CS 61B Fall 2024

Inheritance

Exam-Level 03: September 16, 2024

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

2 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.

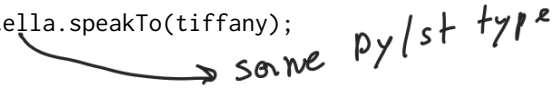
```
1  interface Person {
2      void speakTo(Person other);
3      void watch(Athlete other);
4  }
5
6  public class Athlete implements Person {
7      @Override
8      public void speakTo(Person other) { System.out.println("i love sports"); }
9      @Override
10     public void watch(Athlete other) { System.out.println("ball is life"); }
11 }
12
13 public class SoccerPlayer extends Athlete {
14     @Override
15     void speakTo(Person other) { System.out.println("join 61ballers"); }
16 }
```

Read the code below and fill in the table on the next page.

For lines 1-11, write down the static type of the object being created in the "Compile Time (Static)" column, the dynamic type in the "Runtime (Dynamic)" column. For the output, write nothing if there are no errors, write CE if there's a compiler error, and write RE if there's a runtime error.

For lines 13-25, identify the method that's been saved during compile time, and write down its name and the class it belongs to in the "Compile Time (Static)" column. Identify the method executed at runtime, and write down its information in the "Runtime (Dynamic)" column. Write output in the "Output" column, if anything. 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.

```
1  Person ryan = new Person();
2
3  Athlete daniel = new SoccerPlayer();
4
5  SoccerPlayer vanessa = daniel;  SP is subclass of athelete, compiler only knows static types
6
7  Person erik = new Athlete();
8
9  Athlete stella = new Athlete();
10
11 SoccerPlayer tiffany = new SoccerPlayer();
12
13 erik.watch(daniel);
14
15 stella.speakTo(tiffany);
```



same Pylst type

```
16
17 tiffany.speakTo(erik);
18
19 ((Athlete) tiffany).speakTo(erik);
20
21 ((Person) tiffany).speakTo(erik);
22
23 ((Athlete) erik).speakTo(stella);
24
25 ((SoccerPlayer) erik).watch(tiffany);
```

on runtime it still will be SP, casting isnt there for tiffany

erik can be casted to athlete, as his Dynamic type is athlete

Dynamic type is athlete, SP's superclass is Athlete. so we can't cast erik down to SP cause athlete isnt neccessarily a SP

Line	Compile Time (Static)	Runtime (Dynamic)	Output
1	Person	CE (cant instantiate an interface)	
3	Athlete	SoccerPlayer	
5	SP	CE	
7	Person	Athlete	
9	Athlete	Athlete	
11	SP	SP	
13	Person.watch	Athelete.watch	ball is life
15	Athlete.speakTo		I love sports
17	SP.speakTo		join Globallers
19	Athlete.speakTo	SP.speakTo	
21	Person.speakTo	SP.speakTo	
23	Athlete.speakTo	Athlete.speakTo	I love sports
25	RE		