

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 4\_Q1

Attempt : 1

Total Mark : 10

Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

In a publishing company, editors often need to quickly analyze passages of text to check for punctuation usage. To assist them, you are asked to write a program that counts the number of specific punctuation marks in each passage.

The punctuation marks of interest are:

Commas (,)Periods (.)Question marks (?)

##### *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 commas, periods, and question marks 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, world. How are you?  
Output: 1 1 1

### ***Answer***

```
import java.util.Scanner;

class PunctuationCounter {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int T = Integer.parseInt(scanner.nextLine());

        for (int i = 0; i < T; i++) {
            String passage = scanner.nextLine();
            int commas = 0, periods = 0, questions = 0;

            for (char c : passage.toCharArray()) {
                if (c == ',') commas++;
                else if (c == '.') periods++;
                else if (c == '?') questions++;
            }

            System.out.println(commas + " " + periods + " " + questions);
        }

        scanner.close();
    }
}
```

**Status :** Correct

**Marks :** 10/10

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 4\_Q2

Attempt : 1

Total Mark : 10

Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

Anu is developing a tool for a conference registration system. Participants submit keywords related to their fields of interest. The organizer wants to sort these keywords alphabetically to generate tags for session grouping.

Write a program that accepts at least five keywords as input arguments and outputs them in sorted alphabetical order.

##### *Input Format*

The first line of input contains an integer n, representing the number of keywords.

The second line of input contains n space-separated keywords (string).

##### *Output Format*

The output prints n space separated strings representing the sorted keyword in alphabetical order.

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: 5

Blockchain Cloud AI Data Cybersecurity

Output: AI Blockchain Cloud Cybersecurity Data

### ***Answer***

```
import java.util.*;

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

        int n = Integer.parseInt(scanner.nextLine());
        String[] keywords = scanner.nextLine().split(" ");

        Arrays.sort(keywords);

        for (int i = 0; i < keywords.length; i++) {
            System.out.print(keywords[i]);
            if (i != keywords.length - 1) {
                System.out.print(" ");
            }
        }

        scanner.close();
    }
}
```

**Status : Correct**

**Marks : 10/10**

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 4\_Q3

Attempt : 1

Total Mark : 10

Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

Bechan Chacha is seeking help to filter out valid mobile numbers from a list provided by his crush. He can only pick his crush's number if the list contains valid mobile numbers.

A mobile number is considered valid if:

It has exactly 10 digits. It consists only of numeric values (0–9). It does not begin with zero.

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

#### *Input Format*

The first line contains an integer T, representing the number of mobile numbers

to check.

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

#### ***Output Format***

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

Otherwise, print "NO".

Refer to the sample output for formatting specifications.

#### ***Sample Test Case***

Input: 1  
9876543210

Output: YES

#### ***Answer***

```
import java.util.*;  
  
class MobileNumberValidator {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        int T = Integer.parseInt(scanner.nextLine());  
        for (int i = 0; i < T; i++) {  
            String s = scanner.nextLine().trim();  
  
            if (isValidMobileNumber(s)) {  
                System.out.println("YES");  
            } else {  
                System.out.println("NO");  
            }  
        }  
        scanner.close();  
    }  
  
    public static boolean isValidMobileNumber(String s) {  
        if (s.length() != 10) {
```

```
        return false;
    }

    for (char c : s.toCharArray()) {
        if (!Character.isDigit(c)) {
            return false;
        }
    }

    return s.charAt(0) != '0';
}
```

*Status : Correct*

*Marks : 10/10*

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 4\_Q5

Attempt : 1

Total Mark : 10

Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

In a secure banking system, customers are required to create PIN codes for accessing their accounts. The bank wants to validate these PIN codes before accepting them.

A PIN code is considered valid if:

It consists of exactly 4 digits. All characters must be numeric (0–9). It cannot contain all identical digits (e.g., 1111 is invalid).

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

##### *Input Format*

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

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

### ***Output Format***

For each PIN code 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

1234

Output: YES

### ***Answer***

```
import java.util.*;  
  
class PinValidator {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        int T = Integer.parseInt(scanner.nextLine());  
  
        for (int i = 0; i < T; i++) {  
            String pin = scanner.nextLine().trim();  
            if (isValidPin(pin)) {  
                System.out.println("YES");  
            } else {  
                System.out.println("NO");  
            }  
        }  
        scanner.close();  
    }  
  
    public static boolean isValidPin(String pin) {  
        if (pin.length() != 4) {  
            return false;  
        }  
    }  
}
```

```
}

for (char c : pin.toCharArray()) {
    if (!Character.isDigit(c)) {
        return false;
    }
}

char first = pin.charAt(0);
if (pin.charAt(1) == first && pin.charAt(2) == first && pin.charAt(3) == first) {
    return false;
}

return true;
}
```

**Status :** Correct

**Marks :** 10/10

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### 2028\_REC\_OOPS using Java\_Week 4\_Q4

Attempt : 1

Total Mark : 10

Marks Obtained : 10

#### Section 1 : Coding

##### 1. Problem Statement

Arjun is learning how to filter words from a sentence based on grammar rules. He wants to identify the valid words in a sentence.

A word is considered valid if it satisfies all these conditions:

The word contains only alphabets (a–z, A–Z). The word length is at least 2 characters. The word should not contain digits or special characters.

Your task is to read a sentence and print all the valid words in it.

##### *Input Format*

The input contains a single line containing a sentence S.

##### *Output Format*

The output prints all the valid words separated by spaces.

If no valid word exists, print "No valid words."

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: Hello world1 123 ab" @#\$ Hi

Output: Hello Hi

### ***Answer***

```
import java.util.*;  
  
class ValidWordFilter {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        String sentence = scanner.nextLine();  
        scanner.close();  
  
        String[] words = sentence.split(" ");  
        List<String> validWords = new ArrayList<>();  
  
        for (String word : words) {  
            if (isValidWord(word)) {  
                validWords.add(word);  
            }  
        }  
  
        if (validWords.isEmpty()) {  
            System.out.print("No valid words.");  
        } else {  
            for (int i = 0; i < validWords.size(); i++) {  
                System.out.print(validWords.get(i));  
                if (i != validWords.size() - 1) {  
                    System.out.print(" ");  
                }  
            }  
        }  
    }  
}
```

```
private static boolean isValidWord(String word) {  
    if (word.length() < 2) {  
        return false;  
    }  
    for (char c : word.toCharArray()) {  
        if (!Character.isLetter(c)) {  
            return false;  
        }  
    }  
    return true;  
}
```

*Status : Correct*

*Marks : 10/10*

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

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**

Meera is practicing her English vocabulary. She wants to focus on words that have more vowels in them, as they help improve her pronunciation. She decides to extract only those words from a sentence that contain at least two vowels.

Your task is to help Meera by writing a program that finds such words from the given sentence.

##### ***Input Format***

The input contains a string representing the sentence.

##### ***Output Format***

The output prints all the words that contain at least two vowels, separated by a space.

If no such word exists, print "No words with two vowels".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: This is an example sentence

Output: example sentence

### ***Answer***

```
import java.util.*;

class WordsWithTwoVowels {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String sentence = scanner.nextLine().trim();
        scanner.close();

        String[] words = sentence.split(" ");
        List<String> resultWords = new ArrayList<>();

        for (String word : words) {
            if (countVowels(word) >= 2) {
                resultWords.add(word);
            }
        }

        if (resultWords.isEmpty()) {
            System.out.print("No words with two vowels");
        } else {
            for (int i = 0; i < resultWords.size(); i++) {
                System.out.print(resultWords.get(i));
                if (i != resultWords.size() - 1) {
                    System.out.print(" ");
                }
            }
        }
    }
}
```

```
private static int countVowels(String word) {  
    int count = 0;  
    for (char c : word.toLowerCase().toCharArray()) {  
        if (c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u') {  
            count++;  
        }  
    }  
    return count;  
}
```

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**

```
import java.util.*;  
  
class TitleAnalyzer {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        int T = Integer.parseInt(scanner.nextLine());  
  
        for (int i = 0; i < T; i++) {  
            String title = scanner.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);  
        }  
    }  
}
```

**Status :** Correct

**Marks :** 10/10

### 3. 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***

```
import java.util.*;
class main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        int T = Integer.parseInt(scanner.nextLine());

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

```
}

private static boolean isValidUsername(String username) {
    int len = username.length();

    if (len < 5 || len > 15) {
        return false;
    }

    if (!Character.isLetter(username.charAt(0))) {
        return false;
    }

    for (char c : username.toCharArray()) {
        if (!Character.isLetterOrDigit(c)) {
            return false;
        }
    }

    return true;
}
```

**Status :** Correct

**Marks :** 10/10

#### 4. 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 contains 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.*;

class main {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String title = scanner.nextLine().trim();
        scanner.close();

        String[] words = title.split(" ");
        String longestWord = "";
        int maxLength = 0;

        for (String word : words) {
            if (word.length() > maxLength) {
                maxLength = word.length();
                longestWord = word;
            }
        }

        System.out.print(longestWord);
    }
}
```

*Status : Correct*

*Marks : 10/10*

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### REC\_2028\_OOPS using Java\_Week 4\_MCQ

Attempt : 1

Total Mark : 15

Marks Obtained : 14

#### Section 1 : MCQ

1. What is the output of the following code?

```
class Main
{
    public static void main(String args[])
    {
        StringBuffer c = new StringBuffer("Hello");
        c.delete(0,2);
        System.out.println(c);
    }
}
```

*Answer*

llo

Status : Correct

Marks : 1/1

2. What will be the output of the following program?

```
class Main {  
    public static void main(String[] args) {  
        String greet = "Welcome\n";  
        System.out.print("String: " + greet);  
        int length = greet.length();  
        System.out.print("Length: " + length);  
    }  
}
```

*Answer*

String: WelcomeLength: 8

**Status : Correct**

**Marks : 1/1**

3. What will be the output of the following code?

```
class Main {  
    public static void main(String args[]) {  
        StringBuffer sb = new StringBuffer("Hello");  
        System.out.println("buffer before = " + sb);  
        System.out.println("charAt(1) before = " + sb.charAt(1));  
        sb.setCharAt(1, 'i');  
        sb.setLength(2);  
        System.out.println("buffer after = " + sb);  
        System.out.println("charAt(1) after = " + sb.charAt(1));  
    }  
}
```

*Answer*

buffer before = HellocharAt(1) before = ebuffer after = HicharAt(1) after = i

**Status : Correct**

**Marks : 1/1**

4. What will be the output of the following program?

```
class Main {
```

```
public static void main(String[] args) {  
    String s = new String("5");  
    System.out.println(1 + 1111 + s + 1 + 1010);  
}  
}
```

**Answer**

1112511010

**Status :** Correct

**Marks :** 1/1

5. What will be the output of the following code?

```
class Main {  
    public static void main(String args[]) {  
        char c[] = {'j', 'a', 'v', 'a'};  
        String s1 = new String(c);  
        String s2 = new String(s1);  
        System.out.println(s1);  
        System.out.println(s2);  
    }  
}
```

**Answer**

javajava

**Status :** Correct

**Marks :** 1/1

6. What will be the output for the following code?

```
class Main {  
    public static void main(String[] args) {  
        String languages[] = { "C", "C++", "Java", "Python", "Ruby"};  
        for (String sample: languages) {  
            System.out.println(sample);  
        }  
    }  
}
```

**Answer**

CC++JavaPythonRuby

**Status : Correct**

**Marks : 1/1**

7. What will be the output of the following program?

```
class Main {  
    public static void main(String args[]) {  
        String name="Work Hard";  
        name.concat("Success");  
        System.out.println(name);  
    }  
}
```

**Answer**

Work Hard

**Status : Correct**

**Marks : 1/1**

8. Predict the output for the following code.

```
public class Main {  
    public static void main(String[] args) {  
        String a = "java";  
        char temp = a.charAt(1);  
        System.out.println(temp);  
    }  
}
```

**Answer**

a

**Status : Correct**

**Marks : 1/1**

9. What will be the output of the following program?

```
class Main {
```

```
public static void main(String[] args) {  
    String s1 = "EDUCATION";  
    String s2 = new String("EDUCATION");  
    String s3 = "EDUCATION";  
    if (s1 == s2) {  
        System.out.println("s1 and s2 equal");  
    }  
    else {  
        System.out.println("s1 and s2 not equal");  
    }  
    if (s1 == s3) {  
        System.out.println("s1 and s3 equal");  
    }  
    else {  
        System.out.println("s1 and s3 not equal");  
    }  
}
```

**Answer**

s1 and s2 not equals1 and s3 equal

**Status :** Correct

**Marks :** 1/1

10. Predict the output for the following code:

```
class Main {  
    public static void main(String args[]) {  
        StringBuffer sb = new StringBuffer("I Java!");  
        sb.insert(5, "like ");  
        System.out.println(sb);  
    }  
}
```

**Answer**

I Javlike a!

**Status :** Correct

**Marks :** 1/1

11. What will be the output of the following program?

```
class Main {  
    public static void main(String args[]) {  
        StringBuffer sb = new StringBuffer("Hello");  
        System.out.println("buffer = " + sb);  
        System.out.println("length = " + sb.length());  
        System.out.println("capacity = " + sb.capacity());  
    }  
}
```

**Answer**

buffer = Hello length = 5 capacity = 21

**Status :** Correct

**Marks :** 1/1

12. Predict the output for the following code.

```
class Main {  
    public static void main(String[] fruits) {  
        String fruit1 = new String("apple");  
        String fruit2 = new String("orange");  
        String fruit3 = new String("pear");  
        fruit3 = fruit1;  
        fruit2 = fruit3;  
        fruit1 = fruit2;  
        System.out.println(fruit1);  
        System.out.println(fruit2);  
        System.out.println(fruit3);  
    }  
}
```

**Answer**

appleappleapple

**Status :** Correct

**Marks :** 1/1

13. What will be the output of the following code?

```
class Main {  
    public static void main(String args[]) {  
        String s1 = "Hello i love java";  
        String s2 = new String(s1);  
        System.out.println((s1 == s2) + " " + s1.equals(s2));  
    }  
}
```

**Answer**

false true

**Status : Correct**

**Marks : 1/1**

14. What will be the output of the following program?

```
public class Main {  
    public static void main(String[] args) {  
        String str = "1234.34";  
        int a = Integer.parseInt(str);  
        System.out.println(a);  
    }  
}
```

**Answer**

1234

**Status : Wrong**

**Marks : 0/1**

15. Predict the output for the following code:

```
public class Main {  
    public static void main(String[] args) {  
        float a = 10.0f;  
        String temp = Float.toString(a);  
        System.out.println(temp);  
    }  
}
```

**Answer**

10.0

**Status :** Correct

**Marks :** 1/1

# Rajalakshmi Engineering College

Name: samyuktha ss

Email: 241001217@rajalakshmi.edu.in

Roll no:

Phone: 7539908242

Branch: REC

Department: IT - Section 2

Batch: 2028

Degree: B.E - IT

Scan to verify results



## 2024\_28\_III\_OOPS Using Java Lab

### **REC\_2028\_OOPS using Java\_Week 4\_PAH**

Attempt : 1

Total Mark : 40

Marks Obtained : 40

#### **Section 1 : Coding**

##### **1. Problem Statement**

Ravi is analyzing text messages for his research on typing patterns. He wants to count the number of uppercase letters, lowercase letters, and digits in a sentence to understand typing trends.

Your task is to help Ravi by writing a program that takes a sentence and prints the count of uppercase letters, lowercase letters, and digits.

##### ***Input Format***

The input contains a single line containing a sentence (string).

##### ***Output Format***

The output prints three integers separated by spaces:

- Number of uppercase letters
- Number of lowercase letters
- Number of digits

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: Hello World 123

Output: 2 8 3

### ***Answer***

```
import java.util.*;  
  
class TypingPatternAnalysis {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        String sentence = scanner.nextLine();  
        scanner.close();  
  
        int upperCaseCount = 0;  
        int lowerCaseCount = 0;  
        int digitCount = 0;  
  
        for (char c : sentence.toCharArray()) {  
            if (Character.isUpperCase(c)) {  
                upperCaseCount++;  
            } else if (Character.isLowerCase(c)) {  
                lowerCaseCount++;  
            } else if (Character.isDigit(c)) {  
                digitCount++;  
            }  
        }  
        System.out.print(upperCaseCount + " " + lowerCaseCount + " " + digitCount);  
    }  
}
```

**Status :** Correct

**Marks :** 10/10

## 2. Problem Statement

Sana is analyzing text for a secret code. She wants to find all words in a sentence that start and end with the same letter. These words are considered "special words" for her analysis.

Your task is to write a program that extracts and prints all words that start and end with the same letter (case-insensitive).

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

### ***Input Format***

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

### ***Output Format***

The output prints all words that start and end with the same letter separated by a space.

If no word satisfies the condition, print "No special words found".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: Anna went to the civic center

Output: Anna civic

### ***Answer***

```
import java.util.*;

class SpecialWordsFinder {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String sentence = scanner.nextLine();
        scanner.close();

        String[] words = sentence.split(" ");
        List<String> specialWords = new ArrayList<>();
```

```

for (String word : words) {
    if (word.length() >= 1) {
        char first = Character.toLowerCase(word.charAt(0));
        char last = Character.toLowerCase(word.charAt(word.length() - 1));

        if (first == last) {
            specialWords.add(word);
        }
    }
}

if (specialWords.isEmpty()) {
    System.out.print("No special words found");
} else {
    for (int i = 0; i < specialWords.size(); i++) {
        System.out.print(specialWords.get(i));
        if (i != specialWords.size() - 1) {
            System.out.print(" ");
        }
    }
}
}

```

*Status :* Correct

*Marks :* 10/10

### 3. Problem Statement

Riya is preparing a puzzle game for her friends. She wants to include a feature that highlights special words in a sentence – specifically, palindromic words (words that read the same forward and backward).

Your task is to help Riya by writing a program that extracts all palindrome words from the given sentence. If there are no palindromes, print "No palindromes found".

#### *Input Format*

The input contains a single string S representing a sentence.

### ***Output Format***

The output prints all palindromic words separated by a space.

If no palindrome exists, print "No palindromes found".

Refer to the sample output for formatting specifications.

### ***Sample Test Case***

Input: madam went to school

Output: madam

### ***Answer***

```
import java.util.*;

class PalindromeWordFinder {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        String sentence = scanner.nextLine().trim();
        scanner.close();

        String[] words = sentence.split(" ");
        List<String> palindromes = new ArrayList<>();

        for (String word : words) {
            if (isPalindrome(word)) {
                palindromes.add(word);
            }
        }

        if (palindromes.isEmpty()) {
            System.out.print("No palindromes found");
        } else {
            for (int i = 0; i < palindromes.size(); i++) {
                System.out.print(palindromes.get(i));
                if (i != palindromes.size() - 1) {
                    System.out.print(" ");
                }
            }
        }
    }
}
```

```

        }
    }

private static boolean isPalindrome(String word) {
    int left = 0;
    int right = word.length() - 1;

    while (left < right) {
        if (word.charAt(left) != word.charAt(right)) {
            return false;
        }
        left++;
        right--;
    }

    return true;
}
}

```

**Status :** Correct

**Marks :** 10/10

#### 4. Problem Statement

At a digital library, the system needs to analyze passages to identify the frequency of vowels, since they are key for linguistic research. You are asked to write a program that counts the number of vowels in each passage of text.

The vowels of interest are:

a, e, i, o, u (both uppercase and lowercase).

#### *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 a single integer representing the total number of vowels 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 World

Output: 3

### ***Answer***

```
import java.util.*;  
  
class VowelCounter {  
    public static void main(String[] args) {  
        Scanner scanner = new Scanner(System.in);  
        int T = Integer.parseInt(scanner.nextLine());  
        for (int i = 0; i < T; i++) {  
            String passage = scanner.nextLine();  
            int vowelCount = 0;  
  
            for (char c : passage.toCharArray()) {  
                if (isVowel(c)) {  
                    vowelCount++;  
                }  
            }  
            System.out.println(vowelCount);  
        }  
        scanner.close();  
    }  
  
    private static boolean isVowel(char c) {  
        c = Character.toLowerCase(c);  
        return c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u';  
    }  
}
```

```
        return c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u';  
    }  
}
```

**Status :** Correct

**Marks :** 10/10