# Sardar Patel Institute of Technology

Bhavan's Campus, Munshi Nagar, Andheri (West), Mumbai-400058, India

(Autonomous College Affiliated to University of Mumbai)

**Mid Semester Examination**

September 2018

**Max. Marks: 20** **Duration: 1Hrs**

**Class:T.E.** **Semester:V**

**Course Code:IT51** **Branch:IT**

**Name of the Course: Software Engineering**

**Instruction:**

**(1) All questions are compulsary**

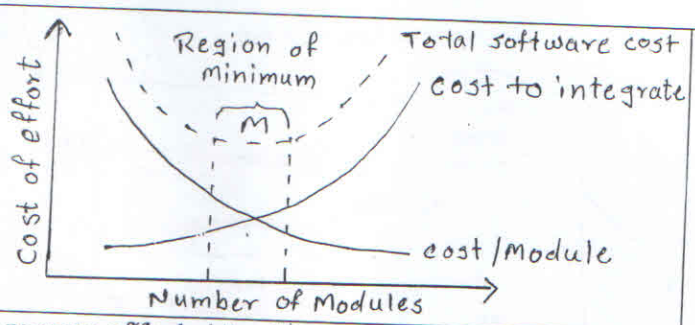
**(2) Draw neat diagrams**

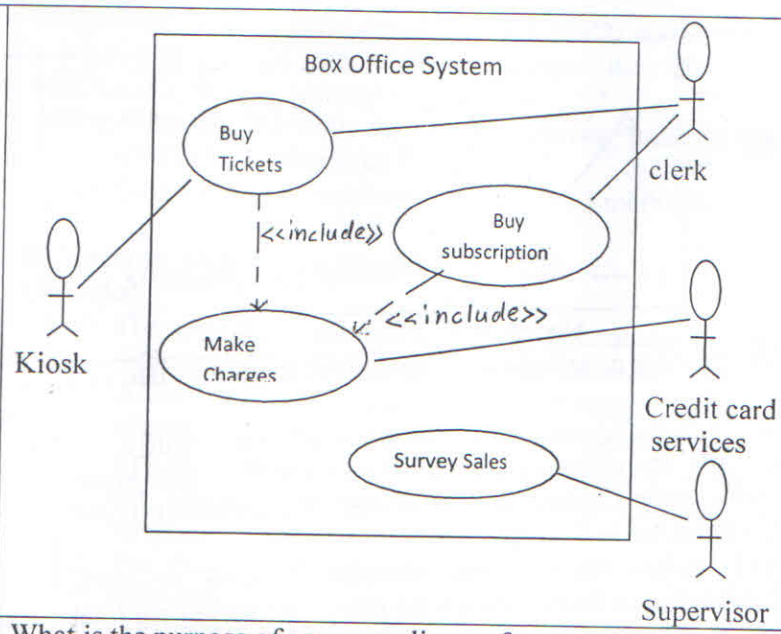
**(3) Assume suitable data if necessary**

## Question paper Synoptic

| QNo | Question | Marks | CO |
|---|---|---|---|
| 1 | What is the main idea behind an incremental development process? Illustrate your answer using following example,<br>A system with 15 functional requirements (leading to 15 user services) where requirements Fr1,..Fr5 have the lowest priority (i.e. they are the least important), requirements Fr6,..Fr10 are more important, and requirements Fr11,..Fr15 are the most important to users.<br>Give two advantages incremental development. | 4<br><br>1 | CO 1 |
| Ans | Discussion on incremental development process –**(4 Marks)**<br>Answer should contain following point:<br>1) Prioritization of requirement –**(2marks)**<br>**2)** Development in sequential or parallel-**(2marks)**<br>For example.<br>Functional requirements are prioritized (divided into different groups with different priorities). A number of delivery increments are then defined (corresponding to different groups). The increments are developed either sequentially or in a phased parallel manner.<br>For example – requirements Fr11-Fr15 are included in the first increment, requirements Fr6-Fr10 in the second and requirements Fr1-Fr5 in the third increment<br>Advantage-**(1mark)**<br>1) It provides on the rigid nature of sequential approach.<br>2) This method is of great help when organization is low on staffing.<br>OR | | |

| | | | |
|---|---|---|---|
| 1 | Component based systems development (CBSD) methods place a lot of emphasis on component reuse, hence they differ from 'traditional' systems development methods.<br>1) What are the main differences between 'traditional' and CBSD life cycle model?<br>2) Give the purpose of each stage of CBSD methods. | 2<br>3 | |
| Ans. | Differences between 'traditional' and CBSD life cycle model: **(2marks)**<br>1) The process of component-based system development differs from 'traditional' development processes. The main difference is in the separation of the development process of components from the development process of systems.<br>2) For the system-level process, the emphasis is on finding the proper components and verifying/evaluating and integrating them.<br>3) For the component-level process, design for reuse is the main concern.<br>4) Component assessment is a new (possibly separate) process for finding and evaluating components.<br>Stages of CBSD methods: **(3 Marks)**<br>1) Identification of Components<br>2) Look up components in library<br>3) Extract components if available<br>4) Build components if unavailable<br>5) Put new components in library<br>6) Construct nth iteration of system | | |
| 2 | What is the relationship between the concepts of information hiding as an attribute of effective modularity?<br>With the help of diagram discuss when modularity is considered. | 2<br>3 | CO1 |
| Ans | Relationship between the concepts of information hiding as an attribute of effective modularity. **(2Marks)**<br>1) Information hiding suggests that modules be 'characterized by design decisions that hides from all others'<br>2) Hiding implies effective modularity can be achieved by defining a set of independent modules that communicate with one another only that information necessary to achieve software function.<br>3) Hiding defines and enforces access constraints to both procedural details within a module and any local data structure used by the module.<br>4) It helps when modifications are required during testing and later during software maintenance.<br>Diagram: **(3Marks)** | | |

| | | | |
|---|---|---|---|
| |  | | |
| 3 | Give the purpose of "include" and "extends" relationship in use case diagram.<br>Draw a use case diagram for following box office case study.<br>Consider the box office include the clerk, supervisor and kiosk. The Kiosk accepts orders from customer. The customer is not directly connected to the application.<br>It should include buying tickets through the kiosk or clerk, buying subscription (only through the clerk) and surveying total sales (at the request of the supervisor).Buying tickets and buying subscriptions include a common fragment of making charges to the credit card services. | 1<br><br>4 | CO 2 |
| Ans | Purpose of "include" and "extends" relationship in use case diagram. **(1Mark)**<br>Include -<br>Include relationship implies one use case include the behaviors of another use case in its sequence of events and actions.<br>Extends-<br>Extends relationship allow users show optional system behavior. It is normally used to capture alternate paths or scenarios.<br><br>Use case diagram **(4Marks)**<br>Diagram should contain:<br>1) Proper identification of Use cases.<br>2) Proper identification of Actor.<br>3) Proper identification of association between Actor and Use cases. | | |




| | | | |
|---|---|---|---|
| 4 | What is the purpose of sequence diagram?<br>Draw a sequence diagram for the buy tickets use case from Q3. | 1<br>3 | CO2 |
| Ans | Purpose of sequence diagram **(1 Mark)**<br>Show the behavior sequence of a use case arranged in a time sequence.<br><br>A sequence diagram **(3Marks)** | | |