Rajalakshmi Engineering College

Name: SANTHOSH S
Email: 241801251@rajalakshmi.edu.in
Roll no: 241801251
Phone: 9790911586
Branch: REC
Department: AI & DS - Section 3
Batch: 2028
Degree: B.E - AI & DS

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 college, students are required to create unique usernames for accessing the digital library.

The librarian needs your help to verify whether the usernames entered by students are valid.

A username is considered valid if:

It contains only letters (a–z, A–Z) and digits (0–9).Its length is between 5 and 15 characters (inclusive).It must start with a letter (not a digit).

Your task is to determine whether each username in the list is valid or not.

Input Format

The first line of input contains an integer T, representing the number of usernames to check.

The next T lines each contain a string S, representing a username.

Output Format

For each username S, the output print "YES" if it is valid.

Otherwise, the output print "NO".

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1
Alice123
Output: YES

Answer

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

public class Main{
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      int T = Integer.parseInt(sc.nextLine()); // Number of usernames

      for (int i = 0; i < T; i++) {
         String username = sc.nextLine().trim();
         if (isValidUsername(username)) {
            System.out.println("YES");
         } else {
            System.out.println("NO");
         }
      }

      sc.close();
   }

   public static boolean isValidUsername(String username) {
```

```
        int length = username.length();

        // Condition 1: Length should be between 5 and 15
        if (length < 5 || length > 15) {
            return false;
        }

        // Condition 2: Must start with a letter
        if (!Character.isLetter(username.charAt(0))) {
            return false;
        }

        // Condition 3: Should contain only letters and digits
        for (int i = 0; i < length; i++) {
            char ch = username.charAt(i);
            if (!Character.isLetterOrDigit(ch)) {
                return false;
            }
        }

        return true;
    }
}
```

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

2. Problem Statement

A library wants to analyze book titles to count the number of words that start with an uppercase letter. This helps the library track proper nouns and important words in titles.

Your task is to write a program that, for each given title, counts and prints the number of words that start with an uppercase letter.

Input Format

The first line contains an integer T, representing the number of book titles.

Each of the next T lines contains a single title (string).

Output Format

For each title, the output print a single integer representing the number of words starting with an uppercase letter.

Refer to the sample output for formatting specifications.

Sample Test Case

Input: 1
The Chronicles of Narnia

Output: 3

Answer

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

public class Main{
   public static void main(String[] args) {
      Scanner sc = new Scanner(System.in);
      int T = Integer.parseInt(sc.nextLine()); // Number of book titles

      for (int i = 0; i < T; i++) {
         String title = sc.nextLine().trim();
         String[] words = title.split(" ");
         int count = 0;

         for (String word : words) {
            if (!word.isEmpty() && Character.isUpperCase(word.charAt(0))) {
               count++;
            }
         }

         System.out.println(count);
      }

      sc.close();
   }
}
```

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

3. 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.*;

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

```
        String[] words = sentence.split(" ");
        boolean found = false;

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

        if (!found) {
            System.out.println("No long words found");
        }

        sc.close();
    }
}
```

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

4. 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 sc = new Scanner(System.in);
        String sentence = sc.nextLine().trim();
        String[] words = sentence.split(" ");
        boolean found = false;

        for (String word : words) {
            if (hasRepeatedChars(word)) {
                System.out.print(word + " ");
                found = true;
            }
        }

        if (!found) {
            System.out.println("No repetitive words found");
        }

        sc.close();
    }

    // Function to check if a word has repeated characters
    private static boolean hasRepeatedChars(String word) {
        int[] freq = new int[256]; // ASCII characters
        for (int i = 0; i < word.length(); i++) {
            char ch = word.charAt(i);
            freq[ch]++;
            if (freq[ch] > 1) {
                return true;
```

```
            }
        }
        return false;
    }
}
```

Status : Correct   *Marks : 10/10*