Name(s) _____ Period _____ Date _____

Extra Practice - Removing Elements

Check for Understanding

What is printed as a result of the following code segment?

```
ArrayList<Integer> zipCodes = new ArrayList<Integer>();
zipCodes.add(22032);
zipCodes.add(30351);
zipCodes.add(24301);
zipCodes.add(2, 7804);
zipCodes.remove(2);
zipCodes.remove(2);
System.out.println(zipCodes);
```

A. `[22032, 30351, 24301, 7804]`

B. `[22032, 30351, 7804, 24301]`

C. `[22032, 30351, 24301]`

D. `[22032, 30351]`

E. `[22032, 24301]`

AP Exam Prep

Consider the following method.

```
/** Removes all occurrences of dessertName from dessertNameList.
* @param dessertNameList a list of dessert names
* @param dessertName name of the dessert to be removed from dessertNameList
*/
public void removeDessertName(ArrayList<String> dessertNameList, String dessertName)
{
    /* missing implementation */
}
```

Which of the following can be used to replace **`/* missing implementation */`** so that **`removeDessertName`** will work as intended?

I.
```
for (String d : dessertNameList)
 {
     if (d.equals(dessertName))
         d.remove();
 }
```

II.
```
for (int k = 0; k < dessertNameList.size(); k++)
 {
     if (dessertNameList.get(k).equals(dessertName))
         dessertNameList.remove(k);
 }
```

III.
```
for (int k = dessertNameList.size() - 1; k >= 0; k--)
 {
     if (dessertNameList.get(k).equals(dessertName))
         dessertNameList.remove(k);
 }
```

A. I only

B. II only

C. III only

D. II and II only

E. I, II, and III

Extra Practice

Do This: The `removeScores` method is intended to remove all instances of `target` from the `ArrayList` object `scores` passed as a parameter. The method does not work as intended for all inputs.

```
public void removeScores(ArrayList<Integer> scores, int target)
{
    for (int j = 0; j < scores.size(); j++)
    {
        if (scores.get(j).equals(target))
        {
            scores.remove(j);
        }
    }
}
```

Assume that the `ArrayList` object `tests` and the `int` variable `low_score` have been properly declared and initialized. In which of the following cases will the method call `removeScores(tests, low_score)` fail to produce the intended result?

A. When `tests` is `[0, 1, 0, 3, 0, 5]` and `low_score` is `0`

B. When `tests` is `[2, 5, 0, 4, 8, 0]` and `low_score` is `0`

C. When `tests` is `[8, 8, 5, 8, 6, 6]` and `low_score` is `5`

D. When `tests` is `[2, 4, 6, 9, 9, 1]` and `low_score` is `1`

E. When `tests` is `[6, 6, 4, 2, 2, 5]` and `low_score` is `2`