

## GENERAL GUIDELINES

- This book is to be brought to the class daily.
- Students are not permitted to attend the class without the identity card.
- Students should be well in time for the first class and subsequent classes thereafter.
- Students should keep the Classrooms, Laboratories and Workplace clean and tidy.
- **Writing on desks, walls is strictly prohibited, failing which the students will be fined heavily. If the identity of the individual is not established the entire class / students in the block will be fined.**
- Students are advised to show due respect to all faculty regardless of their department and maintain affable personality.
- Students are to maintain absolute discipline and decorum, so as to promote the fair name of the college in all its activities.
- **Students securing less than 85% attendance in any individual subject in the semester will not be allowed to take up the exams. Attendance and test performance will be announced in the concerned department notice board as per the calendar of events and the students are advised to bring any discrepancy to the notice of the HOD.**
- Students are informed to clarify their doubts in the respective subjects with the faculty by taking prior appointment.
- Students are to inform their parents that they should follow up the progress of their wards by being in touch with the college authorities at regular intervals.
- **Ragging is punishable under Karnataka Education Act and is strictly prohibited. Any student involved in ragging, will be severely punished.**
- **Students should come prepared with algorithm / flowchart / program for all the experiments before attending the laboratory session.**
- Students should bring the data sheets and laboratory records completed in all respect to the laboratory.
- Take the PRINTOUTS of the SOURCE LISTING and OUTPUT of the program after EXECUTION and DELETE your files.
- Students are not supposed to alter the configuration of the system / any software on the system.
- Students who secure less than 60% in the University exam and / or who secure less than 60% in the internal test are to go through the Academic Support programme compulsorily. The guidelines will be issued separately by the concerned department.
- Students are to present the mentor meeting failing which appropriate disciplinary action will be taken.

### SEMESTER - VI - TIME TABLE

Day / Time	8.15 - 9.15	9.15 - 10.15	10.45 -11.45	11.45-12.45	1.30-2.30	2.30-3.30	3.45-4.45	4.45-5.45
<b>Monday</b>								
<b>Tuesday</b>								
<b>Wednesday</b>								
<b>Thursday</b>								
<b>Friday</b>								
<b>Saturday</b>								

### SCHEME OF STUDY AND EXAMINATIONS FOR BE DEGREE COURSE IN COMPUTER SCIENCE & ENGINEERING

Sl. No	Code No.	Subject	No. of Hr. / week		Duration of Exams		Class Sessional Marks	Exam Marks	Total Marks
			Theory	Practical	Theory	Practical			
1.	CS 61	Unix System Programming	04	--	03	--	25	100	125
2.	CS 62	Software Engineering	04	--	03	--	25	100	125
3.	CS 63	Computer Graphics	04	--	03	--	25	100	125
4.	CS 64	Computer Networks – I	04	--	03	--	25	100	125
5.	CS 65	System Simulation and Modeling	04	--	03	--	25	100	125
6.		ELECTIVE-I (Group A)	04	--	03	--	25	100	125
7.	CSL67	Visual Programming and Data Base Applications Lab	--	03	--	03	25	50	75
8.	CSL68	Graphics Lab	--	03	--	03	25	50	75
<b>Total</b>			24	06			200	700	900

### STUDENT SELF APPRAISAL

SUBJECT	MARKS			ATTENDANCE		
	TEST 1	TEST 2	TEST 3	REPORT 1	REPORT 2	REPORT 3
Unix System Programming						
Software Engineering						
Computer Graphics						
Computer Networks – I						
System Simulation and Modeling						
ELECTIVE-I (Group A)						
Visual Programming and Data Base Applications Lab						
Graphics Lab						

# CS61: UNIX SYSTEM PROGRAMMING

Faculty:

No of hours: 52

Class #	Chapter Title/ Reference Literature	Topics covered	% of portion covered	
			Reference chapter	Cumulative
1	<b>Chapter 1</b> <b>UNIX and ANSI Standards</b> <b>UNIX and POSIX APIs</b> T1: Chapters 1,5 T2: Chapters 1,2	The ANSI C standard The ANSI/ISO C++ standards	12%	12%
2		Differences between ANSI C and C++		
3		The POSIX standards, The POSIX.1 FIPS standard		
4		The X/OPEN standards		
5		The POSIX APIs		
6		The UNIX and POSIX development environment API common characteristics		
7	<b>Chapter 2</b> <b>UNIX Files</b> <b>UNIX File APIs</b> T1: Chapters 6,7 T2: Chapters 3,4,5,6	File Types UNIX and POSIX file systems, file attributes	23%	35%
8		Inodes in UNIX System V Application program interface to files UNIX kernel support for files		
9		Relationship of C stream pointers and file descriptors Directory files, hard and symbolic links		
10		General file APIs		
11		General file APIs contd.,		
12		File and record locking		
13		Directory file APIs		
14		Device file APIs		
15		FIFO file APIs, symbolic link file APIs, General file class		
16		regfile class for regular files, dirfile class for directory files		
17		FIFO and Device File Class		
18		Symbolic link file class, file listing program		
19	<b>Chapter 3</b> <b>UNIX Processes</b> T1: Chapter 8 T2: Chapters 7,8,9	Introduction, main function, process termination Command line arguments, environment list	30%	65%
20		Memory layout of a C program, shared libraries Memory allocation, Environment variables		
21		setjmp and longjmp functions Getrlimit, setrlimit functions UNIX kernel support for processes		
22		Process identifiers, fork, vfork functions		
23		Exit, wait, waitpid, wait3, wait4 functions		
24		Race conditions, exec functions		
25		Changing user Ids and group Ids		
26		Interpreter files, system function		
27		Process accounting, user identification,		
28		Process time, I/O redirection		
29		Terminal logins, network logins		
30		Process groups, sessions		
31		Controlling terminal, tcgetgrp and tcsetgrp functions		
32		Job control		
33		Shell execution of programs		
34		Orphaned process groups		
35	<b>Chapter 4</b> <b>Signals and Daemon Processes</b> T1: Chapter 9 T2: Chapters 10, 13	UNIX kernel support for signals	15%	80%
36		Signal, signal mask, sigaction		
37		SIGCHLD signal and waitpid function		
38		Sigsetjmp and siglongjmp functions		
39		Kill, alarm		
40	Interval timers, POSIX.1b timers			
41	Daemon characteristics, Coding rules			
42	Error logging, Client-server model			
43	<b>Chapter 5</b> <b>Interprocess Communication</b> T1: Chapter 10 T2: Chapters 14,15	Overview of IPC methods	20%	100%
44		Pipes, popen, pclose functions		
45		Coprocesses, FIFOs		
46		System V IPC		
47		Message queues		
48		Semaphores		
49		Shared memory		
50		Client Server properties		
51		Stream pipes, Passing file descriptors		
52	Open Server version 1, Client server connection functions			

## Literature

Book Type	Code	Title & Author	Publication Info		
			Edition	Publisher	Year
Text Book	T1	<b>UNIX System Programming using C++</b> by Terrence Chan	First	PHI	1999
Text Book	T2	<b>Advanced Programming in the UNIX Environment</b> W Richard Stevens	Second	AW	1993
Reference Book	R1	<b>The Design of the UNIX operating System</b> by Maurice.J.Bach		AW	1998
Reference Book	R2	<b>UNIX Internals</b> by Uresh Vahalia		PHI	1984

## QUESTION BANK

### INTRODUCTION

**OBJECTIVE:** This chapter gives an over view of the UNIX, ANSI and POSIX standards, the UNIX and POSIX file system and the APIs.

1. What does POSIX stand for? 2
2. How do you ensure that a user program conforms to POSIX standards? 2
3. What are the features of test macros? 2
4. What do you mean by file attributes? 2
5. Define inode. 2
6. What is a device file? 2
7. List the differences between user Id and effective Id. 2
8. What do you mean by file descriptors? 2
9. List a few general APIs. 2
10. How do you lock a file? 2
11. How are directory files created? 2
12. Define socket. 2
13. What is a system call? Give example. 3
14. List the various file attributes. 3
15. How do you obtain the attributes of a particular file? 3
16. Distinguish between record locking and file locking. 3
17. Differentiate between advisory locks and mandatory locks. 3
18. How are device files created? 3
19. What is the purpose of X/OPEN standards? 4
20. Differentiate between user mode and system mode of operation. 4
21. What happens when a system call fails? 4
22. What do you mean by a file system? List a few file systems. 4
23. Explain how do you mount a file system? 4
24. Discuss the file structure. 4
25. List the features of FIPS Conforming systems. 5\*
26. Explain the different functions provided by the UNIX system to perform common set of APIs. 5\*
27. Discuss the differences between ANSI C and C. 5
28. Discuss the relationship between C stream pointers and file descriptors. 5
29. Differentiate between hard link and symbolic link with an example. 5\*
30. Write a note on FIFO file APIs. 6
31. List the salient features of ANSI C with examples. 8
32. Explain few errors that might occur when an API execution fails. 8
33. Write a note on POSIX limits. 8
34. What is the purpose of sysconf, pathconf, fpathconf functions? 8
35. What is a link? Explain the types of links. How are links created? 8
36. Explain the operations on directory files with related functions. 8
37. Explain the device file class? 8
38. What do you mean by environment variable? Explain the environment list functions. 8
39. Write a note on the various file types in UNIX or POSIX systems. 10\*
40. Write a program that prints the POSIX defined configuration options supported on any given system using feature test macros. 10\*
41. Explain the following APIs. 15\*
  - i. Open
  - ii. fcntl
  - iii. Link
  - iv. Stat
  - v. write
  - vi. Read
  - vii. close

### UNIX PROCESSES

**OBJECTIVE:** This chapter deals with the processes, relationships between processes, process control and related system calls.

42. What are the different ways by which a process can terminate? 2
43. What is an environment list? 2
44. What is race condition? 2
45. Define polling. 2
46. Define session. 2
47. List the advantages of shared library concept. 3
48. Differentiate between interpreter file and interpreter? 3

49.	What is the purpose of "system" function?	3
50.	Write a note on UNIX process.	4
51.	Discuss the memory layout of a C program.	5
52.	Write notes on exec functions.	5
53.	Explain the structure for accounting records.	8
54.	Explain terminal logins.	8
55.	Write notes on network logins.	8
56.	Describe the feature of job control.	8
57.	Write a program in C or C++ to emulate the UNIX ls -l command.	10*
58.	Write a program in C to obtain process attributes.	10
59.	Write notes on	10
	i) exit      ii) wait      iii)waitpid      iv)wait3      v) wait4	
60.	Write notes on	10
	i) setjmp      ii) longjmp      iii) setrlimit      iv) Getrlimit	

## **SIGNALS, DAEMON PROCESSES**

**OBJECTIVE:** This chapter explains the concept of signals, timers and the daemon processes.

61.	What do you mean by unreliable signals?	2
62.	Define signal mask.	2
63.	What is a signal set?	2
64.	What is a daemon?	2
65.	What are the advantages of POSIX.1b timers over UNIX timers?	3
66.	Write notes on slow system calls.	4
67.	Write notes on reentrant functions.	4
68.	List the differences between alarm and setitimer APIs.	4
69.	How are signals handled by the processes?	5
70.	Explain the signal function.	5
71.	How are signals generated?	5
72.	Describe the client server model.	5*
73.	What is a signal? List the various signals and their purpose.	8*
74.	What are the basic rules to coding a daemon?	8
75.	Write a program in C to setup a real time clock internal timer using alarm API?	10*
76.	Explain	10
	i) kill      ii) raise      iii) alarm      iv) pause      functions	
77.	Explain	10
	i) sigaction      ii) sleep      iii) abort      functions	
78.	Discuss the POSIX.1b APIs timer manipulation.	10
79.	Write notes on SVR4 streams log driver and bsd syslog facility.	10

## **INTERPROCESS COMMUNICATION**

**OBJECTIVE:** This chapter explains how interprocess communication takes place through semaphores, shared memory, message queues and the POSIX.1b IPC.

80.	Give the different forms of system V IPC.	2
81.	Define semaphores.	2
82.	Define stream pipe.	2
83.	Differentiate UNIX filter and coprocess.	3
84.	In IPC, How do processes interact with each other?	3
85.	List the drawbacks of system V IPC methods.	3
86.	Give the characteristics of POSIX.1b IPC methods.	3
87.	Give the different types of IPC.	4
88.	Write short notes on memory mapped I/O.	4*
89.	Distinguish between pipes and named pipes.	5*
90.	How are FIFOs created?	5
91.	Explain the IPC permission structure.	6
92.	Explain the UNIX kernel support for messages	8*
93.	Explain the shared memory concept.	8*
94.	Explain the functions used to send and receive file descriptor.	8
95.	Explain in detail the open server 1.	8
96.	Explain in detail the open server 2.	8
97.	Explain in detail the POSIX.1b message APIs	8
98.	Write notes on POSIX.1b semaphores.	8

99. Explain IPC through pipes 10  
100. Explain IPC using semaphores 10\*  
101. Describe in detail the properties of client and servers that are affected by different types of IPC 10

Marks	No of Questions
2	24
3	12
4	12
5	18
6	04
8	21
10	15
12	01
15	01
<b>Total</b>	<b>101</b>

### NOTES

# CS62: SOFTWARE ENGINEERING

**Faculty:**

**No of Hours: 52**

Class #	Chapter Title/ Reference Literature	Topics to be covered	% of portions covered	
			Reference chapter	Cumulative
1.	<b>Chapter #:1</b> Overview T1: Page#3-20, 42-70 R1: Page #22-49	<b>Software Engineering</b> –An Introduction	12%	12%
2.		Professional and Ethical responsibility		
3.		<b>Software Processes-</b> Software Process Models		
4.		Process iteration, Software specification		
5.		Software design and implementation		
6.		Software validation and evolution, Automated process support		
7.	Chapter #:2 Requirement Engineering T1 : Page #97-120,121-146,148-169,171-190 R2 : Page #33-69	<b>Software Requirements</b> -functional, non-functional	23%	35%
8.		User and system requirements		
9.		SRS document		
10.		<b>Requirement Engineering Process</b> -Feasibility studies		
11.		Requirement elicitation and analysis		
12.		Requirement validation and management		
13.		<b>System Models</b> -context models		
14.		Behavioral models		
15.		Data models, object models, CASE workbenches		
16.		<b>Software Prototyping</b> - Prototyping in software process		
17.	Rapid prototyping techniques	20%	55%	
18.	User Interface prototyping			
19.	<b>Architectural Design</b> -system structuring			
20.	Control models			
21.	Modular decomposition			
22.	Design specific architectures			
23.	<b>Object oriented design</b> -Object and classes			
24.	Object oriented design process, Design evolution			
25.	<b>User Interface Design</b> -principles			
26.	User interaction			
27.	Information presentation	13%	68%	
28.	User Support, Interface evaluation			
29.	Verification and validation planning			
30.	Software inspections			
31.	Automated static analysis			
32.	Clean room software development			
33.	<b>Software testing</b> -defect testing			
34.	Integration testing			
35.	Object oriented testing, Testing workbenches			
36.	Chapter #:5 Critical systems T1: Page#353-370,371-390			<b>Critical systems</b> - availability and reliability
37.		Safety and security		
38.		<b>Critical system specification</b> -software reliability specification		
39.		Safety specification		
40.	Chapter #:6 Software Management T1: Page#71-92,511-534 R1: Page #111-124,132-150,185-194	<b>Project Management</b> - activities	8%	93%
41.		Project planning		
42.		Project scheduling, Risk management		
43.		<b>Software cost Estimation</b> -productivity		
44.		Estimation techniques, Algorithmic cost modeling		
45.		Project duration and staffing		
46.		<b>Quality management</b> -Quality assurance and standards		
47.		Quality planning and control		
48.		Software measurements and metrics		
49.	Chapter #:7 Software Evolution T1: Page#581-599,622-639	<b>Legacy systems</b> - Legacy systems structures	7%	100%
50.		Legacy system design and assessment		
51.		<b>Software Re-engineering</b> - Source code translation, Reverse engineering		
52.		Program structure improvement, modularization, Data reengineering		

## Literature

Book type	Code	Title and Author	Publication specification		
			Edition	Publication	Year
Text Book	T1	Software Engineering by Ian Sommerville	6 <sup>th</sup>	Pearson Education	2001
Reference Book	R1	Software Engineering: A Practitioner's Approach by Roger Pressman	4 <sup>th</sup>	TMH	
Reference Book	R2	An Integrated Approach to Software Engineering by Pankaj Jalote		Narosa Publications	
Reference Book	R3	Object oriented and classical software engineering by Stefan R Schach		TMH	2002

## QUESTION BANK

### OVERVIEW

**OBJECTIVE:** The objective of this topic is to understand the

- The Software engineering Definition
  - The Software Engineering Process and Product
  - Various Process Models- Case study and comparisons
1. Define software product and software engineering (IEEE Def). 05
  2. Mention properties of software. 05
  3. "Software is not merely a collection of programs- Is programming systems product". Elucidate. 05
  4. What are the main phases in software development? 05
  5. Explain the requirements analysis phase of software development. 08
  6. Explain the design phase of software development. 05
  7. Explain the coding phase of software development 05
  8. Explain the testing phase of software development.
  9. Explain the software development model definition. 05
  10. Compare the software development models. 08
  11. Explain waterfall model. 08
  12. Explain prototyping model. 08
  13. Explain iterative enhancement model. 08
  14. Explain the spiral model. 08
  15. Which process model can be used for  
A simple data processing system. 05  
An outline reservation system.
  16. Mention the drawbacks of each of the software development models. 05
  17. Explain the role of metrics. 05
  18. Comment on the error distribution in the different phases. 05
  19. Comment on the distribution of effort in the life span of software development. 05
  20. What are major phases in the entire life cycle of the software? Which of them consume maximum effort? 05
  21. Is it reasonable to assume that if software is easy to test, it will be easy to maintain. Suppose that putting in extra effort in design and coding you increase the cost of these phases by 15% but you reduce the cost of maintenance by 5%. Will you decide to put in the extra effort and why? 08
  22. Describe the major phases in software development. Discuss the error distribution and cost of correcting the errors during development. 08
  23. Explain the spiral model. Discuss the features of a software project for which the spiral model could be a preferred model. Justify your answer. 08
  24. Describe the role of management on software development 08
  25. What is software engineering? Describe the following process model with their relative merits and demerits 10\*
    - i. Waterfall model.
    - ii. Evolutionary development.
  26. What are the objectives of software engineering? 05\*
  27. What software engineering? How is it different from traditional engineering approach? 05\*
  28. Explain four significant attributes that every software product should possess. 05\*
  29. What is the need for validating the requirements? Explain any requirement validation techniques. 10\*
  30. Differentiate between the following terms 05\*
    - i. Milestone and deliverable
    - ii. Requirements Definition and Specification
  31. Distinguish between a software product and a software process. 06\*
  32. Explain different process models along with their relative merits and demerits. 14\*
  33. Give a brief description of software prototyping and briefly discuss the various prototyping techniques. 08\*
  34. What are software engineering methods? What are its components? 06
  35. What are the key challenges facing software engineering? 06
  36. "Software engineers have responsibilities to the engineering profession and society. They should not simply be concerned with the technical issues". Justify the statement. 08

## **REQUIREMENT ENGINEERING**

**OBJECTIVE:** The objective of this topic is to understand the

- Basic requirement analysis process
- Various system models
- Requirement definition and specification
- Software requirement and specification document
- Software prototyping
- Various prototyping techniques

37.	What is SRS?	05
38.	Explain the DFD?	05
39.	What is structured analysis?	05
40.	Write a SRS for the following: a) Student registration system b) Diet management control	08
41.	Describe the basic goals and desirable characteristics of an SRS. Discuss the important system properties that an SRS should specify.	05
42.	Develop a zero level and first level DFD for payroll system with suitable specification.	05
43.	Explain DFD and their role in problem analysis.	05
44.	Develop a DFD for a computerized evaluation scheme. Assume marks of all students are stored in a file.	05
45.	What are the different aspects that should be specified in an SRS?	08
46.	Develop a zero level and first level DFD for banking system with suitable specifications.	08
47.	How can u specify 'user friendliness' of the software in the quantitative terms?	05
48.	With an example describe the role of SRS dataflow diagrams in structured system analysis.	05
49.	Write an example; illustrate the use of prototyping as a method for problem analysis. Discuss its advantages and disadvantages.	05
50.	Describe the structure of SRS and discuss the technologies for verifying the requirements.	05
51.	Explain the following terms 1. Ethnography 2. Data dictionary 3. COTS 4. Stake holder	05*
52.	How to organize so that it can be easily manageable?	05*
53.	What is software prototyping? Explain throwaway and evolutionary prototyping techniques.	10*
54.	Draw a DFD for the following applications A salary system which computes employee salary per week and deductions. Input in to the system is a worksheet containing empid, name, number of hours worked, etc. System maintains table holding tax rates and pay rate for various employee classes. The output is the cheque to the employee.	06*
55.	What are the differences between requirements definition and requirements specification?	06*
56.	Discuss the requirements change management process.	06
57.	Suggest who will be stakeholders in a university management system. Explain why it is almost inevitable that the requirements of the different stakeholders will conflict in some ways.	06
58.	Who should be involved in a requirements review? Draw a process model showing how a requirements review might be organized.	06

## **SOFTWARE DESIGN**

**OBJECTIVE:** The objective of this topic is to understand the

- The software design process
- Design principles
- Function oriented design
- User interface design

59.	How are changes in software specifications monitored at various stages of software development?	05
60.	Why is design a two level process? Why should the system design be furnished before starting the detail design, rather than starting the detailed design right after the	08

	requirements?	
61.	How are cohesion and coupling related? Give an example of where cohesion increases and coupling decreases.	05
62.	What is the fundamental difference between the approaches of structured design and object oriented design?	08
63.	Differentiate between top down approach and bottom approach.	05
64.	Develop a structured design for interactive user interface queries for basic income tax information with suitable assumptions.	08
65.	What is the difference between functional abstraction and data abstraction?	08
66.	If a module has logical cohesion what kind of coupling is this module likely to have with others?	08
67.	What are major methods of verifying design? If the design is expressed in a normal language, can an automated tool help in verification and in what manner?	05
68.	What is object oriented design? Explain with a real world problem.	05
69.	Give design specification and design document for a library system.	05
70.	What needs are to be specified during detail design? What are the described characteristics of a language to express detailed design?	08
71.	What is the relation between cohesion metric and cyclomatic complexity?	08
72.	What are the methods of verifying a detailed design? Which of these in your opinion is likely to be most effective in revealing design errors?	08
73.	Describe briefly methods of code verification. Which of these is best suited for "catching" the maximum number of errors and which of these is the most cost effective?	05
74.	What are the major concepts that help in making program more readable?	05
75.	"object-oriented design tends to be programming language independent" why? What is the fundamental reference of this design approach compare to structural design?	08*
76.	Define the object-oriented design methodology and contrast it with structure design methodology.	05
77.	Explain the module level concepts of coupling and cohesion; and the relation between them.	05
78.	What is cyclomatic number? With an example, describe how this metric is calculated? Compare the metric with cohesion metric.	05
79.	Explain the different methods of verifying a detailed design and discuss their effectiveness in revealing design errors.	05
80.	Write short notes on structure charts and PDL.	05
81.	Discuss coupling and cohesion.	05
82.	Discuss structure and cohesion.	05
83.	Discuss object oriented approach to software design.	05
84.	What is Halstead's measure? Explain.	05
85.	Write short notes on PDL.	05
86.	What is the difference between functional abstraction and data abstraction? Explain with suitable example.	05
87.	How are cohesion and coupling related? Give example where cohesion increases and coupling decreases.	05
88.	What is the fundamental difference between the approaches of structures chart, extend the notion for structured charts to represent the structure of programs where modules communicate using global variable?	05
89.	What is the fundamental difference between the approaches of structures design and object oriented design?	05
90.	Give the design specification and design document for student information system.	05
91.	What are the different methods of verifying a detailed design? Which of them will reveal design error more effectively?	05
92.	Describe the basic design principles of problem partitioning and abstraction.	05
93.	What is structure chart and how are different types of modules represented in a structure chart? Illustrate with suitable example.	05
94.	Explain the use of design reviews in verifying a design.	05
95.	With suitable examples illustrate the use of process design language for specifying the detailed design.	05
96.	Explain with suitable examples and illustrate the use of process design language for specifying the detailed design.	05
97.	Explain a technique for computing the cohesion metric of a module.	05
98.	Explain the use of critical design reviews for verifying a detailed design.	05
99.	Briefly outline the process of modular design of a system.	05

100. Define and Explain the measure effective modularity of software systems.	05
101. Discuss the important methods for producing quantitative specifications for modular interfaces with suitable example.	05
102. Describe the object oriented design methodology and discuss its comparative advantages and disadvantages.	05
103. Describe the salient features of process design language and discuss its importance in software engineering.	05
104. Explain the most commonly used size metrics and complexity metrics for code.	05
105. What is cyclomatic number? Discuss its significance with suitable example.	05
106. Why is it important to use different notations to describe design?	04*
107. Differentiate between function oriented designs and object oriented design strategies.	10*
108. Explain why maximizing cohesion and minimizing coupling leads to more maintainable systems. What other design attributes influence system more maintainable.	10*
109. What are the principles to be followed while designing user interface.	05*
110. Write a bubble sort procedure in PDL and its flow-graph. Also calculate McCabe's Cyclomatic complexity?	05
111. Describe system architecture modeling with the help of a neat diagram.	06*
112. Explain the different software design strategies with proper illustrations.	08*
113. Write a brief note on data flow design principle.	04*
114. Describe user interface design principles.	06*
115. What are the difference between coupling and cohesion?	06*
116. Write a short note on data dictionary.	08*
117. Explain why it may be necessary to design the system architecture before the specifications are written.	04
118. Giving reasons for your answer, suggest an appropriate structural model for the following: An automated ticket issuing system used by passengers at a railway station. A computer controlled video conferencing system which allows video, audio and computer data to be visible to several participants at the same time.	08
119. Suggest situations where it is unwise or impossible to provide a consistent user interface.	04

## **VERIFICATION AND VALIDATION**

**OBJECTIVE:** The objective of this topic is to understand the

- Software reliability
- Software reuse
- Software testing
- Clean room software testing

120. Explain bottom up and top down testing.	05
121. Define error, fault and failure; bring out the difference between each with a suitable example.	05
122. What is functional testing and how it differs from structural testing? Explain any two methods for selecting test cases for functional testing.	05
123. Explain the complete testing process with help of suitable diagrams.	08
124. Write short notes on test oracles, instrumentation for structural testing.	05
125. Discuss the various levels of testing.	05
126. Compare functional testing with structural testing.	05
127. Write short notes on verification and validation.	05
128. Why does software fails after passes acceptance testing?	05
129. Discuss different levels of testing and set goals of these different levels.	05
130. Explain error, fault and failure. What is the difference between a failure and a fault? Explain bottom up and top down testing.	06
131. Describe the levels of testing and goals at each level.	05
132. Discuss different coverage criteria for structural testing.	05
133. What is a test plan and what does a typical test plan contains?	05
134. Write short notes on psychology of testing and mutation testing.	05
135. Explain the following: a) Verification b) Validation.	05
136. Discuss the important objectives of testing software system.	05
137. Discuss the problems with exhaustive testing.	05
138. Explain the following with ex. a) Basic path testing b) Control structure testing.	05
139. Short notes on bottom up testing, criteria for completion of testing and debugging.	06
140. Write short notes on a) Fault tolerance b) Defensive programming c) Software re-	06

usability.	
141. What is software reliability? Explain reliability metrics.	08*
142. Explain briefly the various software reliability metrics.	08*
143. Briefly bring out the difference between verification and validation.	06*
144. Explain the stages of testing process.	06*
145. Describe briefly the following:	08*
a) Defensive programming	
b) Cleanroom software development	
146. Explain the advantages and disadvantages of reusing software components.	06*
147. How do you distinguish between a program inspection and other types of quality reviews?	04
148. What are static program analyzers? What are the stages involved in static analyses?	08
149. Discuss the levels of testing involved in object oriented system.	06

## **CRITICAL SYSTEMS**

**OBJECTIVE:** The objective of this topic is to understand the

- Dimensions of dependability
- Importance of dependability to critical systems
- Critical system specification

150. What are the most important dimensions of system dependability?	04
151. What are the factors you need to consider when specifying the overall system reliability?	04
152. Explain why there is a close relationship between system availability and system security?	04
153. Suggest six reasons why dependability is important in critical systems.	06
154. Discuss the stages involved in security specification process.	06

## **SOFTWARE MANAGEMENT**

**OBJECTIVE:** The objective of this topic is to understand the

- Project Planning
- Project Scheduling
- Managing people
- Cost Estimation

155. Explain the importance of project staffing.	04*
156. What are the characteristics of clean room software development?	05*
157. What is SEI process maturity model?	05*
158. "Members of a well –led, cohesive group are loyal to the group". Justify	06
159. Explain in detail the various management activities.	08
160. Explain in detail software cost estimation.	08
161. Explain cost estimation techniques.	08
162. What is COCOMO model? Describe its approach to estimate person months.	08
163. Briefly explain how algorithmic cost can be used for estimating software cost.	10*
164. Write short note on:	15*
a) CASE tools	
b) Software quality assurance	
c) SEI CMM	
165. Explain in detail	15
a) Software metrics	
b) Quality Metrics	
c) SEI-CMM Model	
d) CASE	
e) Process analysis and measurement.	

## **SOFTWARE EVOLUTION**

**OBJECTIVE:** The objective of this topic is to understand the

- How legacy systems are critical to the operation of many businesses
- Legacy system structures
- Re-engineering as an option for software system evolution
- Activities such as reverse engineering and program restructuring

166. Why is source level translation necessary in some cases?	04
167. List the principle factors that affect re-engineering costs.	04
168. Differentiate between reverse engineering and re-engineering.	04

169. What is data re-engineering? List out the approaches to data engineering. 06
170. Under what conditions should a legacy system be scrapped rather than being re-engineered? 06
171. "Replacing a legacy system is a risky business strategy." Justify. 06
172. Explain the activities involved in re-engineering process with an illustrative figure. 10
173. Highlight the importance of DFD in Software Engineering life cycle. 06
174. Bringout the advantages and disadvantages of a shared repository. 10
175. Mention five levels in P-CMM model. Explain each of them. 08
176. COCOMO Model. 10

Marks	No of Questions
04	14
05	92
06	25
08	34
10	09
14	01
15	01
<b>Total</b>	<b>176</b>

## NOTES

# CS63: COMPUTER GRAPHICS

Faculty:

No of Hours: 52

Class #	Chapter Title/ Reference Literature	Topics to be covered	% of portions covered	
			Reference chapter	Cumulative
1.	<b>Chapter #:1,4</b> Introduction T1:Page#1-22 Graphics Hardware T1:Page#145-197 & Page#155-165	<b>Introduction</b> - Image processing as Picture analysis, Representative uses of computer graphics, Advantages of Interactive Graphics, Classification of applications	15%	15%
2.		Development of Hardware and Software for Computer Graphics		
3.		Conceptual Framework For Interactive Graphics		
4.		Raster scan Display Systems, Video Controller		
5.		Random Scan Display processor		
6.		Problems on Pixels / resolution		
7.	<b>Chapter #:3</b> Raster Graphics Algorithms T1 : Page #67-140 & Page #88- 91	Input Devices for operator Interaction, Image Scanners.	25%	40%
8.		Hardcopy Technologies		
9.		<b>Raster Graphics Algorithms-</b> Overview, Scan converting lines		
10.		BILD Algorithm		
11.		Bresenham's Algorithm - Line, problems		
12.		Bresenham's Algorithm - Circle, problems		
13.		Scan Converting Circles		
14.		Filling polygons		
15.		Filling rectangles, Filling ellipse arcs,		
16.		Pattern filling, Thick primitives		
17.		Line style and pen style, Clipping in raster world		
18.	<b>Chapter #:5</b> Geometrical Transformations T1:Page#201-226	Clipping Lines	25%	65%
19.		Clipping Lines continued, Clipping Circles and Ellipses		
20.		Clipping Polygons, Generating Characters ,Antialiasing		
21.		<b>Geometrical Transformations –2D transformations</b>		
22.		Homogeneous coordinates and matrix representation of 2D transformations		
23.		Compositions of 2D transformations		
24.		The window-to-viewport transformations, efficiency		
25.	Matrix representation of 3D transformations			
26.	<b>Chapter #:6</b> Viewing in 3D T1:Page#229-252	Composition of 3D transformations	10%	75%
27.		Transformations as a change in coordinate system		
28.		<b>Projections - Introduction</b>		
29.		Top, Front, Side, View		
30.	<b>Chapter #:8,9</b> Interaction Techniques, Dialogue Design and user Interface T1:Page#347-388	Perspective View	5%	80%
31.		Specifying an arbitrary 3D view		
32.		Examples of 3D viewing		
33.		<b>Interaction hardware</b>		
34.		Basic interaction tasks, Composite interactions tasks,		
35.		The form and content of user-computer dialogues,		
36.	User-interface styles			
37.	<b>Chapter #:10</b> Three - Dimensional Object Representations T2 :page #304-404 T2:Page#312-326,#345-358,#387-404	Important design considerations	10%	90%
38.		Modes and syntax, Visual design.		
39.		The design methodologies		
40.		<b>Three – Dimensional Object Representations</b> Polygon		
41.		Surfaces, tables, Equations, Meshes		
42.		Curved Lines & Surfaces		
43.		Quadric Surfaces		
44.	<b>Chapter #15</b> Visible Surface Determination T1:Page#392-430	Bezier Curves & Surfaces	10%	100%
45.		B-Spline Curves & Surfaces		
46.		Octrees, BSP Trees		
47.		Fractal Geometry Methods.		
48.		<b>Visible Surface Determination</b> -Functions of two variables		
49.		Techniques for efficient visible-surface algorithm		
50.		Algorithms for visible-line determination		
51.		Z-buffer algorithm, list-priority algorithms, Scan-line algorithms,		
52.		Area-subdivision algorithms, algorithms for octrees,		
		Algorithm for curved surfaces, visible-surface ray tracing		
	Solutions for Previous Test Question Paper			
	Revision			
	Revision			
	Revision			

**Literature:**

	Code	Title and Author	Publication specification		
			Edition	Publication	Year
Text Book	T1	Computer Graphics by Foley,Vandam,Feiner,Hughes	2 <sup>nd</sup>	Addison Wesley	1997
Text Book	T2	Computer Graphics by Hearn & Baker	2 <sup>nd</sup>	Pearson Education	2003
Reference Book	R1	Computer Graphics by Zhigang Xiang & Roy Plastok	2 <sup>nd</sup>	Schaum's Outlines	--
Reference Book	R2	Procedural Elements for Computer Graphics	2 <sup>nd</sup>	Schaum's Outlines	2001

## QUESTION BANK

### INTRODUCTION

**OBJECTIVE:** A detailed fundamental study of

- Basics of Computer Graphics
- Hardware – Display (CRT), Controllers, Scanners.
- Software package SRGP

Coverage of syllabus according to syllabus of VTU

Chapter – 1: Introduction

Chapter – 4: Graphics hardware

1. Give classification of monitors. 4\*
2. Write short notes on : 5x2  
a) Thick primitives. b) Classification of application of computer graphics \*
3. Bring out the advantages and disadvantages of raster scan w.r.t random scan devices. 5\*
4. With the necessary block diagram, explain the working of a raster display system. 10\*
5. Write a note on graphics software standard. 5\*
6. What is modeling? Briefly explain. 5\*
7. Explain sample mode and event mode. Give procedures supported by SRGP for above modes. 5\*
8. Define dot size, addressability, inter for distance, resolution and vertical retrace time. 5\*
9. Explain the role of display processor in a graphics system. 5\*
10. With briefly about the applications of computer graphics. 10\*
11. Bring out differences between graphics and image processing 8
12. List out some of input devices and explain each of them 10
13. Discuss the working principle of: i. Mouse ii. Tablet iii. Light pen iv. Keyboard 10
14. Write a detailed note SRGP canvas and various functions used for creating and modifying a canvas. 10\*
15. With neat diagram explain the raster display system with integrated processor. 10\*
16. What are the differences between raster scan & random scan? Display techniques for Video monitors. Explain the DVST system. 10\*
17. Explain the working of shadow mask CRT with neat sketches. Compare its performance with beam penetration CRT. 10
18. Explain the working of shadow mask CRT with neat sketches. Compare its performance with beam penetration CRT. 10
19. Explain the principle of operation of random scan systems and raster scan systems. Give their merits and demerits. 10
20. Explain with neat diagram the architecture of 10\*  
i) Vector display ii) Raster display

### RASTER GRAPHICS ALGORITHMS

**OBJECTIVE:** The purpose of this chapter is to look at SRGP from a **package implementer's point of view** (i.e. Fundamental algorithms).

- Scan converting primitives to pixels subject to their attributes.
- Clipping them against an upright rectangle
- Approach is through 2D-integer grid
- Minimizing jaggies by use of systems ability to vary a pixel's intensity.

21. Explain DDA line Drawing algorithm with example. 5
22. Bring out the advantages and disadvantages of Bresenham's algorithm over DDA algorithm. 5
23. Explain the Bresenham's algorithm for Ellipse generation. 5
24. Consider the line from (0,0) to (5,7), Use the mid point line scan conversion technique to rasterise this line. 5\*
25. Explain the attributes of line. 5
26. What is meant by clipping? Explain Cohen-Sutherland algorithm with neat figure. 10\*
27. With an algorithm, Explain mid point line scan conversion technique. 10\*
28. Explain Cyrus beck algorithm with neat diagram and give an algorithm for generalized line clipping. 10\*
29. Bring out the list of points to be plotted for generating a circle with center as (100,100) and Radius 25 using Bresenham's circle generating algorithm. 10

30. Explain the various techniques of area filling.	10
31. Explain scan line algorithm.	10
32. Bring out the list of points to be plotted for the points(120,110)(130,118) using a) DDA algorithm      b) Bresenham's algorithm	10
33. Give and explain midpoint circle scan-conversion algorithm.	10*
34. Obtain the coordinate points for a straight line between the points (40,52) and (48,62) Using the DDA algorithm.	10
35. With an algorithm, explain mid point line algorithm.	10*
36. What is meant by clipping? Explain Cohen Sutherland algorithm.	10*
37. Discuss typical attributes of line and their implementation.	10*
38. List the steps required to scan convert a circle using Bresenham's algorithm. Explain how this can be modified to generate elliptical shapes. Outline a method for Antialiasing a circle boundary.	10*
39. What is Antialiasing? Explain the various Antialiasing techniques used for better quality images.	10*
40. Describe the technique for polygon clipping developed by Sutherland and Hodgman.	10*
41. Explain Cohen-Sutherland algorithm for clipping.	10*
42. Explain various techniques for text clipping and blanking.	10*
43. With illustrations, show the working of the Sutherland-Hodgman's polygon clipping algorithm.	10*
44. Explain the Sutherland and Hodgman polygon clipping technique and give its flowcharts.	10*
45. Briefly explain the circle generating algorithms.	10*
46. Discuss the Bresenham's circle generating algorithm. Get all the pixel co-ordinates to draw a circle of radius 10 with center.	10*

## **GEOMETRICAL TRANSFORMATIONS**

**OBJECTIVE:** The chapter introduces the basic 2–Dimensional and 3–Dimensional transformations used in CG.

<ul style="list-style-type: none"> <li>• Basic transformations : Translation, Scaling, Rotation, Reflection</li> <li>• Composite transformation</li> <li>• Entire process is to change the position, orientation and size of objects in drawing.</li> </ul>	
47. What is concentration? How does it affect the efficiency of transformation?	5
48. Explain with illustration the basic 2D geometrical transformations used in CG. Also give their matrix representations in homogeneous coordinate system.	10*
49. Find the transformation of triangle A(1,0), B(0,1) and C(1,1) by 1. Rotating $45^\circ$ about the origin and then translating one unit in x and y directions. 2. Translating one unit in x and y direction and then rotating $45^\circ$ about the origin	10*
50. Prove that successive 2D translations are additive.	10*
51. Briefly explain window to viewport transformation. Given a window and viewport, derive the transformation matrix that maps the window from world co-ordinates into the viewport in screen co-ordinates.	10*
52. Explain with examples, the basic 2D geometrical transformations used in computer graphics.	10*
53. Magnify the triangle with vertices A(0,0),B(1,1),C(5,2) to twice its size while keeping C(5,2) fixed. Derive the translation matrix.	10*
54. Prove that multiplication of transformation matrices for the following sequence of operations is commutative. i. Two successive rotations ii. Two successive scaling also obtain 2D-transformation matrix for translation.	10*
55. Explain the concept of window, viewport and window to viewport transformation.	10*
56. Define translation, scaling, rotation and mirror reflection transformations. Obtain the matrix Representation for translation, scaling and rotation in clockwise direction. Also express them in homogeneous coordinate system.	10*
57. Explain the terms window, viewport and zooming. Discuss window to viewport transformation.	10*
58. Compare the Cohen-Sutherland and Liang-Barsky line-clipping algorithm.	10*
59. Determine the form of the transformation matrix for a reflection about an arbitrary line with equation $Y=X+B$	10
60. Discuss window to viewport transformation.	10
61. Prove that multiplication of two successive rotational transformational matrices follow the commutative law.	10
62. Consider a line p1p2 from (x1,y1,z1) to(x2,y2,z2) in a right handed co-ordinate system. Rotate this line, such that p1 lies on the origin and p2 on positive 2-axis. Find the transformed line.	10*

## **VIEWING IN 3D**

**OBJECTIVE:** Specification of a view volume in the world, a projection into a projection plane and a view port on the view surface.

- Window to viewport display.
- Variety of strategies visible surface determination and shading.

63.	Write a note on perspective and parallel projections.	10*
64.	Briefly explain types of orthographic projections.	10*
65.	Obtain the matrix representation of a point about an arbitrary axis in 3 dimensions.	10*
66.	Describe how viewing transformations are made in case of 3D viewing.	10*
67.	Given a window and viewport, derive the transformation matrix that maps the window from world coordinate system into a viewport in screen coordinate system.	10*
68.	Get the composite transformation matrix to rotate the 3D object by 45 about an axis Passing through the points(1,1,1) and (6,6,6).	10*
69.	Describe various techniques used to display 3D images on 2D screen.	10*
70.	How are perspective projections being categorized? Explain briefly with examples.	10*
71.	Explain the parallel and perspective projection techniques along with their transformation matrix.	10*
72.	Give the steps involved in rotating a 3-D object about an arbitrary axis and derive the Transformation matrix.	10*

## **INTERACTION TECHNIQUES, DIALOGUE DESIGN AND USER INTERFACE SOFTWARE**

**OBJECTIVE:** A discussion on basic elements of user interfaces.

- Input devices, interaction techniques and interaction tasks.
- Ways to use input devices to enter information into a computer.
- Fundamental types of information entered with interaction techniques are the primitive building blocks from which User Interface is crafted
- Attractiveness of the interfaces is a real market place concern.

73.	Discuss the advantages and disadvantages of interaction devices based on three levels.	5
74.	Discuss briefly the main issues related to the position interaction tasks.	10*
75.	Compare single level versus hierarchical design method.	5
76.	Write short notes on: User interface styles.	4*
77.	Write a note on 3D interaction tasks.	5
78.	Write short notes on: Image scanner.	4*
79.	Write the state diagram for rubber band line drawing.	5
80.	What are different methods involved in dynamic manipulation?	5
81.	Discuss the following i. Locator devices    ii. Keyboard devices    iii. Valuator devices    iv. Choice devices	10*
82.	Write a note on 3D interaction devices.	10*
83.	Explain the different types of interaction tasks.	10
84.	Define terms, co-ordinate terms, resolution, grid, feedback, direction preference, learning time.	10
85.	Explain the objects that can be selected by the method of naming and pointing.	10
86.	Write a note on polygon meshes, highlighting the three poly-mesh representations.	10*
87.	Explain select interaction tasks.	10
88.	Explain the following graphical; input techniques. i. Menu selection    ii. Rubber handling    iii. Blinking and highlighting    iv. Dragging	10
89.	Mention and explain three major forms of composite interaction tasks.	10
90.	Explain rubber band technique.	10
91.	Explain different forms of computer dialogue boxes.	10
92.	Explain B-spline techniques of generating curves and illustrate with an example.	10*

## **THREE DIMENSIONAL REPRESENTATIONS**

**OBJECTIVE:** The need to represent curve's surface arises in 2 cases

- Modeling existing objects
- Modeling "from scratch"

Introduction to a general area of surface modeling – a quite broad area, three most common representations for 3D surfaces are discussed

- Polygon meshes surfaces
- Parametric curves surfaces
- Quadric surfaces

- |  |     |
|--|-----|
| 93. Define the terms with respect to curve generation<br>i. Control points ii. Order of continuity iii. Knots iv. Curve points | 5*  |
| 94. Explain the Bezier technique of generating curves and illustrate it with an example.                                       | 10* |
| 95. What is Bezier curve? Explain how Bezier can be generated.   | 8   |
| 96. Explain parametric cubic curves.   | 10  |
| 97. Explain Bezier technique for generating curves and enumerate its important properties.                                     | 10* |
| 98. Define polygon meshes. Explain different methods of representing polygon meshes.   | 10  |
| 99. Write a note on hermite cubic curves.  | 10  |

### **VISIBLE SURFACE DETERMINATION**

**OBJECTIVE:** Given a set of 3D objects and a viewing specifications, we wish to determine which lines / surfaces of the objects are visible, either from center of projections (prospective) or along the direction of projection (parallel) which enables to display only the visible lines / surfaces.

- |   |     |
|---|-----|
| 100. Explain scan line method of removing hidden surfaces.  | 5   |
| 101. Explain the following terms<br>i. Coherence ii. Object coherence iii. Depth coherence iv. Implied edge coherence | 8   |
| 102. Discuss the object space and image space methods of hidden surface removal.<br>Explain depth buffer algorithm.   | 10  |
| 103. Explain depth-sorting algorithm for eliminating hidden surfaces.   | 10* |
| 104. Explain the technique for efficient visible surface algorithm.   | 10  |
| 105. Write a note on perspective transformation.  | 10  |
| 106. Explain the painters algorithm for eliminating hidden surfaces   | 10* |
| 107. Explain Robert's algorithm and apple's algorithm.  | 10  |
| 108. Give the Z-Buffer algorithm for hidden surface removal.  | 10* |
| 109. Explain z-buffer algorithm.  | 10* |
| 110. Explain List priority algorithm.   | 10  |
| 111. Explain algorithms for octrees.  | 10  |
| 112. Write an algorithm for building binary space partitioning (bsp) tree.  | 10  |
| 113. Explain Warnock's algorithm and Weiler-Atherton algorithm.   | 10  |
| 114. Discuss area sub division algorithms.  | 10  |

Marks	No of Questions
4	05
5	21
8	03
10	85
<b>Total</b>	<b>114</b>

### **NOTES**

# CS64: COMPUTER NETWORKS – I

**Faculty:**

**No. of Hours: 52**

Class #	Chapter Title / Reference Literature	Topics to be Covered	% Portions covered	
			Each chapter	Cumulative
1	Introduction 10 sessions T1: Chap. 1 Pages 1-37, 50-65	<b>Uses of Computer Networks</b>	7 Hrs, 13.5 %	13.5%
		Business Applications		
		Home applications		
		Mobile users		
2		<b>Social issues</b>		
3		<b>Network Hardware</b>		
		Local Area Networks		
		Metropolitan Area Networks		
		Wide Area Networks		
		Wireless Networks		
4-5		<b>Network Software</b>		
		Protocol Hierarchies		
		Design issues for the layer		
		Connection oriented and Connectionless Services		
		Service primitives		
	The relationship of services to protocols			
6-7	<b>Example of Networks</b>			
	Internet,			
	Connection oriented networks			
	X.25			
	Frame relay			
	ATM			
8-9	Local Area Networks 13 sessions T2: Chap. 6 Pages 421-459, 465-479	<b>LAN Protocols</b>	10 Hrs, 19.2%	32.7%
		LAN Structure		
		The Medium Access Control Sub layer		
10-12		The Logical Link Control Sub layer		
		<b>LAN standards</b>		
		Ethernet Protocol		
		Frame structure		
		Physical layers		
		Fast, Gigabit and 10 Gigabit Ethernet		
		Token Ring and IEEE 802.5 LAN Standard		
		FDDI		
13-14		<b>Wireless LAN and IEEE 802.11 Standard</b>		
		Ad-hoc and infrastructure Networks		
		Frame structure and addressing		
		Medium access control		
15-17	<b>LAN Bridges and Ethernet Switches</b>			
	Transparent bridges			
	Source routing bridges			
	Mixed media bridges,			
	Virtual LANs			
18-21	The Network Layer 21 Sessions T1: Chap. 5 Pages 343- 473	<b>Network Layer Design Issues</b>	18 Hrs, 34.6%	67.3%
		Store and Forward Packet Switching		
		Services provided to the Transport Layer		
		Implementation of connectionless services		
		Implementation of connection-oriented service		
		Comparison of virtual circuit and datagram subnets		
22-26		<b>Routing Algorithms</b>		
		The optimality principle		
		Shortest path routing		
		Flooding		
		Distance Vector Routing		
		Link State Routing		
		Hierarchical routing		
		Broadcasting routing		
		Multicasting routing		
	Routing for Mobile hosts			
	Routing in ad-hoc Networks			
	Node look up in Peer-to-Peer Networks			
27-28	<b>Congestion Control Algorithms</b>			
	General principles			
	Congestion Prevention policies			
	Congestion control in Virtual circuit subnets			
	Congestion control in Datagram Subnets			
	Load shedding			
	Jitter control			
29-30	<b>Quality of Service</b>			
	Requirements			
	Techniques for achieving good quality of service			
	Integrated Services			
	Differentiated services			
31-32	<b>Internetworking</b>	How networks differ		
		How networks can be connected		
		Concatenated Virtual circuits		
		Connectionless Internetworking		
		Tunneling		
		Internet Routing		
	Fragmentation			
33-35	<b>The Networking Layer in the internet</b>			
	The IP Protocol			
	IP Addresses			
	Internet Control Protocols			

		OSPF		
		BGP		
		Internet Multicasting, Mobile IP		
		IPv6		
<b>36-37</b>	<b>The Transport Layer</b> <b>14 sessions</b> T1: Chap. 6 Pages: 481-573	<b>The Transport Service</b>	<b>12 Hrs, 23.1%</b>	<b>90.4%</b>
		Services provided to the upper layer		
		Transport Service Primitives		
		Berkeley Sockets		
<b>38-40</b>		<b>Elements of Transport Protocols</b>		
		Addressing		
		Connection establishment		
		Connection release		
		Flow control and Buffering		
		Multiplexing		
		Crash recovery		
<b>41</b>		<b>Internet Transport Protocol: UDP</b>		
		Introduction to UDP		
		Remote Procedure Call		
		The Real-Time Transport Protocol		
<b>42-45</b>		<b>Internet Transport Protocol: TCP</b>		
		Introduction to TCP		
		The TCP service model		
		The TCP protocol		
		The TCP Segment header		
		TCP Connection Establishment and Release		
		Modeling TCP connection establishment		
		TCP Congestion Control and TCP Timer Management		
		TCP transmission policy		
		TCP congestion control		
		TCP Timer management		
	Wireless TCP and UDP			
	Transactional TCP			
<b>46</b>	<b>Performance Issues</b>			
	System Design for better performance			
	Protocols for Gigabit Networks			
<b>47-52</b>	<b>ATM Networks</b> <b>8 sessions</b> T2: Chap. 9 Pages 661-698	ATM Basics	<b>5 Hrs, 9.6%</b>	<b>100%</b>
		ATM – BISDN Reference Model		
		ATM Layer		
		ATM Adaptation Layer		
		ATM Signaling		
		PNNI Routing		
	Classical IP over ATM			

#### Literature:

Book Type	Code	Title & Author	Publication Info		
			Edition	Publisher	Year
Text Books	T1	Computer Networks, Andrew S Tannenbaum	4th	PHI	2002
Text Books	T2	Communication Networks – Fundamental Concepts and Key Architectures, Alberto Leon Garcia & Indra Widjaja	2nd	Mc Graw Hill	2004
Reference Books	R1	Data Communications and Networking, Behrouz A. Forouzan	3rd	Mc Graw Hill	
Reference Books	R2	Data and Computer Communication, William Stallings	5th	PHI	2002
Reference Books	R3	Computer Networking – A Top-Down Approach Featuring the Internet, James F. Kurose and Keith W. Ross	2nd	Pearson	
Reference Books	R4	Computer Networks, Larry L. Peterson and Bruce S. Davie	3rd	Elsevier-Morgan Kauffmann	

## QUESTION BANK

### INTRODUCTION

**OBJECTIVE:** Computer network means an interconnected collection of autonomous computers. These days it has become mandatory for computer professionals to know about networks, their configurations, topologies, and other various issues used in networking. On completion of this chapter, you will know

- Definition, objectives and uses of computer networks
- Network Topologies
- Network hardware and Software
- The Internet
- X.25, Frame Relay & ATM

- |   |            |
|---|------------|
| 1. Explain the following with example:<br>(a) Broadcast network (b) Point-to-Point network (c) LAN (d) WAN (e) MAN<br>(f) Wireless networks | 03<br>each |
| 2. Explain how the Store-and-Forward scheme differs from Broadcasting scheme.   | 04         |
| 3. Explain Service Primitives used in Network Software.   | 04         |
| 4. Explain the terms: Protocol, SAP, Subnet, Internet and PDU.  | 05*        |
| 5. Briefly discuss the design issues in computer networking.  | 05*        |
| 6. Explain the protocol hierarchies used in network software.   | 06         |
| 7. Differentiate between Connection-Oriented and Connectionless Services.   | 06         |
| 8. What are computer networks? What are their objectives (uses)?  | 08         |
| 9. What is a Computer Network? How is it different from distributed system? What are the goals of networking?                               | 10*        |
| 10. What is encapsulation? Explain.   | 04         |
| 11. Explain the following:  | 10*        |
| i) Connection – oriented and connectionless services  |            |
| ii) Confirmed and unconfirmed services  |            |
| iii) Services and protocols   |            |
| 12. Compare/Contrast LAN, WAN, MAN.   | 12         |
| 13. Distinguish between unicasting, multicasting and broadcasting.  | 04         |
| 14. Give the relationship of Services to Protocols.   | 03         |
| 15. Write short notes on ARPANET.   | 10         |
| 16. What were the four traditional applications of the Internet?  | 04         |
| 17. Write notes on NFSNET.  | 05         |
| 18. Explain the ATM Reference Model.  | 08         |
| 19. Explain an ATM virtual circuit and an ATM cell.   | 04         |
| 20. Write about X.25 and Frame relay.   | 04         |
| 21. Explain the terms: IMP, NAP and AAL.  | 03         |
| 22. What are the differences between [June 2004]  |            |
| a) distributed networks and distributed networks  | 03         |
| b) Protocols and services   | 03         |
| c) TCP / IP model and OSI reference model   | 03         |
| 23. Distinguish between Broadcast networks and point-to-point networks.   | 4          |
| 24. With a neat diagram explain TCP/IP model.   | 10         |

### LOCAL AREA NETWORKS

**OBJECTIVE:** The development of LANs was motivated by the need to share resources and information among workstations in a department or workgroup. The short distances between computers imply that low-cost, high-speed, reliable communications is possible. The purpose of this chapter is

- To explain the LAN Structure, the Medium Access Control Sub-layer and the Logical Link Control Sub-layer.
- To introduce the Fiber Distributed Data Interface (**FDDI**).
- To give a detailed presentation of the following
  - Ethernet and IEEE 802.3 LAN Standard
  - Token-Ring and IEEE 802.5 LAN Standard
  - Wireless LANs and IEEE 802.11 Standard
- To learn about the various LAN Bridges and Ethernet Switches.

- |                                |    |
|--------------------------------|----|
| 25. Explain the LAN Structure. | 04 |
| 26. What are the uses of NIC?  | 04 |

27.	Contrast the various IEEE 802 LAN standards with a layered model.	10
28.	Give the LLC PDU structure.	05
29.	Contrast the services provided by the MAC and LLC sub-layers.	05
30.	Write notes on LLC sub-layer.	08
31.	What are NIC, SAP and SNAP?	06
32.	Explain the Ethernet protocol.	08
33.	Compare the three types of physical addresses.	03
34.	Give the IEEE 802.3 MAC frame structure.	06
35.	Explain the Ethernet II frame structure.	06
36.	Contrast IEEE 802.3 10 Mbps medium alternatives i.e., 10Base2, 10Base5, 10BaseT and 10BaseF w.r.t. Medium, Max. Segment length and Topology.	10
37.	Write notes on Fast Ethernet.	06
38.	Compare 1000BaseSX, 1000BaseLX, 1000BaseCX and 1000BaseT.	06
39.	Write short notes on 10 Gigabit Ethernet.	04
40.	Compare various LANs (IEEE 802.3, 802.4, 802.5).	08
41.	Explain Token-Ring Protocol.	06
42.	Explain the IEEE 802.5 Token and data frame structure.	08
43.	Write short notes on FDDI.	10
44.	Discuss the advantages and disadvantages of wireless networks.	06
45.	What are ad hoc and infrastructure networks?	05
46.	Write a note on IEEE 802.11 frame structure.	08
47.	Explain the basic CSMA-CA operation.	06
48.	What is the DCF?	04
49.	Contrast the Transmission of MPDU with and without RTS/CTS.	06
50.	What is the PCF?	05
51.	Write short notes on transparent bridges.	04
52.	Explain the spanning tree algorithm.	06
53.	Describe source routing bridges.	05
54.	What are mixed-media bridges?	04
55.	What are the two approaches to bridging between transparent and source routing bridging domains?	04
56.	Write short notes on VLANs.	06

## **THE NETWORK LAYER**

**OBJECTIVE:** The network is concerned about getting packets from source to destination. It may require making many hops at intermediate routers, i.e., the network layer deals with end-to-end transmission. To achieve its goals, the network layer must know about the topology of the communication subnet (set of all routers) and choose appropriate paths through it. It must ensure that it avoids overloading or under loading routers. It must also resolve the differences that arise when source and destination are in different networks. We shall cover IP protocol (IPv6) and Internet Control Protocols.

57.	Give two example applications for which connection-oriented service is appropriate. Also give two examples for which connectionless service is best.	04
58.	What are routing algorithms? What are their objectives and uses?	04
59.	Mention the drawbacks of Multi destination routing.	04
60.	Differentiate between Flow Control and Congestion Control.	04
61.	Mention at least two differences between leaky bucket and token bucket algorithms.	04
62.	Explain how networks can be interconnected.	04
63.	Explain the implementation of connectionless and connection-oriented services.	08
64.	Differentiate Transparent and Nontransparent fragmentation.	04
65.	Explain the following terms with respect to a network: (a) Repeater (b) Bridge (c) Transport Gateway (d) Application Gateway	04
66.	What is a sink tree?	03
67.	Explain the following routing algorithms taking an example: (a) Shortest Path (b) Flooding (c) Hierarchical (d) Broadcast (e) Multicast	each 04
68.	Write a short note on Routing for mobile hosts.	06
69.	In Split Horizon Hack procedure, mention the special case where the Count to Infinity problem exists.	04
70.	Write a note on Routing in Ad Hoc Networks.	10
71.	Write notes on Distance Vector Routing.	08
72.	What are node identifiers?	04
73.	Differentiate between virtual circuits and datagram subnets.	04*
74.	List the desirable goals in any solution of Mobile IP.	05

75. Explain the IP options of the Internet Protocol.	05
76. What improvements have been made in Ipv6 compared to Ipv4?	05*
77. What are the different devices used in internetworking?	05*
78. Write explanatory notes on Broadcast routing.	05*
79. With an example, explain the reverse path-forwarding scheme.	06
80. Does traffic shaping help prevent congestion control? Explain.	06
81. Compare/ Contrast Adaptive Vs Non-adaptive routing algorithms.	08
82. What is Multicast routing? What are its disadvantages?	08
83. What is congestion control? How it can be prevented?	08
84. Give the general principles of Congestion Control.	06
85. How is congestion controlled in Virtual-Circuit Subnets?	06
86. Explain the Leaky bucket algorithm used in congestion control.	08
87. Explain any 3 congestion control methods in datagram subnets.	06
88. Explain the principle of working of token bucket algorithm.	08
89. Compare Static Routing Vs Dynamic Routing algorithms.	08
90. Explain Flow-Based Routing with an example.	08
91. List various ways in which networks differ.	08
92. Explain internetworking using concatenated virtual circuits.	04
93. Explain the datagram (connectionless) inter-network model.	04
94. With an example, explain what is tunneling.	05
95. Write a note on Fragmentation.	06
96. What is inter-network routing?	05
97. Give an account of the stringency of the QoS requirements for a few applications.	06
98. Clearly explain any 4 techniques for achieving good QoS.	12
99. Distinguish flow-based Vs. class-based Quality of Service.	08
100. Clearly explain RSVP with an example.	08
101. Compare expedited and assured forwarding.	08
102. What is the significance of the MPLS header?	04
103. Compare data-driven and control-driven approaches for creating the forwarding table entries.	04
104. Explain the concept of internetworking.	04*
105. What is Internet Protocol (IP)? Explain its header format clearly.	08
106. Explain the address formats used in IP.	08
107. Explain the Internet Control Message Protocol (ICMP) and its message types.	08
108. Clearly explain the Open Shortest Path First (OSPF).	08
109. Explain the CIDR technique.	08
110. Explain the design issues of the network layer.	08*
111. What is congestion? Explain how congestion control is achieved in the following: i) Load shedding ii) Jitter control	08*
112. Explain the Dijkstra's algorithm to find shortest path between a given pair of nodes.	10
113. Explain any one adaptive and one non-adaptive routing algorithm.	10
114. Explain the working principle of Link State routing taking an example.	10
115. Discuss the principle of working of: (a) Weighted Fair Queuing (b) Hop-by-Hop Choke Packets	10
116. What do you mean by traffic shaping? Discuss various traffic shaping methods with respect to congestion control.	10
117. Explain the IP addressing scheme and formation of subnets.	10*
118. Explain the link state routing algorithm in detail.	10*
119. Discuss the Address Resolution Protocol (ARP) and Reverse ARP in the IP.	10
120. Mention the header format of Ipv6 giving details of all the fields.	10
121. Explain the following policies used in congestion control: (a) Load shedding (b) Jitter control (c) RSVP protocol	12
122. Explain Network Address Translation.	08
123. Discuss the BOOTP and DHCP protocols.	06
124. Clearly explain the Border Gateway Protocol.	08
125. What is internet multicasting?	03
126. What is the significance of extension headers in IPv6?	05
127. A computer on 6 Mbps network is regulated by a token bucket, which is filled at the rate of 1 Mbps. It is initially filled to capacity with 8 megabits. How long can the computer transmit at full 6 Mbps.	04
128. A class B network on the internet has a subnet mask of 255.255.240.0. What is the maximum number of hosts per subnet?	05
129. A computer network has 6 nodes A, B, C, D, E & F, located at the successive nodes of a	10

hexagon. The physical links and their associated costs are as given below:

Link	Cost
AB	10
BC	25
CD	20
DE	15
EF	25
FA	05
BE	15
FC	20
FD	30

Determine the costs of the shortest paths from the source node 'A' to each and every other node. All links are full duplex.

- |  |   |
|--|---|
| 130. Explain different design issues that should be followed to design a layered software protocols. [June 2004] | 6 |
| 131. Explain hierarchical routing and flooding algorithms. [June 2004]   | 8 |
| 132. What are the different ways by which congestion can be controlled in virtual circuit subnets? [June 2004]   | 8 |
| 133. What is internetworking? Explain contented virtual circuits. [June 2004]                                    | 4 |
| 134. Explain different formats of IP addressing and formation of subnets.  | 8 |
| 135. Explain the IPV6 protocol format.   | 4 |
| 136. How are choke packets used for congestion control?  | 6 |
| 137. With a diagram explain leaky bucket algorithm.  | 8 |
| 138. Distinguish between address resolution protocol and reverse address resolution protocol.                    | 8 |
| 139. Briefly explain the working of border gateway protocol.   | 6 |
| 140. Write short notes on broadcast routing.   | 5 |
| 141. Write short notes on Routing for mobile hosts.  | 5 |
| 142. Write short notes on tunneling.   | 5 |

## **THE TRANSPORT LAYER**

**OBJECTIVE:** The transport layer is the heart of whole protocol hierarchy. It provides reliable, cost-effective data transport from source machine to the destination machine, independent of the physical network or networks currently in use. After reading this chapter, you will know the following details provided by the transport layer

- Services
- Design
- Protocol
- Performance

- |  |     |
|--|-----|
| 143. Explain the functions of transport layer and the different transport service primitives in transport layer. | 10* |
| 144. Illustrate the logical relationship of the network, transport and application layers.                       | 05  |
| 145. Draw the state diagram for a simple connection management scheme.   | 05  |
| 146. Give the format of pseudo header included in the TCP checksum.  | 04  |
| 147. State Nagle's algorithm.  | 04  |
| 148. State Jacobson's algorithm.   | 04  |
| 149. What is Karn's algorithm?   | 05  |
| 150. Write short notes on Berkeley Sockets.  | 05  |
| 151. Explain what Clarke's silly window syndrome is and how it affects the performance of TCP.                   | 05  |
| 152. Illustrate the relationship between the NSAP, TSAP and transport connection.                                | 04  |
| 153. Write short notes on UDP.   | 05  |
| 154. Briefly explain the multiplexing of transport connections.  | 05* |
| 155. Explain the terms: Port, Urgent data, MTU, UDP and bandwidth-delay product.                                 | 05* |
| 156. Write explanatory notes on QoS and Option Negotiation.  | 05* |
| 157. What are transport service primitives?  | 06  |
| 158. What are chained fixed-size and chained variable-size buffers? How they are useful in flow control?         | 06  |
| 159. Explain how congestion is controlled in TCP.  | 06  |
| 160. Discuss four protocol scenarios for releasing a connection.   | 08  |
| 161. Explain transport services provided by the transport protocol.  | 08  |
| 162. Discuss various scenarios for establishing a connection using a three-way handshake.                        | 08  |
| 163. Explain the TCP Segment Header.   | 08  |

164. Explain the TCP and UDP headers.	08*
165. Explain the TCP protocol.	10
166. Write short notes on TCP congestion control.	06
167. Explain upward and downward multiplexing.	10*
168. Write short notes on Crash recovery.	08
169. Clearly explain the various elements of transport protocols.	12
170. Explain the "Connection Management" aspect of the transport layer. Discuss the concept of timer based connection management.	12*
171. Briefly explain the Real-time Transport Protocol.	10
172. Explain about: (i) TCP retransmission timer (ii) Persistence timer (iii) Keepalive timer	03 each
173. What are TTCP and SCTP?	06
174. Write short notes on Wireless TCP and UDP.	08
175. How network performance can be measured? What are its pitfalls?	06
176. List the system design rules for better network performance.	07
177. Compare TCP and IP with respect to their headers.	08
178. In a network that has a maximum TPDU size of 128 bytes, a maximum TPDU lifetime of 30 sec, and an 8-bit sequence number, what is the maximum data rate per connection?	04
179. Explain connection establishment and connection release protocol scenarios using a three way handshake of the transport layer	8
180. With an example explain TCP transmission policy	5
181. What is a socket? List and explain the socket primitives of TCP	10
182. Explain UDP with its frame format	6
183. Describe : a) Multiplexing b) Flow control and buffering in the transport layer	10

## **ATM NETWORKS**

**OBJECTIVE:** The concept of **ATM** (*Asynchronous Transfer Mode*) networks emerged from standardization activities directed at the development of *Integrated Services Digital Networks (ISDNs)*. ATM was developed to combine the attributes of time-division circuit-switched networks and packet-switched networks. ATM also provides the capability of providing Quality-of-Service support in a connection-oriented packet network. At the end of this chapter, you would know

- BISDN reference model
- ATM layer and ATM adaptation layer
- ATM signaling
- PNNI routing
- Classical IP over ATM

184. Explain the motivation for the development of ATM networks.	05
185. Briefly explain the Broadband ISDN model.	08
186. Draw a neat diagram of the ATM physical layer.	04
187. Describe the ATM cell header format.	06
188. What are the two types of virtual connections provided by ATM networks?	04
189. Clearly explain the QoS Parameters in ATM networks.	06
190. Explain what Traffic descriptors are with reference to ATM networks.	05
191. Explain the following terms: (i) Constant bit rate (ii) Real-time variable bit rate (iii) Non real-time variable bit rate (iv) Available bit rate (v) Unspecified bit rate	05
192. Explain the network functions of CAC and UPC w.r.t. Traffic Management.	04
193. Explain various Service categories used in ATM networks.	05
194. Explain the following processes with diagrams: (i) AAL1 process (ii) AAL2 process (iii) AAL3/4 process (iv) AAL5 process	04 each
195. Give the CPS packet structure.	08
196. Draw a sketch of AAL1 and AAL5 PDUs.	06
197. Explain SAAL process.	05
198. Draw the ATM address formats.	08
199. Write short notes on Service Specific Connection-Oriented Protocol (SSCOP).	05
200. Briefly describe UNI signaling with an example.	06
201. Explain PNNI signaling with an example.	08
202. Explain GCAC algorithm with its parameter requirements.	06
203. Give an example of PNNI hierarchy.	05
204. Explain the CLIP model.	06
205. Explain network layer in ATM networks.	08
206. Explain the structure of ATM adaptation layer and explain the AAL-1 cell format.	07
207. Write a note on ATM adaptation layer.	04

Marks	No. of Questions
3	15
4	41
5	38
6	34
7	04
8	43
10	27
12	05
<b>Total</b>	<b>207</b>

## NOTES

# CS65: SYSTEM SIMULATION & MODELING

Faculty:

No of Hours: 52

Class #	Chapter Title/ Reference Literature	Topics to be covered	% of portions covered	
			Reference chapter	Cumulative
1.	<b>Chapter #:1</b> Introduction to Simulation T: Page# 3 – 20	When Simulation is the Appropriate Tool; When Simulation is not Appropriate. Advantages And Disadvantages of Simulation;	7.6%	7.6%
2.		Areas of Application; System and System Environment		
3.		Components of a System; Discrete and Continuous System		
4.		Model of a system ; Types of Models; Discrete-event System simulation; Steps in a Simulation Study		
5.	<b>Chapter # :2</b> Simulation Example T: Page# 24 – 47	Characteristics of Queuing Systems:	11.53%	19.13%
6.		Queuing Notation.		
7.		Simulation Of queuing Systems.		
8.		Simulation Of queuing Systems.		
9.		Simulation Of Inventory Systems.		
10.	Simulation Of Inventory Systems.			
11.	<b>Chapter # :3</b> General Principles T: Page# 64 – 85	Concepts in Discrete-Event Simulation	7.6%	26.73%
12.		The Event-Scheduling/Time Advance Algorithm		
13.		World Views		
14.		Manual Simulation Using Event Scheduling.		
15.	<b>Chapter # :4</b> Random-Number Generation T: Page# 255 -284	Properties of Random Numbers.	11.53%	38.26%
16.		Generation of Pseudo-Random Numbers.		
17.		Generation of Pseudo-Random Numbers.		
18.		Techniques for generating Random Numbers		
19.		Tests for Random Numbers.		
20.	Numerical problem solving			
21.	<b>Chapter # :5</b> Random-Variate Generation T: Page# 290 –295, 301 – 307, 310 – 314	Inverse Transform Technique: Exponential Distribution.	7.6%	45.86%
22.		Uniform Distribution.		
23.		Discrete Distributions:		
24.		Acceptance - Rejection Technique; Poisson Distribution		
25.	<b>Chapter # :6</b> Input Modeling T: Page# 324 – 358	Data Collection	13.46%	59.32%
26.		Identifying the Distribution with data.		
27.		Parameter Estimation		
28.		Goodness of Fit Tests		
29.		Goodness of Fit Tests		
30.		Selecting Input		
31.	Multivariate and time series input models.			
32.	<b>Chapter # :7</b> Verification and Validation of Simulation Models T: Page# 368 – 393	Model Building	9.6%	68.92%
33.		Verification and Validation.		
34.		Verification of Simulation Models;		
35.		Calibration and Validation of Models.		
36.	Calibration and Validation of Models.			
37.	<b>Chapter # :8</b> Output Analysis for A Single Model T: Page#399 – 441	Types of Simulations With Respect to Output Analysis	15.3%	84.22%
38.		Stochastic Nature of Output Data		
39.		Measure Of Performance and their Estimation.		
40.		Measure Of Performance and their Estimation.		
41.		Output Analysis for Terminating Simulations.		
42.		Output Analysis for Terminating Simulations.		
43.		Output Analysis for Steady-stat Simulations.		
44.		Output Analysis for Steady-stat Simulations.		
45.	<b>Chapter # :9</b> Simulation Of Computer Systems T: Page#528 - 566	Introduction	15.3%	100%
46.		Simulation Tools		
47.		Model input		
48.		High-Level Computer System Simulation		
49.		CPU Simulation		
50.		CPU Simulation		
51.		Memory Simulation		
52.	Memory Simulation			

## Literature

Book Type	Code	Title & Author	Publication Info		
			Edition	Publisher	Year
Text Book	T	Discrete- Event System Simulation- Jerry Banks, John S. Carson, Barry L. Nelson, David M. Nicol	3 <sup>rd</sup>	PE/PHI	
Reference Book	R	Simulation Modeling and Analysis by Averill M. Law, W. David Kelton	3 <sup>rd</sup>	McGrawHill	
Reference Book	R	System Simulation by Geoffery Gordon	2 <sup>nd</sup>	PHI	

## QUESTION BANK

### CHAPTER 1: INTRODUCTION, REQUIREMENTS ENGINEERING

**OBJECTIVE:** The objective of this topic is to understand the

- When Simulation is the appropriate tool and not appropriate.
- Advantages And Disadvantages of Simulation
- Areas of Application; System and System Environment
- Components of a System.
- Discrete and Continuous System.
- Model of a system; Types of Models.
- Discrete-event System simulation; Steps in a Simulation Study.

1.	When Simulation is the appropriate tool?	05
2.	When Simulation is not appropriate tool?	05
3.	What are advantages And Disadvantages of Simulation?	08*
4.	How can we offset the disadvantages of simulation?	05
5.	List the application areas/Industry domains of simulation?	05
6.	List 5 typical applications each in manufacturing and transportation systems?	05
7.	List 5 typical applications each in business process simulation & logistics, supply chain and distribution?	05
8.	What is System and System Environment?	05
9.	Explain the terms: (a) entity (b) attribute (c) activity (d) event & (e) state in the system simulation context?	10
10.	Explain and give an example each of continuous and discrete system?	05*
11.	What is Model and Component of the system?	05
12.	Explain Discrete-event System simulation and Steps in a Simulation Study.	12*
13.	Name several entities, attributes, activities, events & state variables of a typical automatic teller machine (ATM)?	12

### CHAPTER 2: SIMULATION EXAMPLE

**OBJECTIVE:** The objective of this topic is to understand the

- Characteristics of Queuing Systems.
- Queuing Notation.
- Simulation of queuing Systems.
- Simulation of Inventory Systems.

14.	Explain the queuing system in simulation.	10
15.	Explain the following queuing system characteristics: (a) calling population (b ) system capacity (c) Arrival process (d) Queue behavior and discipline (e) service time and service mechanism	10*
16.	Describe Kendal-Lee notation for a queuing system	08
17.	Explain the Inventory System in simulation.	10
18.	Explain with suitable examples : (a) Inter-arrival time (b) Service time (c) Utility time (d) Idle time of a queuing system	10*
19.	With a suitable flow chart describe two server queue system	10
20.	A problem on reliability.	12
21.	A problem on News Paper Sellers.	12
22.	A problem on Simulation of a (M,N) inventory system.	12
23.	A problem on Single-Channel Queue.	12
24.	A problem on Able Bakers carhop.	12
25.	A problem on Random normal numbers.	12
26.	A problem on Lead Time demand.	12

### CHAPTER 3: GENERAL PRINCIPLES

**OBJECTIVE:** The objective of this topic is to understand the

- Concepts in Discrete-Event Simulation
- The Event-Scheduling/Time Advance Algorithm
- World Views
- Manual Simulation Using Event Scheduling.

27.	Explain the concept of Discrete-Event Simulation.	12*
-----	---	-----

28.	Explain in detail the event scheduling/time advance algorithm	08
29.	Describe with examples the various world views	10
30.	Prepare a simulation table for a single channel queue system until the clock reaches time 20. The stopping event will be at time 30. Inter-arrival times 4 5 2 8 3 6 Service times 3 5 4 6 1 5	10
31.	Explain manual simulation using event scheduling with the help of a suitable example.	10
32.	Provide the detailed flow chart of a typical arrival event and a departure event in a single channel queuing system	12
33.	What is list processing? Explain.	12

## **CHAPTER 4: RANDOM-NUMBER GENERATION**

**OBJECTIVE:** The objective of this topic is to understand the

•	Properties of Random Numbers.	
•	Generation of Pseudo-random Numbers	
•	Techniques For Generating random Numbers	
•	Tests for random Numbers.	
34.	Explain the properties of random number & its consequences.	10
35.	Explain the generation of Pseudo-random Numbers.	10
36.	Explain the linear congruential method for random number generation?	10*
37.	Explain the combined linear congruential random number generation method?	10
38.	What is the role of maximum density and maximum period in random number generation?	08
39.	Generate a sequence of 15 random numbers for which seed is 342, constant multiplier is 20, increment is 45 and modulus is 30	12*
40.	Explain with an example the Kolmogorov-Smirnov test for random numbers	08
41.	Explain with an example the chi-square test for random numbers?	08
42.	Explain auto correlation Test for random numbers.	08
43.	Using the principles learnt, develop your own combined linear congruential random number generator	15

## **CHAPTER 5: RANDOM-VARIATE GENERATION**

**OBJECTIVE:** The objective of this topic is to understand the

•	Inverse Transform Technique: Exponential Distribution	
•	Uniform Distribution	
•	Discrete Distributions	
•	Acceptance-rejection Technique; Poisson Distribution	
44.	What is inverse transform technique? Explain how it is used for producing random variants for exponential distribution and uniform distribution.	10*
45.	Explain Exponential Distribution.	08
46.	Briefly describe Uniform Distribution.	12
47.	With example explain the various types of discrete distributions.	08
48.	What are all the different acceptance rejection techniques?	10
49.	What is convolution method? Explain.	10

## **CHAPTER 6: INPUT MODELING**

**OBJECTIVE:** The objective of this topic is to understand the

•	Data Collection	
•	Identifying the Distribution with data.	
•	Parameter Estimation	
•	Goodness of Fit Tests	
•	Selecting Input	
•	Multivariate and time series input models.	
50.	State the four steps involved in the development of an input model?	08*
51.	Explain data collection with example.	08
52.	Explain identifying the distribution with data with example.	12
53.	Explain parameter estimation with examples	10*
54.	Explain goodness of fit tests with examples	10*
55.	How can you select input model with out data? Explain with example	10

56.	Define co variance & correlation?	05
57.	Explain AR-1 model?	08
58.	Explain EAR1 model?	08

## **CHAPTER 7: VERIFICATION AND VALIDATION OF SIMULATION MODELS**

**OBJECTIVE:** The objective of this topic is to understand the

- Model Building
- Verification and Validation.
- Verification of Simulation Models
- Calibration and Validation of Models

59.	How model can be build verification and validate? Explain with diagram.	10*
60.	What are the techniques for verification of simulation model?	08
61.	Describe in detail the three step approach for model validation?	5*
62.	What is model reasonable ness & explain how current contents and total count can verify it?	08
63.	Briefly explain the validation of input-output transformations of the model and the various techniques used?	10

## **CHAPTER 8: OUTPUT ANALYSIS FOR A SINGLE MODEL**

**OBJECTIVE:** The objective of this topic is to understand the

- Types of Simulations With Respect to Output Analysis
- Stochastic Nature of Output Data
- Measure Of Performance and their Estimation
- Output Analysis for Terminating Simulations.
- Output Analysis for Steady-stat Simulations.

64.	What are the types of simulations with respect to output analysis?	10
65.	Explain stochastic nature of output data with example.	12
66.	Explain measure of performance and their estimation.	10
67.	Explain output analysis of terminating simulations with examples.	12
68.	With illustrative examples explain output analysis of steady-state simulations	10*
69.	Explain how probabilities and quantiles can be estimated from summary data?	08
70.	Describe initialization bias in steady state simulation	08
71.	Explain batch means for interval estimation in steady state simulation	08

## **CHAPTER 9: SIMULATION OF COMPUTER SYSTEMS**

**OBJECTIVE:** The objective of this topic is to understand the

- Introduction
- Simulation Tools
- Model input
- High-Level Computer System Simulation
- CPU Simulation
- Memory Simulation

72.	Which are all the simulation tools present?	10
73.	What are the different labels of different computer systems?	08
74.	Explain the run time stack in threaded simulation.	12
75.	Explain model input.	08
76.	Briefly describe high-level computer system simulation. With example.	10
77.	Explain memory and CPU simulation.	15*

Marks	No of Questions
05	11
08	20
10	26
12	18
15	02
<b>Total</b>	<b>77</b>

# CS661: PRINCIPLES OF PROGRAMMING LANGUAGES

Faculty:

No of Hours: 52

Class #	Chapter Title / Reference Literature	Topic to be covered	% of portions covered	
			Reference Chapter	Cumulative
1.	<b>Chapter #: 1.0</b> Introduction T1: page#: 1-21, 28-33	Towards higher-level languages	10 %	10 %
2.		Programming paradigms		
3.		Language Implementation: Bridging the gap		
4.		Expression notations		
5.		Abstract syntax trees		
6.	<b>Chapter #: 2.0</b> Types: Data Representation T2: page#: 150-182, 201-226, 249-252	Elementary data types – Data objects, Variables and constants	17 %	27 %
7.		Data types, Declarations		
8.		Type checking, Type conversion		
9.		Numeric data types, Enumerations		
10.		Booleans, Characters		
11.		Structured data types – structured data objects		
12.		Specification of data structure types, Implementation of data structure types		
13.		Vectors and Arrays		
14.		Records, Type equivalence		
15.		<b>Chapter #: 3.0</b> Imperative Programming T2: page#: 308-323, 347-353 T1: page#: 77-90, 155-190		
16.	Structured sequence control			
17.	Handling special cases in loops			
18.	Programming with invariants			
19.	Proof rules for partial correctness			
20.	Procedure activations			
21.	Simple call – return subprograms			
22.	Parameter passing methods			
23.	Scope rules for names			
24.	Nested scopes in the source text, Activation records			
25.	Lexical scope: Procedure as in C			
26.	<b>Chapter #: 4.0</b> Object-Oriented Programming T1: page#: 220-228, 253-274, 281-284	Program design with modules	13 %	61 %
27.		What is an object?		
28.		Object oriented thinking		
29.		Inheritance		
30.		Object oriented programming in C++		
31.		Derived classes		
32.		Information hiding		
33.	<b>Chapter #: 5.0</b> Functional Programming T1: page#: 305-313, 318-335, 341-369, 385-404	Elements of functional programming – A little language of expressions	25%	86 %
34.		Types: values and operations		
35.		Approaches to expression evaluation		
36.		Lexical scope, Type checking		
37.		Functional programming in a typed language – Exploring a list		
38.		Function declaration by cases		
39.		Function as first-class values		
40.		ML: Implicit types, Data types		
41.		Exception handling in ML		
42.		Functional programming with lists – Scheme		
43.		A dialect of lisp		
44.		The structure of lists		
45.	List manipulation			
46.	<b>Chapter #: 6.0</b> <b>Logic Programming</b> T1: page#: 425-450,	Computing with relations	14 %	100%
47.		Introduction to Prolog		
48.		Introduction continued..		
49.		Data structures in prolog		
50.		Data structures in prolog continued..		
51.		Data structures in prolog continued..		
52.		Programming techniques, control in prolog		

**Literature**

Book Type	Code	Title & Author	Publication Info		
			Edition	Publisher	Year
Text Books	T1	Ravi Sethi, Programming languages	2 <sup>nd</sup>	Pearson Education	--
	T2	Terrence W. Pratt, Programming languages Design and Implementation	4 <sup>th</sup>	Pearson Education	--
Reference Books	R1	Allen Tucker, Robert Nonan, Programming languages	--	Tata McGraw-Hill	2002
	R2	Doris Appleby, Julius J. Vandekopple, Programming languages paradigm and practice	2 <sup>nd</sup>	Tata McGraw-Hill	1998

# CS662: DECISION SUPPORT SYSTEMS

Faculty:

No of Hours: 52

Class #	Chapter Title/ Reference Literature	Topics to be covered	% of portions covered	
			Reference chapter	Cumulative
1	<b>Chapter #:1</b> Decision making and computerized support T1 : Chapter 1.2 – 1.11, 1.14, 1.15, 2.2-2.10	Managers and Decision making	24%	24%
2		Managerial decision making and information systems		
3		Managers and computerized support		
4		Frame work and concept of DSS		
5		Group Decision systems, Executive information systems		
6		Expert systems, Artificial neural networks, Hybrid support systems		
7		Evolution and attributes of computerized decision aids, Introduction and definitions		
8		Systems, models and modeling process		
9		Decision making – Intelligent phase		
10		Decision making – Design phase		
11		Decision making – Choice phase		
12		Decision making – Implementation phase		
13	<b>Chapter #:2</b> Decision Support Systems I T1: Chapter 3.2 – 3.13	DSS configuration, characteristics	12%	36%
14		capabilities and components		
15		Data Management sub system		
16		Model Management sub system		
17		Knowledge based management system		
18		User, user interface, hardware Distinguishing DSS from Management science and MIS DSS classification		
19	<b>Chapter #:3</b> Decision Support Systems II T1: Chapter 6.2 – 6.12, 7.2 - 7.11	Introduction to DSS development ,	24%	60%
20		Traditional System development life cycle		
21		Alternate development methodologies		
22		Prototyping: DSS development methodology		
23		DSS technology levels and tools,		
24		Development platforms, development tool selection		
25		Team developed DSS, End-User developed DSS		
26		Developing DSS: Putting the system together		
27		Developing DSS: Putting the system together		
28		DSS research directions and future		
29		Group decision making, Communication support		
30		Collaboration support Group Support Systems and technologies Distance learning, creativity and idea generation GSS and collaborative computing issues and research		
31	<b>Chapter #:4</b> Expert Systems T1 : Chapter 10.1 – 10.9, 10.11 – 10.16, 11.2 – 11.5, 11.12 – 11.19	Knowledge based DSS, Concepts and definitions of AI,	16%	76%
32		AI vs. natural intelligence, The artificial intelligence field		
33		Types of knowledge based DSS, The human element in Expert Systems		
34		How Expert systems work? Problem areas addressed, benefits		
35		Problems and limitations of ES		
36		Types of ES, ES and the internet/intranet web		

37		Knowledge engineering, Scope of knowledge		
38		Difficulties in knowledge acquisition, methods of knowledge acquisition		
39		Machine learning, intelligent agents		
40		selecting an appropriate knowledge acquisition method		
41		Knowledge acquisition from multiple experts		
42		Validation and verification of knowledge base.		
		Analyzing, coding, documenting and diagramming		
		Numeric and documented knowledge acquisition,		
		Knowledge acquisition and Internet/Intranet		
		Induction and table example.		
43	<b>Chapter #:5</b> Expert Systems II T1: Chapter 13.2 – 13.3, 13.5 – 13.13, 14.2 – 14.4, 14.10 – 14.14	Reasoning in AI, Inference with rules, Inference with frames	24%	100%
44		Model based reasoning, Case based reasoning		
45		Explanation and Meta knowledge		
46		Inference with uncertainty, Representing uncertainty		
47		Probability and related approaches		
48		Theory of certainty		
49		ES development life cycle – Phase I: Project initialization		
50		ES development life cycle – Phase II: System analysis and design		
51		ES development life cycle – Phase III: Prototyping		
52		ES development life cycle – Phase IV: System development		
		ES development life cycle – Phase V: Implementation		
		ES development life cycle – Phase VI: Post Implementation		
		Future of ES development process		

#### Literature

Book type	Code	Title and Author	Publication specification		
			Edition	Publication	Year
Text Book	T1	Decision Support Systems and Intelligent Systems by Efraim Turban & Jay E. Aronson	6 <sup>th</sup>	Pearson Education Asia/PHI	2001
Reference Book	R1	Expert Systems: Principles and Programming by Giarratano & Riley		Thomson Brooks/ Cole	2002
Reference Book	R2	Decision Support Systems by Sprague R.H. Jr and H.J. Watson	4th	Prentice Hall	1996

## CSL67: VISUAL PROGRAMMING AND DATABASE APPLICATIONS LABORATORY

### Course Objective:

Visual Programming & database applications laboratory is mainly focused on real time requirements using backend level and effective customization at the front end level. High end programs in DBMS basically involves effective database design at the backend level using any RDBMS and effective customer interface at the front-end level using GUI tool like Visual Basic.

Week #	Experiments Covered
1 & 2	Program 1: Analysis and back-end coding of Insurance database
	Program 1: Analysis and front-end coding of Insurance database
3 & 4	Program 2: Analysis and back-end coding of Order processing database
	Program 2: Analysis and back-end coding of Order processing database
5 & 6	Program 3: Analysis and back-end coding of Student database
	Program 3: Analysis and back-end coding of Student database
8 & 9	Program 4: Analysis and back-end coding of book dealer database
	Program 4: Analysis and back-end coding of book dealer database
10 & 11	Program 5: Analysis and back-end coding of Banking database
	Program 5: Analysis and back-end coding of Banking database
12	Test final evaluation and Viva

### QUESTION BANK

SI #	LIST OF PROGRAMS
1.	<p>Given that an INSURANCE database contains following tables: PERSON, CAR, ACCIDENT, OWNS, PARTICIPATED</p> <p><b>Objectives:</b> Create the above tables by properly specifying the primary keys and the foreign keys Enter at least five tuples for each relation</p> <p>Demonstrate how you</p> <ol style="list-style-type: none"> <li>Update the damage amount for the car with a specific RegNo in the accident with report number 12 to 25000</li> <li>Add a new accident to the database</li> </ol> <p>Find the total number of people who owned cars that were involved in accidents in 2002 Find the number of accidents in which cars belonging to a specific model were involved Generate suitable reports Create suitable front end for querying and displaying the results.</p>
2.	<p>Given that an ORDER PROCESSING database contains following tables: CUSTOMER, ORDER, ORDER- ITEM, ITEM, SHIPMENT, WAREHOUSE</p> <p><b>Objectives:</b> Create the above tables by properly specifying the primary keys and the foreign keys Enter at least five tuples for each relation Produce a listing : CUSTNAME, #of Orders, AVG_ORDER_AMT where the middle column is the total number of orders by the customer and the last column is the average order amount for that customer</p> <p>List the order nos for orders that were shipped from all the warehouses that the company has in a specific city</p> <p>Demonstrate how you delete item#10 from the ITEM table and make that field NULL in the order, item table</p>

	<p>Find the total number of people who owned cars that were involved in accidents in 2002</p> <p>Find the number of accidents in which cars belonging to a specific model were involved</p> <p>Generate suitable reports</p> <p>Create suitable front end for querying and displaying the results.</p>
3.	<p>Given that an INSURANCE database contains following tables: STUDENT, COURSE, ENROLL, BOOK_ADOPTION, TEXT</p> <p><b>Objectives:</b></p> <p>Create the above tables by properly specifying the primary keys and the foreign keys</p> <p>Enter atleast five tuples for each relation</p> <p>Demonstrate how you add a new textbook to the database</p> <p>Produce a list of text books in the alphabetical order for courses offered by the CS department that use more than two books.</p> <p>List any department that has all its adopted books published by a specific publisher</p> <p>Generate suitable reports</p> <p>Create suitable front end for querying and displaying the results.</p>
4.	<p>Given that an BOOK database contains following tables: AUTHOR, PUBLISHER, CATALOG, CATEGORY, ORDER-DETAILS</p> <p><b>Objectives:</b></p> <p>Create the above tables by properly specifying the primary keys and the foreign keys</p> <p>Enter atleast five tuples for each relation</p> <p>Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication after 2000.</p> <p>Demonstrate how you increase the price of the books published by a specific publisher by 10%</p> <p>Find the author of the book which has maximum sales</p> <p>Find the number of accidents in which cars belonging to a specific model were involved</p> <p>Generate suitable reports</p> <p>Create suitable front end for querying and displaying the results.</p>
5.	<p>Given that an BANKING database contains following tables: BRANCH, ACCOUNT, DEPOSITOR, CUSTOMER, LOAN, BORROWER</p> <p><b>Objectives:</b></p> <p>Create the above tables by properly specifying the primary keys and the foreign keys</p> <p>Enter at least five tuples for each relation</p> <p>Find all the customers who have atleast two accounts at the Main branch</p> <p>Find the total customers who have an account at all the branches located in a specific city</p> <p>Demonstrate how you delete all account tuples at every branch located in a specific city</p> <p>Generate suitable reports</p> <p>Create suitable front end for querying and displaying the results.</p>

## CSL68: GRAPHICS LABORATORY

Week #	Experiments covered
1	Program 1
2	Program 2
3	Program 3
4	Program 4
5	Program 5
6	Program 6
7	Program 7
8	Program 8
9	Program 9
10	Program 10
11	2D Graphics Editor
12	2D Graphics Editor
13	2D Graphics Editor
14	2D Graphics Editor

### QUESTION BANK

SI #	Program
1.	Write a program to implement the following using built-in functions available in C / C++ Graphics Library. a) Kite flying                      b) Bar Graph                      c) Spiral model d) Fish movement                  e) Rotation of a wheel              f) Moving car
2.	Write a program to implement the midpoint line algorithm to generate a line with any slope and thickness. Implement the poly line command using this algorithm as a routine that displays the set of straight lines between the 'n' input points. For n=1 the routine displays a single point.
3.	Write a program to display a line graph using midpoint line algorithm. The input to the program is set of data corresponding to the X and Y axes. Data parts to be displayed as asterisk (*) joined according to the input specification.
4.	Write a program to implement midpoint circle drawing algorithm. Modify the algorithm to implement specified arc or sector.
5.	Write a program to draw a rectangle using built in line function & perform the following operations. i)        Rotation about origin followed by translation. ii)       Rotation about arbitrary point. iii)      Apply X shear and Y shear on the rectangle.
6.	Write a program to create a house like figure and perform i)        Scaling with reference to origin followed by translation ii)       Scaling with reference to an arbitrary point. iii)      Reflect about the line $y = mx + c$
7.	Write a program to implement the Cohen-Sutherland line clipping algorithm. Make provision to specify the input line, window for clipping and view port for displaying the clipped image (use built in line and rectangle function).
8.	Write a program to implement Hermite and Bezier curve for given set of control points.
9.	Write a program to implement 3-D rotation with respect to X-axis, Y-axis and Z-axis (wire frame model of a cube). Use appropriate data structures to manipulate the wire frame model.
10.	Write a program to fill any given polygon using scan-line area filling algorithm. (use appropriate data structures).

### **GUIDELINES FOR 2D GRAPHICS EDITOR**

- Minimum Facilities: Cursor movement, Editing picture objects (point, line, circle, box, arc, ellipse, spiral, polygon), Transformations on objects/selected area, Clipping Filling, Load, Save, Clear, etc.
- Good User Interface (GUI/Iconic / Menu based with on-line Help)
- Shall use all most all the Graphical Input Techniques for efficient programming of input devices.
- Shall be device independent package.
- Shall be implemented in any platform using C /C++ language.

Note: It should be supported by a report.