BJP5 Self-Check 10.15: ArrayListMystery1

Consider the following method:

```java
public static void mystery1(ArrayList<Integer> list) {
    for (int i = list.size() - 1; i > 0; i--) {
        if (list.get(i) < list.get(i - 1)) {
            int element = list.get(i);
            list.remove(i);
            list.add(0, element);
        }
    }
    System.out.println(list);
}
```

Write the output produced by the method when passed each of the following ArrayLists:

[2, 6, 1, 8]	1,2,6,8
[30, 20, 10, 60, 50, 40]	10,30,40,20,60,50
[-4, 16, 9, 1, 64, 25, 36, 4, 49]	-4,1,25,4,16,9,64,36,49

Submit

✓ You passed 3 of 3 tests.

BJP5 Self-Check 10.17: ArrayListMystery3

Consider the following method:

```java
public static void mystery3(ArrayList<Integer> list) {
    for (int i = list.size() - 2; i > 0; i--) {
        int a = list.get(i);
        int b = list.get(i + 1);
        list.set(i, a + b);
    }
    System.out.println(list);
}
```

Write the output produced by the method when passed each of the following Arr

[72, 20]	[72, 20]
[1, 2, 3, 4, 5, 6]	[1, 20, 18, 15, 11, 6]
[10, 20, 30, 40]	[10, 90, 70, 40]

Submit

✓ You passed 3 of 3 tests.

BJP5 Exercise 10.7: removeDuplicates

Write a method removeDuplicates that takes as a parameter a sorted ArrayList of Str
from the list. For example, suppose that a variable called list contains the following values
{"be", "be", "is", "not", "or", "question", "that", "the", "to", "to"
removeDuplicates(list); the list should store the following values:
{"be", "is", "not", "or", "question", "that", "the", "to"}

Because the values will be sorted, all of the duplicates will be grouped together.

Type your solution here:

```java
1 public void removeDuplicates(ArrayList<String> given){
2     for (int i = 0; i < given.size() - 1; i++) {
3         if (given.get(i) == given.get(i + 1)) {
4             given.remove(i + 1);
5             i--;
6         }
7     }
8
9 }
```

This is a **method problem.** Write a Java method as described. Do not write a complete program or class; just the me

Submit

✓ You passed 5 of 5 tests.

BJP5 Self-Check 10.16: ArrayListMystery2

Consider the following method:

```java
public static void mystery2(ArrayList<Integer> list) {
    for (int i = list.size() - 1; i >= 0; i--) {
        if (i % 2 == 0) {
            list.add(list.get(i));
        } else {
            list.add(0, list.get(i));
        }
    }
    System.out.println(list);
}
```

Write the output produced by the method when passed each of the following ArrayLists:

[10, 20, 30]	20,10,20,30,30,20
[8, 2, 9, 7, 4]	8,7,8,2,9,7,4,4,2,8
[-1, 3, 28, 17, 9, 33]	33,28,33,-1,3,28,17,9,33,17,-1,33

Submit

✓ You passed 3 of 3 tests.

BJP5 Self-Check 10.18: ArrayListMystery4

Consider the following method:

```java
public static void mystery4(ArrayList<Integer> list) {
    for (int i = 0; i < list.size(); i++) {
        int element = list.get(i);
        list.remove(i);
        list.add(0, element + 1);
    }
    System.out.println(list);
}
```

Write the output produced by the method when passed each of the following ArrayLists:

[10, 20, 30]	[31, 21, 11]
[8, 2, 9, 7, 4]	[5, 8, 10, 3, 9]
[-1, 3, 28, 17, 9, 33]	[34, 10, 18, 29, 4, 0]

Submit

✓ You passed 3 of 3 tests.

BJP5 Exercise 10.14: removeShorterStrings

Write a method removeShorterStrings that takes an ArrayList of Strings as a parameter and
successive pair of values the shorter string in the pair. For example, suppose that an ArrayList calle
values: {"four", "score", "and", "seven", "years", "ago"} In the first pair, "four" an
"four". In the second pair, "and" and "seven", the shorter string is "and". In the third pair, "year
is "ago". Therefore, the call: removeShorterStrings(list); should remove these shorter string
"score", "seven", "years". If there is a tie (both strings have the same length), your method sh
the pair. If there is an odd number of strings in the list, the final value should be kept in the list.

Type your solution here:

```java
1 public void removeShorterStrings(ArrayList<String> given){
2     for (int i = 0; i < given.size()-1; i++){
3         if (given.get(i).length() <= given.get(i+1).length())
4             given.remove(i);
5         else
6             given.remove(i+1);
7     }
8
9 }
```

This is a **method problem.** Write a Java method as described. Do not write a complete program or class; just the method(s) abc

Submit

✓ You passed 5 of 5 tests.