Date: 07/02/2026

Duration: 1 hour 45

CONTINUOUS ASSESSMENT 1
II YEAR – SEMESTER 4
CSE23AE204 - PCP III
(B.Tech. E01,E02,E03,E05,E06)

SET F

Question 1: Problem Statement

A customer support analytics system analyzes **customer chat messages** to detect **symmetric emotional patterns**.

A message is considered **symmetric** if a **contiguous part of the message reads the same forwards and backwards** (palindrome).

Given a string message, your task is to identify and return the **longest symmetric substring** present in the message.

If multiple substrings of maximum length exist, return **any one of them**.

Example 1

Input:

message = "babad"

Output:

"bab"

Input Format

- A single string message representing a customer chat message.

Output Format

- A string representing the **longest palindromic substring** found in the message.

Question 2: Problem Statement

Problem Statement

A customer support analytics system analyzes **customer chat messages** to detect **symmetric emotional patterns**.

A message is considered **symmetric** if a **contiguous part of the message reads the same forwards and backwards** (palindrome).

Given a string message, your task is to identify and return the **longest symmetric substring** present in the message.

If multiple substrings of maximum length exist, return **any one of them**.

Input Format

- A single string message representing a customer chat message.

Output Format

- A string representing the **longest palindromic substring** found in the message.

Example 1

Input:

message = "babad"

Output:

"bab"

Explanation:

Possible palindromic substrings are "bab" and "aba".

Both have length 3. Returning "bab" is valid.

Date: 07/02/2026

Duration: 1 hour 45

CONTINUOUS ASSESSMENT 1
II YEAR – SEMESTER 4
CSE23AE204 - PCP III
(B.Tech. E01,E02,E03,E05,E06)

SET F

Question 3: Problem Statement

You are developing a **Smart Home Device Control System** for a modern IoT-based home.

The system processes **one device command at a time**, provided as a **space-separated string**.
Each device belongs to a specific category and performs operations differently.

The system must:

- Identify the device type from input
- Perform the device operation based on its configuration
- Use **abstraction and polymorphism** to support multiple device types
- Assign a unique device ID using a **static variable**
- Return the final computed output of the device operation

Device Operation Rules

Smart Light

- Power value represents brightness per level
- Final output is calculated based on brightness and level

Smart Fan

- Power value represents speed power
- Final output is calculated based on power and speed level

Smart AC

- Power value represents cooling capacity
- Final output is calculated based on capacity and mode level

Input Format

A **single space-separated string**.

Smart Light: LIGHT <DeviceName> <PowerValue> <Level>

Smart Fan: FAN <DeviceName> <PowerValue> <Level>

Smart AC: AC <DeviceName> <PowerValue> <Level>

Output Format

Return an **integer value** representing the **final device operation result**.

Example 1

Input

{ "device": "FAN Fan1 60 3" }

Output

180

Explanation

The fan computes the operation result based on its power value and speed level.