```
//S Harshini-185001058
/*1.to find the index of the longest word in a given string*/
import java.util.Scanner;
import java.lang.*;
class Longest
{
public static void main(String argv[])
{
 Scanner in=new Scanner(System.in);
 System.out.println("enter a string");
 String str=in.nextLine();
 String words[]=str.split(" ");
 int n=words.length;
 int max=0,sum=0,c=0;
 for(int i=0;i<n;i++)
 {
  if(words[i].length()>max)
   max=words[i].length();
 }
 for(int j=0;j<n;j++)
 {
  if(max==words[j].length())
  {
   break;
  }
  sum=sum+words[j].length();
  c=c+1;
 }
 System.out.println("the index position of the longest word is "+(sum+c));
}
}

/*sample input/output
cs1058@u6:~/Desktop/harsh-java$ javac Longest.java
cs1058@u6:~/Desktop/harsh-java$ java Longest
enter a string
hii this is harshini nice meetingggg you
the index position of the longest word is 26

*/

/*2.To check if a word is a palindrome or not*/
import java.lang.*;
import java.util.Scanner;
class Palindrome
{
public static void main(String argv[])
{
 int c;
 Scanner in=new Scanner(System.in);
 System.out.println("enter a string");
 StringBuffer str=new StringBuffer("");
 str.append(in.nextLine());
 StringBuffer revstr=new StringBuffer("");
 revstr.append(str);
 str.reverse();
 String str1=str.toString();
 String revstr1=revstr.toString();
 if(str1.compareTo(revstr1)==0)
  System.out.println("Given string is a palindrome");
 else
  System.out.println("Given string is not a palindrome");
}
}

/* Sample input/output
cs1058@u6:~/Desktop/harsh-java$ javac Palindrome.java
cs1058@u6:~/Desktop/harsh-java$ java Palindrome
enter a string
malayalam
Given string is a palindrome
cs1058@u6:~/Desktop/harsh-java$ java Palindrome
enter a string
hello
Given string is not a palindrome
*/


/*3.Chech if a given string is a pangram and count the no of alphabets in it.
*/
import java.util.Scanner;
class Pangram
{
public static void main(String argv[])
{
  char c;
  int a;
  Scanner in=new Scanner(System.in);
  System.out.println("enter a string");
  String str=in.nextLine();
  int l=str.length();
  int[] arr=new int[26];
  for(int k=0;k<26;k++)
   arr[k]=0;
  for(int i=0;i<l;i++)
  {
   c=str.charAt(i);
   if(c!=' ')
   {
    a=(int) c;
    a=a-97;
    arr[a]=arr[a]+1;
   }
  }
 for(int h=0;h<26;h++)
{
if(arr[h]==0)
{
            System.out.println("the given string is not a pangram");
            break;
}
}
  System.out.println("the occurrences of alphabets are");
  for(int j=0;j<26;j++)
  {
   a=97+j;
   c=(char) a;
   System.out.println(c+" - "+arr[j]);
  }
}
}


/*
Sample input/output

cs1058@u6:~/Desktop/harsh-java$ javac Pangram.java
cs1058@u6:~/Desktop/harsh-java$ java Pangram
enter a string
```

```
the quick brown fox jumps over the lazy dogssssss
the occurrences of alphabets are
a - 1
b - 1
c - 1
d - 1
e - 3
f - 1
g - 1
h - 2
i - 1
j - 1
k - 1
l - 1
m - 1
n - 1
o - 4
p - 1
q - 1
r - 2
s - 7
t - 2
u - 2
v - 1
w - 1
x - 1
y - 1
z - 1
cs1058@u6:~/Desktop/harsh-java$ java Pangram
enter a string
harshini
the given string is not a pangram
the occurrences of alphabets are
a - 1
b - 0
c - 0
d - 0
e - 0
f - 0
g - 0
h - 2
i - 2
j - 0
```

```
k - 0
l - 0
m - 0
n - 1
o - 0
p - 0
q - 0
r - 1
s - 1
t - 0
u - 0
v - 0
w - 0
x - 0
y - 0
z - 0
*/
```