Lab Goal : This lab was designed to review basic class creation, the use of ArrayList, and sorting and searching. This lab also focuses on storing references in an ArrayList and manipulating the objects to which the ArrayList locations refer.

Lab Description : Write a class `LongPalindrome` that contains three methods. Method `getAllPalins` will return an `ArrayList` of all palindromes contained within word, maintaining an alphabetically sorted order. Method `insertLocation` will return the insert location for a new Palin. Method `getLongestPalin` will return the single longest palindrome contained within word.

Files Needed ::
Palin.java
LongPalindrome.java
LongPalindromeRunner.jav

```
public class LongPalindrome
{
  /*
   *method getAllPalins will return an array list that contains all of the palindromes contained within word
   *palindromes must have a minimum length of 2
   * getAllPalins must call insertLocation
  */
   public static ArrayList<Palin> getAllPalins(String word)
   {
        return null;
   }

  /*
   *method insertLocation will look at the list and determine where to put the new Palin so that the sorted order is maintained
   */
   private static int insertLocation( ArrayList<Palin> list, Palin pal )
   {
        return 0;
   }

  /*
   *method getLongestPalin will return the longest palindrome contained within word
   *a palindrome must have a minimum length of 2
   * getLongestPalin must call getAllPalins
   */
   public static Palin getLongestPalin(String word)
   {
        return null;
   }
}
```

Sample Data :

```
"bobcat"
"bobracecarcat"
"alligatorslikegroovycatslikemadamandrooroo"
"idrovetothelibraryinmyracecartolearnaboutpanama"
```

Sample Output :

```
[bob]
bob

[aceca, bob, cec, racecar]
racecar

[ada, ama, ll, madam, oo, oo, oo, ooroo, oro, roor]
madam
```

```
[aceca, ama, ana, cec, racecar, rar, tot]
racecar
```