

ArrayList - Lab

AP Computer Science A

1: Write a program called `WordLengths.java` with two functions: `getShortWords(ArrayList<String> list)` and `getLongWords(ArrayList<String> list)`. The function `getShortWords(ArrayList<String> list)` is not to modify the parameter `list` and needs to return the `ArrayList` of Strings that are in `list` and have a length less than or equal to 3. The function `getLongWords(ArrayList<String> list)` is not to modify the parameter `list` and needs to return the `ArrayList` of Strings that are in `list` and have a length larger than or equal to 6.

The `main` function needs to read in several strings until it reads in the “END” String. After this is done, have this method use the functions `getShortWords` and `getLongWords` to output two different lists.

```
mrdagler:> java WordLengths
Enter the words (Stop at END):
Hi
Franklin
DiscGolf
Lisa
Clay
Cat
END
The short words: [Hi, Cat]
The long words: [Franklin, DiscGolf]
```

2: Write a method `markLength4` that takes an `ArrayList` of Strings as a parameter and places a string of four asterisks “****” in front of every string of length 4. This method needs to be part of the class `MarkLength4` that has a `main` method that first reads in an integer value for the number of Strings to read in followed by that many strings. Once this is done, call the function `markLength4` with the `ArrayList` of Strings as the parameter and then output the results. Call the file *MarkLength4.java*

```
mrdagler:> java MarkLength4
How many words are in your list: 9
Enter the words:
This
is
lots
of
fun
for
every
Java
programmer
[****, This, is, ****, lots, of, fun, for, every, ****, Java, programmer]
```

Notice that you leave the original strings in the list, “this”, “lots”, “Java”, but include the four-asterisk string in front of each to mark it.

3: Write a method `removeBadPairs` that accepts an `ArrayList` of an even number of integers and removes any adjacent pair of integers in the list if the left element of the pair is larger than the right element of the pair. Every pair’s left element is an even-numbered index in the list, and every pair’s right element is an odd index in the list. For example, suppose a variable called `list` stores the following element values: `[3, 7, 9, 2, 5, 5, 8, 5, 6, 3, 4, 7, 3, 1]`. We can think of this list as a sequence of pairs: (3, 7), (9, 2), (5, 5), (8, 5), (6, 3), (4, 7), (3, 1). The pairs (9, 2), (8, 5),

(6, 3), and (3, 1) are "bad" because the left element is larger than the right one, so these pairs should be removed. So the call of `removeBadPairs(list)` would change the list to store the following element values: [3, 7, 5, 5, 4, 7]. Call the main class *RemoveBadPairs.java*

This program needs to first asks the user how many pairs to read in, then reads in each pair, and then outputs all of the pairs excluding the bad ones. Call the file *RemoveBadPairs.java*

```
mrdaqler:> java RemoveBadPairs
How many points are you going to enter: 7
Enter the points: x y:
3 7
9 2
5 5
8 5
6 3
4 7
3 1
The list without the bad pairs: [3, 7, 5, 5, 4, 7]
```