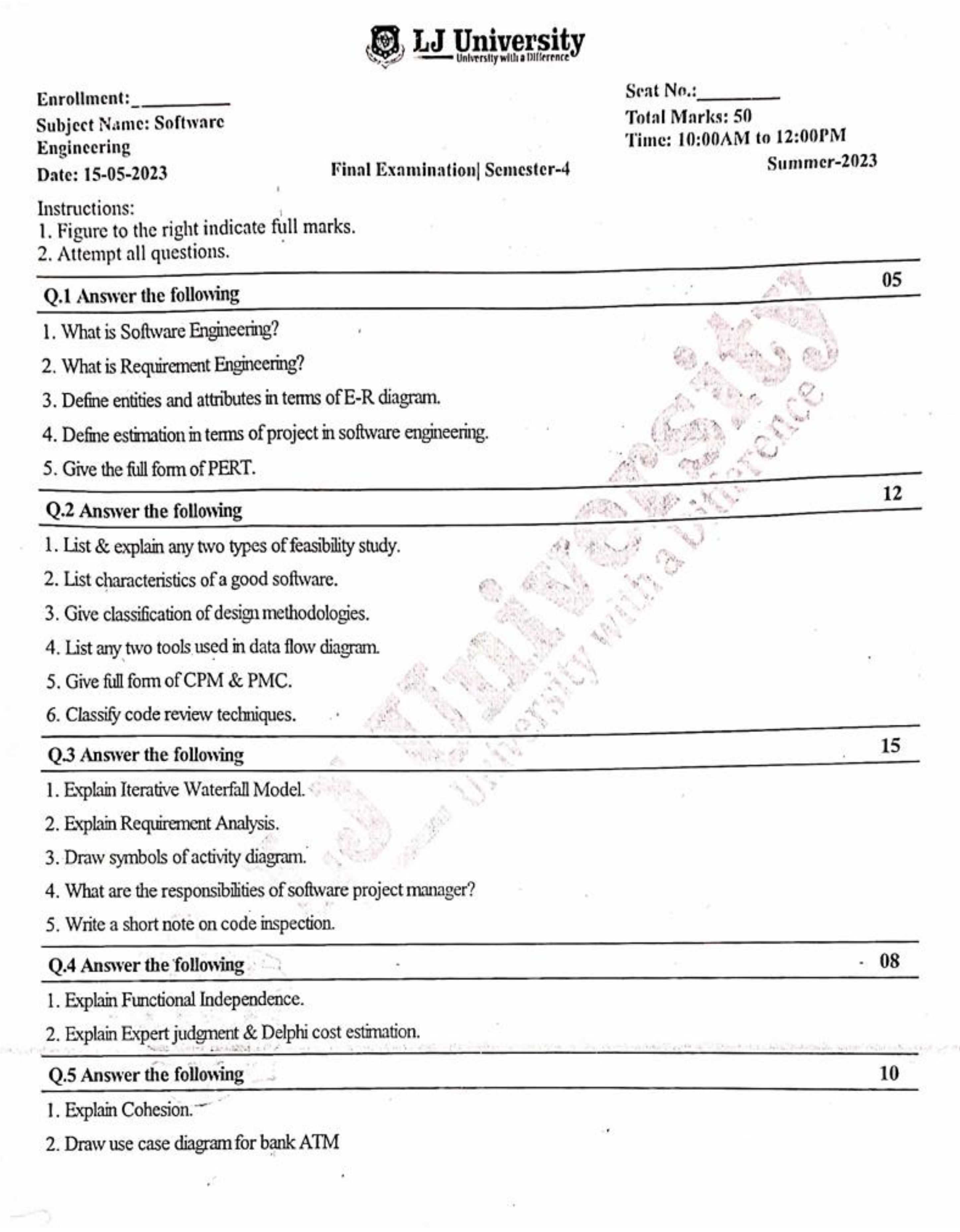
Software Engineering

50marks

previous exam paper solution.

Final exam paper



LJ University
University with a Difference

Enrollment:__________
Subject Name: Software Engineering
Date: 15-05-2023

Final Examination| Semester-4

Seat No.:_________
Total Marks: 50
Time: 10:00AM to 12:00PM
Summer-2023

Instructions:
1. Figure to the right indicate full marks.
2. Attempt all questions.

Q.1 Answer the following **05**

1. What is Software Engineering?
2. What is Requirement Engineering?
3. Define entities and attributes in terms of E-R diagram.
4. Define estimation in terms of project in software engineering.
5. Give the full form of PERT.

Q.2 Answer the following **12**

1. List & explain any two types of feasibility study.
2. List characteristics of a good software.
3. Give classification of design methodologies.
4. List any two tools used in data flow diagram.
5. Give full form of CPM & PMC.
6. Classify code review techniques.

Q.3 Answer the following **15**

1. Explain Iterative Waterfall Model.
2. Explain Requirement Analysis.
3. Draw symbols of activity diagram.
4. What are the responsibilities of software project manager?
5. Write a short note on code inspection.

Q.4 Answer the following **08**

1. Explain Functional Independence.
2. Explain Expert judgment & Delphi cost estimation.

Q.5 Answer the following **10**

1. Explain Cohesion.
2. Draw use case diagram for bank ATM

[1 marks solution]

Ans 1: Software engineering is the systematic application of engineering approaches to the development, operation, and maintenance of software systems.

Ans 2: Requirement engineering is like making a blueprint before building a house. It's the process of figuring out what the software needs to do and how it should work, before actually building it.

Ans 3: - An **entity** represents a real-world object or concept, such as a person, place, thing, or event.
- An **attribute** is a characteristic or property of an entity, describing some aspect of it.

Ans 4: Estimation in projects is like making an educated guess about how much time, money, and effort something will take to complete.

Ans 5: The full form of PERT in software engineering is "Program Evaluation and Review Technique."

[2 marks solution]

Ans 1: Two types of feasibility in project management are:

I) **Technical Feasibility :** This checks if we have the right tools and skills to do the project with the technology we currently have. For example, if we want to build an app, do we have the programmers who know how to do it, and do we have the right software?

II) **Economic Feasibility** : This is about whether the project makes financial sense. We look at things like how much money we'll have to spend to make the project happen and how much we think we'll make from it. If it costs more than it's worth or if we can't make enough money from it, then it might not be a good idea.

Ans 2: **Characteristics of a good:**

- SRS: Concise
- Complete
- Conceptual integrity
- Structured

- Black box view
- Verifiable
- Adaptable
- Portable
- Unambiguous
- Traceable

Ans 3: 1. What are the two main categories of design methodologies?

a. Traditional methodologies
b. Agile methodologies

2. What are the primary approaches within traditional design methodologies?

a. Waterfall model
b. Spiral model

Ans 4: 1. What are two tools commonly used in creating data flow diagrams (DFDs)?

a. Microsoft Visio
b. Lucidchart

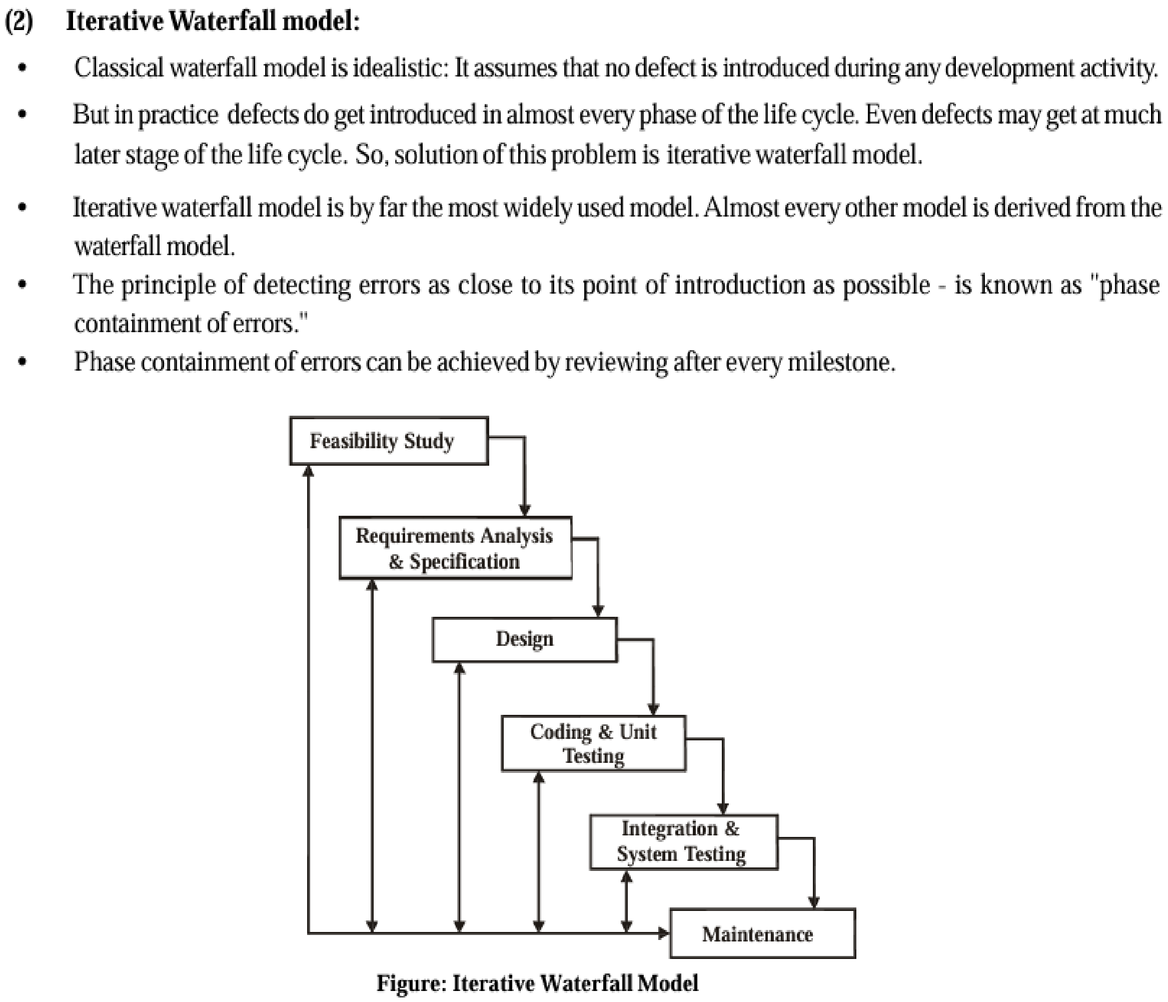
Ans 5: In software engineering:

CPM stands for Critical Path Method.
PMC stands for Project Management Committee.

Ans 6: Code review is a peer-review process used to examine code to identify problems and improve software quality. Code review is an important task in the development of software for embedded systems, especially those that require certification.

[3 marks solution]

Ans 1:



(2) Iterative Waterfall model:

- Classical waterfall model is idealistic: It assumes that no defect is introduced during any development activity.
- But in practice defects do get introduced in almost every phase of the life cycle. Even defects may get at much later stage of the life cycle. So, solution of this problem is iterative waterfall model.
- Iterative waterfall model is by far the most widely used model. Almost every other model is derived from the waterfall model.
- The principle of detecting errors as close to its point of introduction as possible - is known as "phase containment of errors."
- Phase containment of errors can be achieved by reviewing after every milestone.

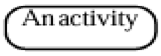
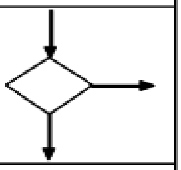
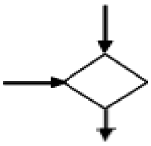
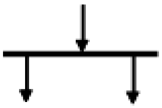
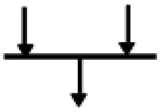
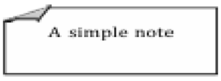
Feasibility Study
Requirements Analysis & Specification
Design
Coding & Unit Testing
Integration & System Testing
Maintenance

Figure: Iterative Waterfall Model

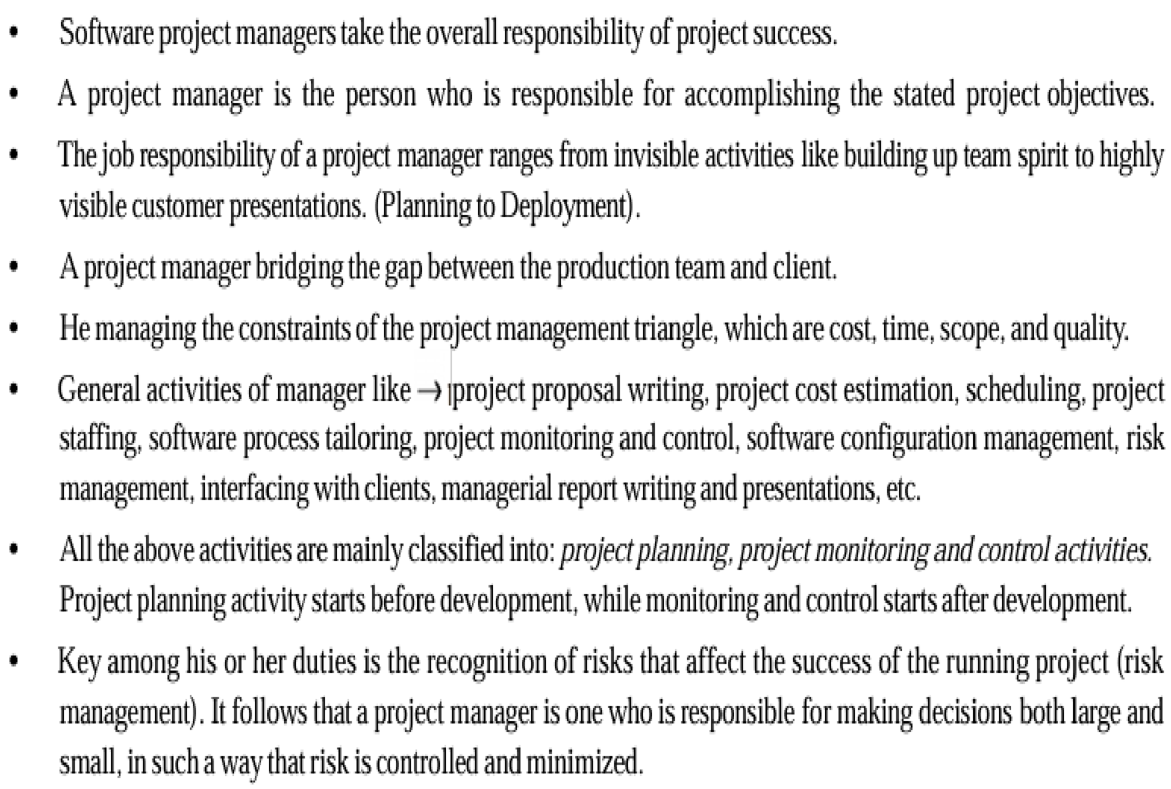
Ans 2: Requirement analysis is the process of gathering, documenting, and analyzing the needs and constraints of stakeholders to define the features and functionalities of a software system.

It involves understanding user expectations, business objectives, and technical specifications to ensure the software meets its intended purpose effectively.

Ans 3:

Elements or Components and its description	Symbol
Activity 1. **Initial activity (OR Start activity)** – This shows the starting point or first activity of the flow. – It is denoted by a solid circle. 2. **Final activity (OR End activity)** – The end of the activity diagram shown by a bull's eye symbol. It represents the end point of all activities.	 ● ◉
Flow or Transition – A flow (also termed as edge or transition) is represented with a directed arrow. – This is used to show transfer of control from one activity to another.	→
Decision (Branch) – A decision node represented with a diamond. – It is a branch where single transition (flow) enters and several outgoing transitions.	
Merge – This is represented with a diamond shape with two or more input transitions and a single output transition.	
Fork – Fork is a point where parallel activities begin. – Fork is denoted by black bar with one incoming transition and several outgoing transitions. – When the incoming transition is triggered, all the outgoing transitions are taken Into parallel.	
Join – Join is denoted by a black bar with multiple incoming transitions and single outgoing transition. – It represents the synchronization of all concurrent activities.	
Note – UML allows attaching a note to different components of diagram to present some textual information. – It could be some comments or may be some constraints. – A note generally attached to a decision point to indicate the branching criteria. – It is denoted by a rectangle with cut a side.	
Partition or Swimlanes – Different components of an activity diagram can be logically grouped into different areas, called partition or swimlanes. – They often correspond to different users or different units of organization. – It is denoted by drawing vertical parallel lines. – Partitions in an activity diagram are not mandatory.	
Guard conditions – Guard conditions control transition from alternative transitions based on condition. – These are represented by square brackets.	**[ ]**

Ans 4:



- Software project managers take the overall responsibility of project success.
- A project manager is the person who is responsible for accomplishing the stated project objectives.
- The job responsibility of a project manager ranges from invisible activities like building up team spirit to highly visible customer presentations. (Planning to Deployment).
- A project manager bridging the gap between the production team and client.
- He managing the constraints of the project management triangle, which are cost, time, scope, and quality.
- General activities of manager like → project proposal writing, project cost estimation, scheduling, project staffing, software process tailoring, project monitoring and control, software configuration management, risk management, interfacing with clients, managerial report writing and presentations, etc.
- All the above activities are mainly classified into: *project planning, project monitoring and control activities.* Project planning activity starts before development, while monitoring and control starts after development.
- Key among his or her duties is the recognition of risks that affect the success of the running project (risk management). It follows that a project manager is one who is responsible for making decisions both large and small, in such a way that risk is controlled and minimized.

Ans 5: In software engineering, "code inspiration" refers to the process of getting ideas and solutions from existing code.

It's like finding inspiration from a painting or a story.

When programmers encounter challenges or need to solve a problem, they often look at how others have tackled similar issues in their code.

This can help them learn new techniques, improve their own coding skills, and ultimately build better software.

Code inspiration can come from various sources such as open-source projects, online forums, or even code written by colleagues.

By studying and understanding different approaches, programmers can create more efficient and elegant solutions to their own coding problems.

[4 marks solution]

Ans 1: In software engineering, a functional dependency defines a relationship between attributes in a database table.
It describes how the value of one attribute uniquely determines the value of another within the same table.
For example, in a table of employee records, the employee ID uniquely determines the employee's name.
Functional dependencies are crucial in database design as they ensure data integrity and help in normalization, which reduces redundancy and improves efficiency.
Identifying and understanding functional dependencies allows developers to design robust database schemas that accurately represent the relationships between data elements, leading to more efficient and maintainable software systems.

Ans 2: Expert judgment and the Delphi cost estimation technique are both methods used in software engineering for estimating project costs and timelines.

Expert judgment relies on the insights and experience of individuals or a panel of experts in the field. These experts analyze the project requirements, scope, and other relevant factors to provide educated estimates based on their knowledge and past experiences. This method is subjective but valuable, especially when experts have deep domain knowledge and experience in similar projects.

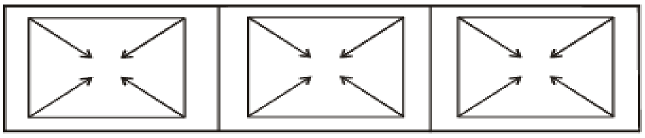
The Delphi method, on the other hand, is a structured approach that involves multiple rounds of anonymous feedback and iteration. Experts provide their estimates independently, and these estimates are then aggregated and shared anonymously. After each round, experts review the aggregated estimates and provide revised estimates based on the group feedback. This process continues until a consensus is reached.

Both methods have their advantages and can be effective depending on the context of the project and the availability of expertise. Expert judgment is more straightforward but can be influenced by biases, while the Delphi method provides a more structured approach to achieve consensus but requires more time and effort.

[5 marks solution]

Ans 1:

- **Cohesion:**
  - Cohesion is → a measure of functional strength of a module.
  - Cohesion keeps the internal modules together, and represents the functional strength.
  - Cohesion of a module represents how tightly bound the internal elements of a module are to one another.



Cohesion = strengths of relations within modules

Classification of cohesion:

Coincidental	Logical	Temporal	Procedural	Communicational	Sequential	Functional

Worst Best

(Low) → (High)

Coincidental cohesion

- It is the lowest cohesion. Coincidental cohesion occurs when there are no meaningful relationships between the elements.
- A module is said to have coincidental cohesion, if it performs a set of tasks that relate to each other very loosely.
- It is also called random or unplanned cohesion.

Logical cohesion

- A module is said to be logically cohesive if there is some logical relationships between the elements of module, and the elements perform functions that fall into same logical class.
- For example: the tasks of error handling, input and output of data.

Temporal cohesion

- Temporal cohesion is same as logical cohesion except that the elements are also related in time and they are executed together.
- A module is in temporal cohesion when a module contains functions that must be executed in the same time span.
- Example: modules that perform activities like initialization, cleanup, and start-up, shut down are usually having temporal cohesion.

Procedural cohesion

- A module has procedural cohesion when it contains elements that belong to common procedural unit.
- A module is said to have procedural cohesion, if the set of the modules are all part of a procedure (algorithm) in which certain sequence of steps are carried out to achieve an objective.

Communicational cohesion

- A module is said to have communicational cohesion, if all functions of the module refer to or update the same data structure, for example the set of functions defined on an array or a stack.
- These modules may perform more than one function together.

Sequential cohesion

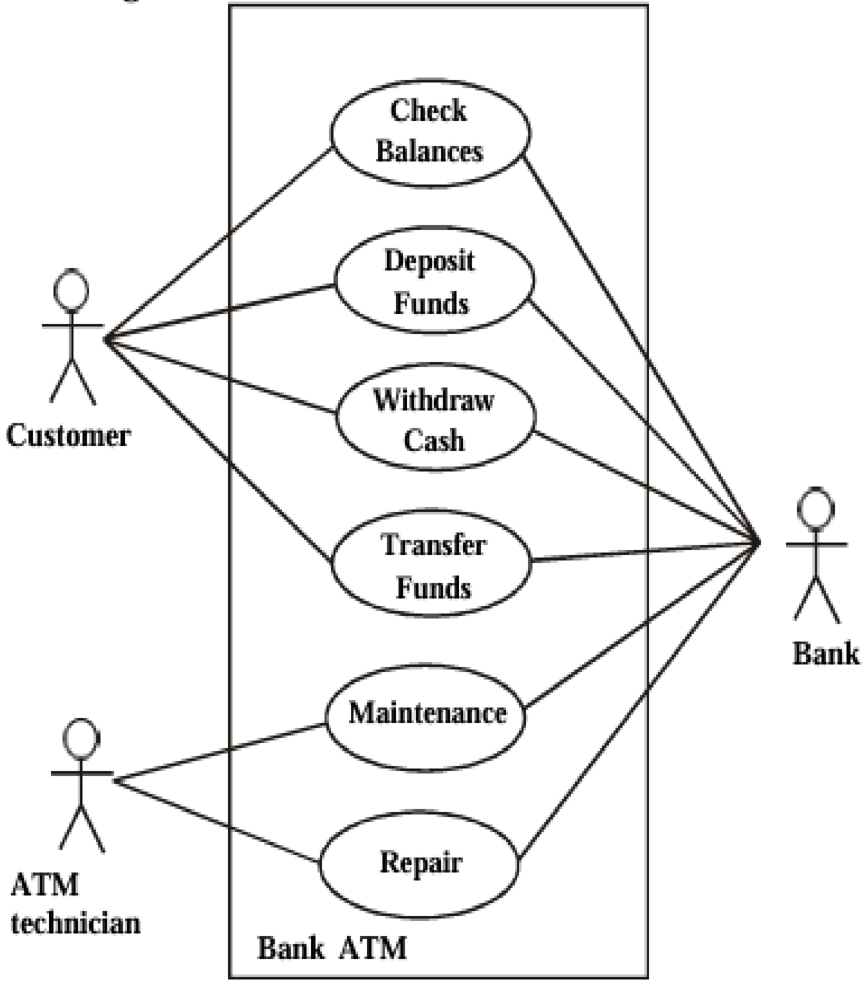
- When the output of one element in a module forms the input to another, we get sequential cohesion.
- Sequential cohesion does not provide any guideline how to combine these elements into modules.
- For example, in a TPS (transaction processing system), the get-input, validate-input, sort-input functions are grouped into one module.

Functional cohesion

- Functional cohesion is the strongest cohesion.
- In it, all the elements of the module are related to perform a single task.

Ans 2:

Example: Use case diagram for bank ATM


Check Balances
Deposit Funds
Withdraw Cash
Transfer Funds
Maintenance
Repair
Customer
ATM technician
Bank
Bank ATM