

# ANNA UNIVERSITY, CHENNAI

AFFILIATED INSTITUTIONS

**R - 2009**

CURRICULUM I SEMESTER (FULL TIME)

M.E. SOFTWARE ENGINEERING

SEMESTER I

SL. NO	COURSE CODE	COURSE TITLE	L	T	P	C
<b>THEORY</b>						
1	MA9219	<a href="#">Operations Research</a>	3	1	0	4
2	CS9213	<a href="#">Computer Networks and Management</a>	3	0	0	3
3	SE9213	<a href="#">Object Oriented Software Engineering</a>	3	0	0	3
4	SE9214	<a href="#">Software Architecture</a>	3	0	0	3
5	SE9215	<a href="#">Formal Methods in Software Engineering</a>	3	0	0	3
<b>PRACTICAL</b>						
6	CS9216	<a href="#">Networking Lab</a>	0	0	3	2
7	SE9217	<a href="#">Case Tools Lab</a>	0	0	3	2
<b>TOTAL</b>			<b>15</b>	<b>1</b>	<b>6</b>	<b>20</b>

<b>UNIT I</b>	<b>QUEUEING MODELS</b>	<b>12</b>
Poisson Process – Markovian Queues – Single and Multi-server Models – Little’s formula – Machine Interference Model – Steady State analysis – Self Service Queue.		
<b>UNIT II</b>	<b>ADVANCED QUEUEING MODELS</b>	<b>12</b>
Non- Markovian Queues – Pollaczek Khintchine Formula – Queues in Series – Open Queueing Networks – Closed Queueing networks.		
<b>UNIT III</b>	<b>SIMULATION</b>	<b>12</b>
Discrete Even Simulation – Monte – Carlo Simulation – Stochastic Simulation – Applications to Queueing systems.		
<b>UNIT IV</b>	<b>LINEAR PROGRAMMING</b>	<b>12</b>
Formulation – Graphical solution – Simplex method – Two phase method – Transportation and Assignment Problems.		
<b>UNIT V</b>	<b>NON-LINEAR PROGRAMMING</b>	<b>12</b>
Lagrange multipliers – Equality constraints – Inequality constraints – Kuhn – Tucker conditions – Quadratic Programming.		

**TOTAL: 60 PERIODS**

**TEXT BOOKS:**

1. Winston.W.L. “Operations Research”, Fourth Edition, Thomson – Brooks/Cole, 2003.
2. Taha, H.A. “Operations Research: An Introduction”, Ninth Edition, Pearson Education Edition, Asia, New Delhi, 2002.

**REFERENCES:**

1. Robertazzi. T.G. “Computer Networks and Systems – Queuing Theory and Performance Evaluation”, Third Edition, Springer, 2002 Reprint.
2. Ross. S.M., “Probability Models for Computer Science”, Academic Press, 2002.

<b>UNIT I</b>	<b>HIGH SPEED NETWORKS</b>	<b>9</b>
Frame Relay Networks – Asynchronous transfer mode – ATM Protocol Architecture, ATM logical Connection, ATM Cell – ATM Service Categories – AAL. High Speed LAN’s: Fast Ethernet, Gigabit Ethernet, Fibre Channel – Wireless LAN’s.		

**UNIT II CONGESTION AND TRAFFIC MANAGEMENT 9**

Queuing Analysis- Queuing Models – Single Server Queues – Effects of Congestion – Congestion Control – Traffic Management – Congestion Control in Packet Switching Networks – Frame Relay Congestion Control.

**UNIT III TCP AND ATM CONGESTION CONTROL 10**

TCP Flow control – TCP Congestion Control – Retransmission – Timer Management – Exponential RTO backoff – KARN's Algorithm – Window management – Performance of TCP over ATM. Traffic and Congestion control in ATM – Requirements – Attributes – Traffic Management Frame work, Traffic Control – ABR traffic Management – ABR rate control, RM cell formats, ABR Capacity allocations – GFR traffic management.

**UNIT IV INTEGRATED AND DIFFERENTIATED SERVICES 9**

Integrated Services Architecture – Approach, Components, Services- Queuing Discipline, FQ, PS, BRFQ, GPS, WFQ – Random Early Detection, Differentiated Services.

**UNIT V PROTOCOLS FOR QoS SUPPORT 8**

RSVP – Goals & Characteristics, Data Flow, RSVP operations, Protocol Mechanisms – Multiprotocol Label Switching – Operations, Label Stacking, Protocol details – RTP – Protocol Architecture, Data Transfer Protocol, RTCP.

**TOTAL: 45 PERIODS**

**TEXT BOOKS:**

1. William Stallings, "HIGH SPEED NETWORKS AND INTERNET", Pearson Education, Second Edition, 2002.

**REFERENCES:**

1. Warland & Pravin Varaiya, "HIGH PERFORMANCE COMMUNICATION NETWORKS", Jean Harcourt Asia Pvt. Ltd., II Edition, 2001.
2. Irvan Pepelnjk, Jim Guichard and Jeff Apar, "MPLS and VPN architecture", Cisco Press, Volume 1 and 2, 2003.

**SE9213 OBJECT ORIENTED SOFTWARE ENGINEERING LT P C  
3 0 0 3**

**UNIT I INTRODUCTION 9**

System Concepts – Software Engineering Concepts – Development Activities – Managing Software Development – Unified Modeling Language – Project Organization – Communication

**UNIT II ANALYSIS 9**

Requirements Elicitation – Concepts – Activities – Management – Analysis Object Model – Analysis Dynamic Models

**UNIT III SYSTEM DESIGN 9**

Decomposing the system – Overview of System Design – System Design Concepts – System Design Activities – Addressing Design Goals – Managing System Design

**UNIT IV OBJECT DESIGN AND IMPLEMENTATION ISSUES 9**  
Reusing Pattern Solutions – Specifying Interfaces – Mapping Models to Code – Testing

**UNIT V MANAGING CHANGE 9**  
Rationale Management – Configuration Management – Project Management – Software Life Cycle

**TOTAL:45 PERIODS**

**REFERENCES:**

1. Bernd Bruegge, Alan H Dutoit, Object-Oriented Software Engineering, 2<sup>nd</sup> ed, Pearson Education, 2004.
2. Craig Larman, Applying UML and Patterns, 3<sup>rd</sup> ed, Pearson Education, 2005.
3. Stephen Schach, Software Engineering 7<sup>th</sup> ed, McGraw-Hill, 2007.

**SE9214 SOFTWARE ARCHITECTURE LT P C  
3 0 0 3**

**UNIT I 8**  
Introduction – software design levels – software engineering discipline – architecture business cycle – architectural patterns – reference models – architectural structures, views

**UNIT II 9**  
Architectural styles – pipes and filters – object-orientation – invocation – layered systems – repositories – interpreters – process control – heterogenous architectures – case studies

**UNIT III 10**  
Architecture and functionality – architecture qualities – architecture in the lifecycle - Architectural design - Shared information systems – database integration – integration in software development environments – architectural structures for shared information systems

**UNIT IV 9**  
Architectural design guidance – design space – design rules – applying design space – quantified design space – formal models and specification – formalizing architectural style, design space - z - notation

**UNIT V 9**  
Linguistic issues – requirements for architectural description languages – first class connectors – adding implicit invocation to traditional programming languages – tools for architectural design – universal connector language - Software architecture Documentation – reconstruction

**TOTAL: 45 PERIODS**



**REFERENCE:**

1. Jim Woodcock, Jim Davies, "Using Z Specification, Refinement and Proof", Prentice Hall International, 1996.

**CS9216****NETWORKING LAB****LT P C  
0 0 3 2**

1. Socket Programming
  - a. TCP Sockets
  - b. UDP Sockets
  - c. Applications using Sockets
2. Simulation of Sliding Window Protocol
3. Simulation of Routing Protocols
4. Development of applications such as DNS/ HTTP/ E – mail/ Multi - user Chat
5. Simulation of Network Management Protocols
6. Study of Network Simulator Packages – such as opnet, ns2, etc.

**TOTAL: 45 PERIODS****SE9217****CASE TOOLS LAB****LT P C  
0 0 3 2**

1. Practicing the different types of case tools such as (Rational Rose & other Open Source) used for all the phases of Software development life cycle.
2. Data modeling
3. Semantic data modeling
4. Source code generators
5. Re-engineering

6. Experimenting CASE Environments

- a. Toolkits
- b. Language-centered
- c. Integrated
- d. Fourth generation
- e. Process-centered

7. Implementation of the following using CASE Workbenches:

- a. Business planning and modeling
- b. Analysis and design
- c. User-interface development
- d. Programming
- e. Verification and validation
- f. Maintenance and reverse engineering
- g. Configuration management
- h. Project management

**TOTAL: 45 PERIODS**