.       {
26.          al.add((int)(Math.random()*10));
27.          ll.addLast((int)(Math.random()*10));
28.       }
29.    }
30.       /* get N items by searching for each one. */
31.    public static double timeGetEach(List<Integer> list)
32.    {
33.       double start = System.nanoTime();
34.       for( int i = 0; i < N; i++ )
35.       {
36.          int index = list.get(i);
37.       }
38.       return (System.nanoTime() - start)/1E6;
39.    }
40.    /* add 10000 new objects at position 0  */
41.    public static double timeAddFirst(List<Integer> list)
42.    {
43.       double start = System.nanoTime();
44.       for(int i=0;i<10000;i++)
45.          list.add(0, (int)(Math.random()*10));
46.       return (System.nanoTime() - start)/1E6;
47.    }
48.    /* add 10000 new objects at position list.size() */
49.    public static double timeAddLast(List<Integer> list)
50.    {
51.       double start = System.nanoTime();
52.       for(int i=0;i<10000;i++)
53.          list.add(list.size(), (int)(Math.random()*10));
54.       return (System.nanoTime() - start)/1E6;
55.    }
56.    /* add 10000 new objects at the end.
57.       Uses the LinkedList method addLast().
58.       You must cast List list into a LinkedList.     */
59.    public static double timeAddLastLL(List<Integer> list)
60.    {
61.       double start = System.nanoTime();
62.       LinkedList<Integer> list2 = (LinkedList<Integer>)list;
63.       for(int i=0;i<10000;i++)
64.          list2.addLast((int)(Math.random()*10));
65.       list = list2;
66.       return (System.nanoTime() - start)/1E6;
67.    }
68. }
```

89.