Rajalakshmi Engineering College

Name: DARSHAN M
Email: 241501040@rajalakshmi.edu.in
Roll no: 241501040
Phone: 9360697087
Branch: REC
Department: AI & ML - Section 1
Batch: 2028
Degree: B.E - AI & ML

Scan to verify results

2024_28_III_OOPS Using Java Lab

REC_2028_OOPS using Java_Week 4_CY

Attempt : 1
Total Mark : 40
Marks Obtained : 40

Section 1 : Coding

1. Problem Statement

In a university library, librarians need to track the usage of special characters in students' notes.

To help them, you are asked to write a program that counts the number of specific symbols in each passage of text.

The symbols of interest are:

Exclamation marks (!)Colons (:)Semicolons (;)

Input Format

The first line of input contains an integer T, representing the number of test cases (passages).

Each of the next T lines contains a single passage of text.

Output Format

For each test case, print three integers separated by spaces, representing the number of exclamation marks, colons, and semicolons in the passage.

The first line of output corresponds to the first passage, the second line to the second passage, and so on.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1
Hello! How are you
Output: 1 0 0

Answer

```
// You are using Java
import java.util.*;
public class Main{
   public static void main(String args[]){
      Scanner s = new Scanner(System.in);
      int a=s.nextInt();
      s.nextLine();
      while(a-->0){
      String st =s.nextLine();
      int i=0,c=0,se=0;
      for(char ch: st.toCharArray()){

         if(ch=='!'){
            i++;
         }
         else if(ch==':'){
            c++;
         }
         else if(ch==';'){
            se++;
         }
```

```
        }
        System.out.println(i+" "+c+" "+se);
        }


    }
}
```

Status : Correct ***Marks : 10/10***

2. Problem Statement

Neha is analyzing text messages to identify words that have repeated characters. A word is considered “repetitive” if any character appears more than once in that word.

Your task is to write a program that extracts all words that contain repeated characters from a given sentence.

If no such word exists, print "No repetitive words found".

Input Format

The input contains a single line containing a sentence with multiple words.

Output Format

The output prints all words that contain repeated characters separated by a space.

If no word contains repeated characters, print "No repetitive words found".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: letter balloon apple tree
Output: letter balloon apple tree

Answer

```
// You are using Java
import java.util.*;
public class Main{
    public static void main(String args []){
        Scanner s = new Scanner(System.in);
        String sen =s.nextLine().trim();
        String [] words =sen.split(" ");
        boolean found=false;
        for(String w : words){
            boolean repated=false;
            for(int i=0;i<w.length();i++){
                for(int j=i+1;j<w.length();j++){
                    if(w.charAt(i)==w.charAt(j)){
                        repated=true;
                        break;
                    }
                }
            }
            if(repated){
                found=true;
                System.out.print(w+" ");
                }
        }
            if(!found){
                System.out.println("No repetitive words found");
            }


    }
}
```

Status : Correct **Marks : 10/10**

3. Problem Statement

A bookstore wants to analyze the titles of books to determine their longest word in each title. This helps in designing banners and covers.

Your task is to write a program that, given a sentence (book title), finds and prints the longest word. If multiple words have the same maximum length,

print the first one.

Input Format

The input contais a single line containing a sentence representing the book title.

Output Format

The output prints a string representing the longest word in the sentence (book title).

Refer to the sample output for formatting specifications.

Sample Test Case

Input: The Chronicles of Narnia
Output: Chronicles

Answer

```
import java.util.*;  // Import the Scanner class

public class Main {
  public static void main(String[] args) {
    Scanner s = new Scanner(System.in); // Create Scanner object

    // Read the book title
    String title = s.nextLine().trim();

    // Split the title into words
    String[] words = title.split(" ");

    // Variables to track longest word
    String longestWord = "";
    int maxLen = 0;

    // Loop through each word
    for (String word : words) {
      if (word.length() > maxLen) {
        maxLen = word.length();
        longestWord = word;
      }
```

```
    }

        // Print the longest word
        System.out.println(longestWord);
    }
}
```

Status : Correct ***Marks : 10/10***

4. Problem Statement

Riya is preparing for a vocabulary test. Her teacher told her to focus on long words in her practice sentences, specifically words that have at least 5 letters.

Riya wants to write a program that will help her identify such words quickly.

Your task is to help Riya by printing all the words in a given sentence that have a length greater than or equal to 5.

If no such word exists, display "No long words found".

Input Format

The input contains a single line containing a sentence with multiple words.

Output Format

The output prints all words having length ≥ 5, separated by a space.

If no such word is found, print "No long words found".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: The quick brown fox jumps over the lazy dog
Output: quick brown jumps

Answer

```
// You are using Java
import java.util.Scanner;

public class Main {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        // Read the input sentence
        String sentence = sc.nextLine().trim();

        // Split the sentence into words
        String[] words = sentence.split(" ");

        // To track if we found any long word
        boolean found = false;

        // Loop through each word
        for (String word : words) {
            if (word.length() >= 5) {
                System.out.print(word + " ");
                found = true;
            }
        }

        // If no long word was found
        if (!found) {
            System.out.print("No long words found");
        }
    }
}
```

Status : Correct **Marks : 10/10**