CS 61B Spring 2023

Inheritance

Exam-Level 4: Februrary 6, 2023

1 Forget It, We Ball

The 61Ballers are organizing the best IM team at Cal, but they first need your help with some inheritance issues...

Suppose we have the `Person` interface and the `Athlete`, and `SoccerPlayer` classes defined below.

```
interface Person {
    void speakTo(Person other);
    default void watch(Athlete other) { System.out.println("wow"); }
}

public class Athlete implements Person {
    @Override
    public void speakTo(Person other) { System.out.println("i love sports"); }
    @Override
    public void watch(Athlete other) { System.out.println("ball is life"); }
}

public class SoccerPlayer extends Athlete {
    @Override
    void speakTo(Person other) { System.out.println("join 61ballers"); }
}
```

For each line below, write what, if anything, is printed after its execution. Write CE if there is a compiler error and RE if there is a runtime error. If a line errors, continue executing the rest of the lines.

You may find it helpful to fill out the table below.

```
Person ayati = new Person();

Athlete dhruv = new SoccerPlayer();

SoccerPlayer vidya = dhruv;

Person eric = new Athlete();

Athlete shreyas = new Athlete();

SoccerPlayer yaofu = new SoccerPlayer();

eric.watch(dhruv);
```

```
14
15 shreyas.speakTo(yaofu);
16
17 yaofu.speakTo(eric);
18
19 ((Athlete) yaofu).speakTo(eric);
20
21 ((Person) yaofu).speakTo(eric);
22
23 ((Athlete) eric).speakTo(shreyas);
24
25 ((SoccerPlayer) eric).watch(yaofu);
```

Line	**Compile Time (Static)**	**Runtime (Dynamic)**	**Output**
1			
3			
5			
7			
9			
11			
13			
15			
17			
19			
21			
23			
25			

2 List Inheritance

Modify the code below so that the max method of DMSList works properly. Assume all numbers inserted into DMSList are positive, and we only insert using `insertFront`. You may not change anything in the given code. You may only fill in blanks. You may not need all blanks. (Spring '16, MT1)

```
1  public class DMSList {
2      private IntNode sentinel;
3
4      public DMSList() {
5          sentinel = new IntNode(-1000, ______________________);
6      }
7
8      public class IntNode {
9          public int item;
10         public IntNode next;
11         public IntNode(int i, IntNode h) {
12             item = i;
13             next = h;
14         }
15
16         public int max() {
17             return Math.max(item, next.max());
18         }
19     }
20
21     class LastIntNode extends IntNode {
22         public LastIntNode() {
23
24             ____________________________________________________________;
25         }
26
27         @Override
28         public int max() {
29
30             ____________________________________________________________;
31         }
32     }
33
34     /* Returns 0 if list is empty. Otherwise, returns the max element. */
35     public int max() {
36         return sentinel.next.max();
37     }
38
39     public void insertFront(int x) { sentinel.next = new IntNode(x, sentinel.next); }
40 }
```