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 A

Question 1: Problem Statement

You are working on a **corporate audit platform** that validates numeric console reports generated by legacy systems.

Since the system cannot compare visual console output directly, every report must be returned as a **collection of text rows**, where:

- Each row is stored as a **string**
- The entire report is returned as an **array/list of strings**
- The layout must remain **vertically and horizontally symmetric**
- Numbers represent **column positions**

Your task is to generate the expected report layout based on a given integer N.

Input Format

- A single integer N representing the scale of the report.

Output Format

- Return an **array/list of strings**
- Each string represents **one row** of the report

Example 1

Input

3

Output

```
[
  "  1",
  " 121",
  "12321",
  " 121",
  "  1"
]
```

Question 2: Problem Statement

In a distributed network router, packets are placed in a queue represented by an integer array:

- A **positive integer** → valid packet
- A **zero (0)** → empty slot (no packet received at that millisecond)

To optimize throughput, the system must **shift all valid packets to the front of the queue**, while keeping their **original order** unchanged.

All empty slots (0s) must be moved to the **end**.

Routing Rule:

- **Do NOT create a new array**
- **Do NOT change relative order of valid packets**
- **Move zeros to the end in-place**

Input Format

queue → integer array (containing packets and empty slots)

8. Output Format

integer array → optimized in-place queue

Example 1

Input:

[0,5,0,3,12]

Output:

[5,3,12,0,0]

Example 2

Input:

[1,2,3]

Output:

[1,2,3]

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 A

Question 3: Problem Statement

You are developing a **Payroll Processing Module** for a corporate organization.

The system processes payroll details for **one employee at a time**, provided as a **space-separated input string**. Each employee belongs to a specific category and follows predefined salary rules.

The system must:

- Identify the employee type from input
- Compute gross salary using employee-specific rules
- Deduct tax
- Return the final payable salary

Each employee must be assigned a **unique employee ID**, which should be generated internally using a **static variable**.

Salary Calculation Rules

Permanent Employee

gross = baseSalary + hra − pf

tax = 10% of gross

net = gross − tax

Contract Employee

gross = baseSalary + allowance

tax = 10% of gross

net = gross − tax

Example 1

Input

PERMANENT Ram 50000 10000 5000

Output

49500.0

Explanation

gross = 50000 + 10000 − 5000 = 55000

tax = 10% of 55000 = 5500

net = 49500

Input Format

A **single space-separated string**.

Permanent Employee

PERMANENT <EmployeeName> <BaseSalary> <HRA> <PF>

Contract Employee

CONTRACT <EmployeeName> <BaseSalary> <Allowance>

Output Format

Return a **double value** representing the **net salary after tax deduction**.