The materials included in these files are intended for noncommercial use by AP teachers for course and exam preparation; permission for any other use must be sought from the Advanced Placement Program®. Teachers may reproduce them, in whole or in part, in limited quantities, for face-to-face teaching purposes but may not mass distribute the materials, electronically or otherwise. This permission does not apply to any third-party copyrights contained herein. These materials and any copies made of them may not be resold, and the copyright notices must be retained as they appear here.
Complete method `numWordsOfLength` below.

```java
// postcondition: returns the number of words in this WordList that
// are exactly len letters long
public int numWordsOfLength(int len) {
    int count = 0;
    for (int i = 0; i < myList.size(); i++) {
        if (((String) myList.get(i)).length() == len)
            count++;
    }
    return count;
}
```

Part (b) begins on page 6.
Complete method `removeWordsOfLength` below.

```java
// postcondition: all words that are exactly len letters long
// have been removed from this WordList, with the
// order of the remaining words unchanged
public void removeWordsOfLength(int len) {

    for (int i = 0; i < myList.size(); i++) {
        if ((String) myList.get(i).equals(len)) {
            myList.remove(i);
            i--; // Adjust i to account for the removal
        }
    }
}
```

**GO ON TO THE NEXT PAGE.**
Complete method `numWordsOfLength` below.

```java
// postcondition: returns the number of words in this WordList that
// are exactly len letters long
public int numWordsOfLength(int len)
{
    int c = 0;
    for (int i = 0; i < myList.Length(); i++)
    {
        if (myList[i].Length() == len)
            c++;
    }
    return c;
}
```

Part (b) begins on page 6.
Complete method `removeWordsOfLength` below.

```java
// postcondition: all words that are exactly `len` letters long
// have been removed from this WordList, with the
// order of the remaining words unchanged
public void removeWordsOfLength(int len)
{
    int c = 0;
    for (int i = 0; i < myList.length; i++)
    {
        if (myList[i].length == len)
        {
            myList.remove(i);
        }
    }
}
```

GO ON TO THE NEXT PAGE.
Complete method removeWordsOfLength below.

```java
// postcondition: all words that are exactly len letters long
// have been removed from this WordList, with the
// order of the remaining words unchanged
public void removeWordsOfLength(int len)

    for (int k = 0; k < MyList.length; k++)
        if (len == MyList[k].length)
            MyList.set(k,"");
    k++;

    System.out.println(MyList);
```
Complete method numWordsOfLength below.

// postcondition: returns the number of words in this WordList that
// are exactly len letters long
public int numWordsOfLength(int len)
    int count;
    int listLength;
    for (int k = 0; k < listLength; k++)
        if (k == myList[k].length)
            count = count + 1;
    System.out.println("The total number of words that have " + len " letters is " + count + ");

Part (b) begins on page 6.