Roll Number:

Thapar Institute of Engineering and Technology, Patiala
Computer Science & Engineering Department

BE CoE Third Year
EST- May 26, 2025
Time: 3 Hours; MM: 40

Course Code: UCS503/UCS318
Course Name: Software Engineering
Instructors: Ashima Singh

NOTE: Attempt FOUR Questions Only.

Qno	Questions	Marks	CO	BL
Q1	a) Discuss Extreme Programming with a suitable diagram while elaborating on key phases and related activities. Discuss in detail Spike Solutions, Pair Programming, Velocity, Test driven development b) Create User Stories and User Cards for **E-commerce platform** offering three features such as *Add a Book, Delete a Book, and Issue a Book* in the context of a bookstore admin or seller dashboard. Also, give the Front of the Card along with the Back of the Card giving Success and failure messages.	4,6	CO1	L2 L3
Q2	The GreenTech Institute of Technology (GIT) implemented a comprehensive Net Zero Energy Building (NZEB) strategy by installing a 270 kWp rooftop solar PV system coupled with a 300 kWh lithium-ion battery storage system. This intelligent energy ecosystem operates through seven distinct stages managed by a smart Energy Management System (EMS).During peak solar conditions, the system enters "Surplus Generation" mode, channeling excess power to battery storage. When batteries reach capacity, it transitions to "Grid Export" state, selling energy back through net metering agreements. The "Direct Supply" state activates when solar generation precisely matches building demand, ensuring optimal renewable utilization. During insufficient solar periods, the system prioritizes "Battery Discharge" mode when storage levels permit. As battery State-of-Charge (SoC) drops below critical thresholds, "Grid Import" state engages to maintain continuous supply. The "Islanded Operation" mode provides emergency resilience during grid outages, utilizing available solar and battery resources independently.A dedicated "Maintenance" state ensures system reliability during scheduled inspections or fault conditions. The EMS continuously monitors generation patterns, load demands, and storage status, enabling seamless transitions between operational stages. **a) Draw the State chart diagram of the above case** **b) Give a suitable architectural style best suited to the case. Discuss in detail the architectural style with a suitable diagram.**	6,4	CO2 CO3	L3 L4
Q3	a) A software development project is estimated to have 20000000 lines of code. The project is classified as an Embedded project with a highly experienced development team. The project schedule is very tight. Calculate the effort, and development time using the COCOMO model. b) **Consider the following Precedence Table for the optimal scheduling of tasks.** Task A: UI Design, Task B: Backend Development, Task C: Database Setup, Task D: API Integration, Task E: User Authentication Module, Task F: Expense Tracking Feature, Task G: Graphical Reports Module, Task H: Notifications Integration	4,6	CO5	L3, L4

PTO

Activity	A	B	C	D	E	F	G	H
Immediate Predecessor	-	-	A	B	C	D,E	F	F
Expected Time (days)	15	20	10	25	15	30	20	15

Prepare an Activity-On-Arc Network Diagram—legend to be used for AOA.

Early Start	Early Finish	Slack/Delay
Late Start	Late Finish	

(i) Create the Activity Network graph for this project. Show the calculation on Forward Pass and Backward Pass.

(ii) Calculate the Early Finish Time (EFT), Early Start Time (EST), Latest Start Time (LST) and Latest Finish Time (LFT) for all activities. Show in Table

Activity	ES	EF	LS	LF	Slack

(iii) Give the Critical Path, Critical time, and Critical Activities of that path.

Q4 — **6,4** — **CO3** — **L3**

a). Consider the code snippet and answer the following questions

```
1.  int main() {
2.  float a, b, c;
3.   printf("Enter three sides of the triangle: ");
4.   scanf("%f %f %f", &a, &b, &c);
5.    // Check for triangle validity
6.    if (a + b <= c || a + c <= b || b + c <= a) {
7.     printf("Not a valid triangle.\n");
8.      } else {
9.        // Check for equilateral
10.       if (a == b && b == c) {
11.          printf("Equilateral triangle.\n");
12.       }
13. // Check for isosceles
14.  else if (a == b || b == c || a == c)
15.   printf("Isosceles triangle.\n");
16.    }
17.    // Otherwise scalene
18.   else {
19.     printf("Scalene triangle.\n");
20.       }
21.    }
22.    return 0;
23. }
```

i) Create the context flow graph for the code.

ii) Calculate the cyclomatic complexity using three different methods.

iii) Identify the linearly independent paths and corresponding test cases.

b) Design Test cases for the above case *"Decision on the Type of Triangle"* using the decision table approach.

Q5.. — **5,5** — **CO4** — **L2, L3**

a)Discuss in detail the **key components of API architecture** and explain various **API design styles**. Illustrate your answer with suitable examples.

b)Discuss common types of test cases used in **API Testing**. Illustrate your answer with test scenarios for the **GET /orders/{id}** endpoint in an eCommerce application.