

Bachelor of Technology (Computer Science & Engineering)
Scheme of Studies/Examination
Semester VII

S. No.	Course No.	Subject	L:T:P	Hours/Week	Examination Schedule				Duration of Exam (Hrs.)
					Major Test	Minor Test	Practical	Total	
1	CSE 401N	Unix & Linux Programming	4:0:0	4	75	25	0	100	3
2	CSE 403N	Computer Graphics and Animation	4:0:0	4	75	25	0	100	3
3	PE-I	Elective* – I	3:0:0	3	75	25	0	100	3
4	PE-II	Elective* – II	3:0:0	3	75	25	0	100	3
5	CSE 405N	Computer Graphics Lab	0:0:2	2	0	40	60	100	3
6	CSE 407N	Project-I**	0:0:9	9	0	100	100	200	3
7	CSE 409N	Unix & Linux Programming Lab	0:0:2	2	0	40	60	100	3
8	CSE 411N	Seminar	0:0:2	2	0	100	0	100	
9	CSE 413N	Industrial Training (Viva-Voce)***				100	0	100	
		Total		29	300	480	220	1000	

Code	PE-I	Code	PE-II
CSE-415N	Object Oriented Software Engineering	CSE-421N	Agile Software Engineering
CSE-417N	Cyber Security	CSE-423N	Big Data and Analytics
CSE-419N	Cryptography & Information Security	CSE-425N	Expert Systems

Note:

*The students will choose any two departmental electives courses out of the given elective list in 7thSemester.

**Project should be initiated in the beginning of 7thsemester, and should be completed by the end of 8thsemester with good Report and power-point Presentation etc.

***4-6 weeks hand on training completed after 6thSemester Exams.

CSE-401N	Unix & Linux Programming					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
4	0	0	75	25	100	3 Hrs.
Purpose	Introduces commands and numerous programming concepts and application domains to cover important topics for implementation of the Unix programming concepts.					
Course Outcomes (CO)						
CO1	fundamental understanding of open source software, basic architecture of UNIX.					
CO2	Develop skills to use UNIX tools like grep, sed, awketc for large scale text processing					
CO3	to develop shell scripts to solve and Complete various computing tasks.					
CO4	Understanding static and dynamic library management.					
CO5	To understand File systems and File structures.					
CO6	To understand basic linux administration					

Unit I: Basic Command Usage

Linux Startup: User accounts, accessing Linux - starting and shutting processes, Logging in and Logging out, Unix commands like zip, unzip, pack, unpack, compress, uncompress, Shell Programming, Unix file system: Linux/Unix files, i-nodes and structure, file system related commands, Shell as command processor, shell variables, creating command substitution, scripts, functions, conditionals, loops, customizing environment

Unit II: Filters and File Compression

Regular Expressions and Filters: Introducing regular expressions patterns, syntax, character classes, quantifiers, introduction to grep, egrep, sed, programming with awk and perl, File Compression Techniques: data redundancy elimination using fingerprint generation deduplication and data similarities removal using delta techniques for data reduction storage, parallel compression with Xdelta utility.

Unit III: Program Development Tools

The C Environment: C compiler, vi editor, compiler options, managing projects, memory management, use of makefile, cmake, dependency calculations, memory management – static and dynamic memory, static and dynamic libraries, dynamic loader, debugging tools like gdb, fixed-size and variable-size blocks of data files chunks divisor chunking techniques like Frequency Based Chunking and Content Defined Chunking Unix based open source coding.

Unit IV: Process Control

Processes in Linux: Processes, starting and stopping processes, initialization processes, rc and init files, job control - at, batch, cron, time, network files, security, privileges, authentication, password administration, archiving, Signals and signal handlers, Threading, Linux I/O system, Networking tools like ping, telnet, ftp, route, Firewalls, Backup and Restore tar, cpio, dd.

Case Study: PCOMPRESS open source free software

Text Books:

1. John Goerzen: Linux Programming Bible, IDG Books, New Delhi, 2014.
2. Sumitabha Das: Unix – Concept and Applications, Fourth Edition TMH, 2015.
3. Neil Matthew, Richard Stones: Beginning Linux Programming, 4th. Edition, Wrox-Shroff, 2011.
4. Welsh & Kaufmann: Running Linux, O'Reiley & Associates, 2013.

Reference Book:

1. B.M. Harwani, Unix and Shell Programming, Oxford University Press, 2013.

LESSON PLAN

Week	Unix/Linux	
	Lecture Day	Topic
1	1 st	User accounts
	2 nd	accessing Linux – starting and shutting processes
	3 rd	Command Line simple commands
	4 th	Logging in and Logging out
2	5 th	Unix file system
	6 th	I-nodes and structure and file system related commands
	7 th	Shell as command processor
	8 th	shell variables
3	9 th	scripts
	10 th	creating command substitution
	11 th	functions, conditionals,
	12 th	functions, conditionals,
4	13 th	loops, customizing environment
	14 th	loops, customizing environment
	15 th	introduction to egrep
	16 th	compiler options
5	17 th	Introducing regular expressions patterns
	18 th	syntax, character classes
	19 th	quantifiers
	20 th	sed
6	21 st	programming with awk
	22 nd	Perl
	23 rd	programming with Perl
	24 th	The C compiler
7	25 th	vi editor
	26 th	managing projects
	27 th	memory management,
	28 th	use of makefiles
8	29 th	memory management – dynamic and static memory
	30 th	building and using static and dynamic libraries
	31 st	using ldd
	32 nd	debugging with gdb
9	33 rd	File Compression Techniques
	34 th	data redundancy elimination using fingerprint generation

	35 th	data redundancy elimination using fingerprint generation
	36 th	deduplication and data similarities parallel compression
10	37 th	removal using delta techniques for data reduction storage
	38 th	parallel compression with Xdelta utility
	39 th	dependency calculations
	40 th	soname
11	41 st	dynamic loader
	42 nd	Linux/Unix files
	43 rd	fixed-size and variable-size blocks of data files
	44 th	chunks divisor chunking techniques like Frequency Based Chunking
12	45 th	Content Defined Chunking Unix based open source coding
	46 th	Processes, starting and stopping processes
	47 th	Processes, starting and stopping processes
	48 th	initialization processes, re and init files,
13	49 th	job control – at, batch, cron, time
	50 th	job control – at, batch, cron, time
	51 st	network files, security
	52 nd	network files, security
14	53 rd	privileges, authentication
	54 th	password administration
	55 th	archiving, Signals and signal handlers,
	56 th	Linux I/O system
15	57 th	Networking tools like ping, telnet, ftp, route, Firewalls
	58 th	Networking tools like ping, telnet, ftp, route, Firewalls
	59 th	Backup and Restore tar, cpio, dd
	60 th	Case Study: PCOMPRESS open source free software

TUTORIAL SHEET-1

1. Explain Unix system organization.
2. How are the Linux directory structures organized?
3. Explain the commands that are used for arithmetic operations.
4. What is a shell script?
5. Differentiate between shell scripts and functions.

TUTORIAL SHEET-2

1. Write a note on conditional parameter substitution in bash shell.
2. Explain shell as command processor.
3. How to give the permission access to the files?
4. Write a shell script to merge 2 files and display result in 3rd file.

TUTORIAL SHEET-3

1. What are the editing features of VI editor?
2. How dynamic and static memory management is done in the C environment?
3. What are the functions of dynamic loader? Explain.
4. What are the debugging features of gdb?

TUTORIAL SHEET-4

1. Explain job control features of Linux.
2. How a process can be suspended in Linux?
3. What are the various security privileges of Linux? Explain.
4. What is network file security?
5. Explain Linux I/O system.

Roll No.

Total Pages : 03

BT-7/D-19

37006

UNIX AND LINUX PROGRAMMING

CSE-471

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) What are the different blocks the constitute a Unix File System ? **8**
- (b) Explain the procedure of mounting and unmounting a file system in a Unix Operating System. **7**
2. (a) What is the significance of i-nodes in Unix/Linux ? **8**
- (b) Explain the following commands with their syntax and examples :
 - (i) ls
 - (ii) rmdir
 - (iii) who
 - (iv) cp

- (v) touch
- (vi) cmp
- (vii) finger.

7

Unit II

3. (a) How is the expression g^* different from gg^* ?
What does $grep \ ^*$ look for ? 8
- (b) Briefly explain `egrep` and `sed` with their main roles. 7
4. (a) Develop a control-break AWK program that reads `empn.lst` and prints a report that groups employees of the same department. For each department, the report should print :
- (i) The department name at the top.
 - (ii) The remaining details of every person in the department.
 - (iii) Total salary bill for the department. 8
- (b) Write a PERL based program to convert to binary number specified as argument to decimal. 7

Unit III

5. (a) Name the *three* phases that a program has to go through before an executable is created from it. Also illustrate `cc` command to create an executable program. 8

- (b) A make rule does not always have a dependency and the target need not be a disk file. Explain with an example of a Makefile utility. 7
6. (a) How is a C program compiled assembled to create an object file and linked to create that executable file ? Explain with suitable example. 8
- (b) How are the projects developed and managed in c environment of Unix based platform ? 7

Unit IV

7. (a) Why do the address space of a process is virtual in Unix ? If two users execute the same program, are the memory requirements doubled. Justify with comments. 8
- (b) What is Kernel Mode ? Does a separate process run in this model ? Give the comments. 7
8. (a) How does a process voluntarily relinquish control of the CPU ? Discuss in detail. 8
- (b) How does a process behave when it receives a signal ? What is special about the SIGSTOP and SIGKILL signals ? 7

Roll No.

Total Pages : 02

BT-7/D-19

37147

UNIX AND LINUX PROGRAMMING

CSE-401

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. How is shell used as command processor ? Briefly tell about shell scripting. Briefly explain about i-nodes and its structure. **15**
2. What are shell variables ? Briefly tell about shell programming. What is threading in Linux ? **15**

Unit II

3. How is programming done with awk and perl in unix ? Illustrate with suitable examples. **15**
4. How is parallel compression performed with Xdelta utility ? **15**

Unit III

5. What do you mean by static and dynamic libraries ? How is dynamic memory management performed in Linux ?
15
6. How is boundary-shifting problem of fixed-size chunking resolved by content-defined chunking ? What is frequency based chunking ?
15

Unit IV

7. Explain the various components of PCOMPRESS utility in detail.
15
8. What is Linux I/O system ? Discuss the various networking tools of Linux OS like telnet, ftp and firewalls.
15

UNIX AND LINUX PROGRAMMING
CSE-471

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) What is Unix Command ? Write also different types of Unix commands in detail. 8
- (b) What is a Unix file system ? Briefly describe file system related commands. 7
2. (a) Describe i-nodes and its structure for linux files system. 8
- (b) What do you mean by user accounts in linux ? How are user accounts created in linux ?

Unit II

3. (a) Explain the text editing features of sed using suitable examples. 8
- (b) Explain various grep options to display a count, line number and filenames. 7
4. (a) Explain the main strength of PERL language. "How is it better than AWK language ? 8
- (b) Find the occurrences of three consecutive and identical word characters like ccc or ddd using egrep. 7

Unit III

5. (a) How is memory managed in multi-user environment efficiently by Linux ? 8
- (b) What is dynamic loader in Unix programming ? How it works in Unix ? 7
6. (a) How is a project managed by using Unix programming platform ? Explain. 8
- (b) Write short note on soname and Idd. 7

Unit IV

7. What is the different between a job and a process ? How do you :
- (a) suspend the foreground job
- (b) move a suspended job to background
- (c) bring back a suspended job to the foreground ? 15
8. (a) How is password provide the security to Linux system administrator ? What is system authentication ? 8
- (b) What are various job control facilities available in Linux for manipulating jobs ? 7

Roll No.

Total Pages : 02

BT-7/D-18

37147

UNIX AND LINUX PROG
CSE-401-N

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) Explain with example :
Zip and Unzip, Pack and Unpack, Compress and Uncompress.
- (b) Explain starting and shutting of process.
2. Explain file oriented commands in detail with example.

Unit II

3. Explain redundancy elimination technique.
4. Explain regular expression pattern command with syntax.

(2-70/4) L-37147

P.T.O.

Unit III

5. Explain building of stack and dynamic libraries. Also mention their uses.
6. (a) Explain fixed and variable sized block of data files in brief.
(b) Explain content defined chunking.

Unit IV

7. (a) Explain job control commands in linux with examples.
(b) Explain signals and signal handlers in linux.
8. Explain the following with examples :
 - (a) Networking tools in Linux
 - (b) Firewall
 - (c) Backup.

T-7/D-16
UNIX AND LINUX PROGRAMMING
CSE-471

Time : Three Hours [Maximum Marks : 75

Note : Attempt Five questions in all, selecting at least one question from each Section.

Section – I

- | | | |
|---|---|---|
| 1. a) Write a program in shell script to sort a list of words in given file in ascending order. | 6 | |
| b) Explain the working and use of (i) mv (ii) lpr (iii) ps commands of Linux. | | 9 |
| 2. a) Write short notes to explain the use of (i) ls (ii) cat (iii) proc commands of Linux. | | 9 |
| b) How functions can be used in shell scripts ? Explain using an example. | 6 | |

Section – II

- | | | |
|--|---|---|
| 3. a) What features of awk make it a language useful for shell scripts ? | 6 | |
| b) Write a program in Perl to calculate factorial of a number. | | 9 |
| 4. a) Write a program in Perl to sort ten given numbers in descending order. | | 9 |
| b) Write a program using awk syntax to calculate and print squares of first ten old numbers. | 6 | |

Section – III

- | | | |
|---|---|---|
| 5. a) Which C compilers are available for use on Linux ? What is the role of make files ? What advantages do these offer as compared to normal compilation? | 7 | |
| b) Differentiate between statically linked and dynamically linked binaries. | | 8 |
| 6. a) How is gdb used to debug a program ? Use an example to discuss. | | 7 |
| b) Write short notes to discuss the working and use of
(i) static (ii) dynamic libraries. | | |

Section – IV

- | | | |
|--|---|---|
| 7. a) Write short notes to discuss (i) rc and (ii) init files. How are these useful in system setup and control? | 8 | |
| b) How are network files accessed and used in Linux? | | 7 |
| 8. a) What type of security provisions are there in Linux ? What is role of privileges in security? | | 8 |
| b) How system passwords are handled and managed in Linux? | | 7 |

UNIX AND LINUX PROGRAMMING
CSE-471

Time : Three Hours] [Maximum Marks : 75

Note : Attempt Five questions in all, selecting at least one question from each Unit.

All questions carry equal marks.

Unit I

- I. (a) Describe the shell functions. 5
 (b) Write a shell script to display the number of words and number of lines present in a given file. 10
2. (a) Explain the process of logging-in and logging-out in Linux. 7
 (b) What do you mean by shell variables? Where the shell variables are used? Explain by giving examples. 8

Unit II

3. (a) Explain various grep options to display a count, line number and file names.
 (b) Explain the text editing features of sed using suitable examples. 7, 8
4. (a) Explain the mam strength of peri language. How is it better than awk language? 7
 (b) Find out the occurrences of three consecutive and identical word characters like aaa or bbb using : (i) grep (ii) sed. 8

Unit III

5. (a) Explain the features of gcc compiler along with options. 7
 (b) Write down the various steps in tracing a program using gdb. 8
6. (a) Explain the concept of statically linked and dynamically linked programs. 7
 (b) Write short notes on the following : 8
 (i) Use of make files
 (ii) Soname.

Unit IV

7. (a) What is difference between a job and a process? How do you
 i) Suspend the foreground job. 8
 ii) move a suspended job to background.
 iii) bring back a suspended job to foreground ?
 (b) Write short note on job control at, batch, cron, time. 7
8. (a) What are signals and signal handlers ? Explain. 7
 (b) What is system authentication and password administration? Discuss. 8

CSE-403N Computer Graphics and Animation						
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
4	0	0	75	25	100	03 Hrs.
Purpose	Introduces Computer Graphics that help in designing different kinds of static and movable objects.					
Course Outcomes (CO)						
CO1	Understand the working Of clipping In graphics.					
CO2	understand the concept of hidden surfaces using algorithms					
CO3	graphical problem solved using geometrical transformations					
CO4	understanding and deployment of 2D and 3D graphics processing techniques					
CO5	Understand concept of Shading: illumination and surface modeling.					
CO6	To provide comprehensive introduction about computer graphics system.					

Unit-I

Computer Graphics applications, Display Devices, Point & Positioning Devices, Plotting Techniques for point and Line, Line drawing algorithms: DDA, Bresenham's Circle drawing algorithms, Filled area algorithms: Scan line, Polygon filling algorithms, Boundary filled algorithms.

Unit-II

Window to view port transformation, Window to view port mapping, Two Dimensional transformation: translation, scaling, rotation, reflection and Shear, Homogeneous Coordinate system.
3-D transformation: Rotation, Shear, translation, Numerical Problems of transformation viewing pipeline.

Unit-III

Clipping: Point & Line clipping algorithm, 4-bit code algorithm, Cohen-Sutherland Line clipping algorithms, Liang-Barsky line clipping algorithms. Polygon clipping: Sutherland-Hodgeman Polygon clipping algorithm. Curve clipping, Text clipping.
Projection: Parallel, Perspective, Vanishing Points.

Unit-IV

Representation of 3-D Curves and Surfaces: interpolation and approximation alphas, parametric conditions, Geometric continuity conditions, Beizer curves and surfaces: properties of beizer curves, beizer surfaces.
Hidden Surfaces removal: Hidden surface elimination, depth buffer algorithm, scan line coherence and area coherence algorithm, priority algorithm

Text Books

1. Donald Hearn & M.Pauline Baker, Computer Graphics, 2nd Edition, Pearson Education.
2. William M. Newmann & Robert F. Sproull, Principles of Interactive Computer Graphics, Tata McGraw-Hill Second Edition, New Delhi, India.
3. Zhigang Xiang & Roy A Plastock , Computer Graphics, Second Edition, Schaum's Outline, Tata McGraw Hill Education Private Limited, New Delhi, India.

Reference Book

1. Foley, van Dam, Feiner, and Hughes. Computer Graphics: Principles and Practice, 3rd edition in C.

LESSON PLAN

Month	Class	Topic/Chapter Covered	Academic Activity	Test/Assignment
August L1	B.Tech 7th sem	what is computer graphics, type, applications	Lecture	
L2	B.Tech 7th sem	line drawing, Line and point plotting system	Lecture	
L3	B.Tech 7th sem	What is CRT, working of CRT	Lecture	
L4	B.Tech 7th sem	working Color CRT	Lecture	
L5	B.Tech 7th sem	Raster scan techniques	Lecture	
L6	B.Tech 7th sem	Working of Raster scan system	Lecture	
L7	B.Tech 7th sem	Random scan techniques	Lecture	
L8	B.Tech 7th sem	Working of Random scan system	Lecture	
L9	B.Tech 7th sem	Refresh and storage display	Lecture	
L10	B.Tech 7th sem	frame buffer, display controller, how ICG works	Lecture	
L11	B.Tech 7th sem	Plasma panel, LCD, LED	Lecture	
L12	B.Tech 7th sem	Line Drawing Algorithms	Lecture	
L13	B.Tech 7th sem	Bresenham's Line Drawing Algorithms	Lecture	
L14	B.Tech 7th sem	Numerical on DDA and Bresenham's Algorithm	Lecture	
L15	B.Tech 7th sem	circle drawing algorithm	Lecture	
L16	B.Tech 7th sem	Mid-Point circle drawing algorithm	Lecture	
L17	B.Tech 7th sem	Filled area algorithms: Scan line Algorithm	Lecture	
L18	B.Tech 7th sem	Polygon filling algorithms	Lecture	
L19	B.Tech 7th sem	Boundary filled algorithms	Lecture	
L20	B.Tech 7th sem	Window to view port transformation,	Lecture	
L 21	B.Tech 7th sem	Window to view port mapping,	Lecture	
L22	B.Tech 7th sem	Numerical related to Window to view port mapping,	Lecture	
L 23	B.Tech 7th sem	2D transformation (Translation, scaling)	Lecture	
L 24	B.Tech 7th sem	2D transformation (Rotation, Numerical problem)	Lecture	
L 25	B.Tech 7th sem	Homogenous coordinates and matrix formation	Lecture	
L 26	B.Tech 7th sem	Shearing, Reflection	Lecture	
L 27	B.Tech 7th sem	Numerical on Transformation	Lecture	
L28	B.Tech 7th sem	3D transformation	Lecture	
L29	B.Tech 7th sem	Rotation, Translation, Shear	Lecture	
L30	B.Tech 7th sem	Numerical Problems of transformation	Lecture	
L31	B.Tech 7th sem	viewing pipeline.	Lecture	
L 32	B.Tech 7th sem	Homogeneous Coordinates	Lecture	
L 33	B.Tech 7th sem	Numerical of window to viewport Transformation	Lecture	
L34	B.Tech 7th sem	Clipping: Point & Line clipping algorithm	Lecture	
L35	B.Tech 7th sem	4-bit code algorithm,	Lecture	
L 36	B.Tech 7th sem	Cohen-Sutherland Line clipping algorithms	Lecture	
L 37	B.Tech 7th sem	Liang-Barsky line clipping algorithms	Lecture	
L38	B.Tech 7th sem	Polygon clipping: Introduction	Lecture	
L39	B.Tech 7th sem	Sutherland-Hodgeman Polygon clipping	Lecture	

L40	B.Tech 7th sem	Flow chart of Sutherland-Hodgeman Polygon clipping algorithm	Lecture	
L41	B.Tech 7th sem	Numerical related to Clipping	Lecture	
L 42	B.Tech 7th sem	Curve clipping	Lecture	
L43	B.Tech 7th sem	Text clipping	Lecture	
L44	B.Tech 7th sem	Introduction to Transformation	Lecture	
L46	B.Tech 7th sem	Types of Transformation	Lecture	
L47	B.Tech 7th sem	Vanishing Points	Lecture	
L48	B.Tech 7th sem	Difference between Parallel and Perspective Transformation	Lecture	
L49	B.Tech 7th sem	Parallel Transformation and their types	Lecture	
L50	B.Tech 7th sem	Perspective Transformation and their types	Lecture	
L51	B.Tech 7th sem	Representation of 3-D Curves and Surfaces: An Introduction	Lecture	
L52	B.Tech 7th sem	interpolation	Lecture	
L53	B.Tech 7th sem	parametric conditions	Lecture	
L54	B.Tech 7th sem	Bezier curves and surfaces: An Introduction	Lecture	
L55	B.Tech 7th sem	Properties of Beizer curves	Lecture	
L56	B.Tech 7th sem	Bezier surfaces	Lecture	
L57	B.Tech 7th sem	Introduction to Hidden Surfaces removal	Lecture	
L58	B.Tech 7th sem	Z – Buffer Problem	Lecture	
L59	B.Tech 7th sem	Hidden surface elimination	Lecture	
L60	B.Tech 7th sem	Depth buffer algorithm,	Lecture	

Tutorial Sheet-1

1. Explain CRT with RGB Technology and also differentiate Raster and Random Scan Display
2. What do you mean by Line Drawing and how we can implement DDA line drawing algorithm?
3. Explain the Bresenham's circle drawing algorithm with the example of your own choice?
4. What do you mean by Area Fill algorithms and Boundary Fill algorithm?

Tutorial Sheet-2

1. What is Window to Viewport Mapping and how we can transform images from one device to another device.
2. Explain different type of Transformation with their operations
3. How we can differentiate rotation and reflection in Transformation with the help of examples
4. What do you mean by Curve Clipping and how text is clipped in graphics?

Tutorial Sheet-3

1. What do you mean by Transformation and differentiate Parallel and Perspective Transformation?
2. How we can differentiate Polygon Clipping and Line Clipping with the help of different algorithms
3. What is Projective Transformation and how we can project any object, explain with the help of diagrams.
4. What is 3 D transformation and explain rotation and shearing with the help of Homogenous coordinates.

Tutorial Sheet-4

1. what do you mean by Bezier Curves and explain it with the help of Control points?
2. what is approximation and parametric conditions of curves
3. what is Hidden Surface and explain it with Z buffer algorithm
4. what is Priority algorithm and explain it with the help of diagrams? How it can help to remove the hidden Surface?

Roll No.

Total Pages : 03

BT-7/D-19

37148

COMPUTER GRAPHICS AND ANIMATION
CSE-403

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) Bring out the anatomy of an LCD display device and compare it with CRT display.
(b) Describe the procedure used for filling a polygon when Boundary fill algorithm is used.
2. What is the role of a decision parameter in Bresenham's circle drawing algorithm ? Explain the process of computing increments for Bresenham's circle drawing algorithm. Obtain the points on a circle with centre (4, 6) and radius 5 using Bresenham's circle drawing algorithm.

(3-22/11)L-37148

P.T.O.

Unit II

3. Describe the 2-D viewing transformations that maps a window defined in world co-ordinates onto a normalized view port. Also describe, how a line defined in a viewport can be translated, scaled and rotated ?
4. Scale a rectangle with vertices A(6, 6), B(12, 6), C(6, 10) and D(12, 10) to twice its size keeping point (6, 6) fixed.

Unit III

5. What is the role of parameter in clipping lines using Liang-Barsky line clipping algorithm ? Describe the algorithm in detail and use the algorithm to clip a line with end points (6, 7) and (8, 10) against a viewport with diagonal vertices at (2, 3) and (10, 9).
6. Distinguish between Parallel projection and Perspective projection and describe, how points defined in eye co-ordinate system can be mapped onto screen co-ordinate system using these projections ?

Unit IV

7. What is the parametric representation of cubic curves ?
How are Bezier curves drawn using parametric representation ? Are Bezier curves based on interpolation or approximation ?
8. Bring out a comparison between Depth buffer and area subdivision hidden surface algorithms.

Roll No.

37148

Printed Pages : 2

BT-7 / M-19
COMPUTER GRAPHICS AND ANIMATION
Paper-CSE-403 N

Time allowed : 3 hours [Maximum marks : 75]

Note : All questions are compulsory. Attempt five questions in all, selecting at least one from each unit.

Unit-I

1. (a) Explain any one boundary filled algorithm with example. 8
- (b) Write short note on plotting techniques for line. 7
2. Derive Bresenham's circle drawing algorithm and explain it with an example. 15

Unit-II

3. (a) Discuss window to view point mapping. 7
- (b) How reflection is done in 2-D transformation ? Show it with an example. 8
4. Explain 3-D shearing and rotation with example. 15

Unit-III

5. Explain Liang-Barsky line clipping algorithm with example. 15
6. What do you mean by projection ? Explain parallel and perspective projection in detail. 15

Unit-IV

7. Explain : 15
 - (a) Interpolation and approximation alpiners
 - (b) Bezier Surfaces
8. Explain Scan line coherence and area coherence algorithm with an example. 15

BT-8/M-15 8802
Interactive Computer Graphics
Paper-CSE-404

[Time Allowed:3 Hours]

[Maximum Marks:100]

Note: Attempt five questions in all, selecting at least one question from each unit. All questions carry equal marks.

UNIT-I

- 1.(a) What do you mean by Display processors? Explain the concepts of character generator? 10
(b) Compare and contrast the working of raster scan and random scan display systems? 10
2. Discuss the functioning of monochrome and color display techniques? 20

UNIT-II

- 3.(a) Explain the DDA line drawing algorithm. What are the advantages and disadvantages of DDA algorithm? Implement the DDA algorithm to draw a line from(0,0) to(6,6). 10
(b) What do you mean by 2-D transformation? Explain translation, shearing and rotation in 2-D transformation. 10
- 4.(a) Explain mid-point circle drawing algorithm, plot a circle by using mid-point algorithm whose radius is 10 units. 10
(b) Explain screen co-ordinates and user co-ordinates. 10

UNIT-III

5. (a) Write short note on mouse programming and interactive graphical techniques. 10
(b) Explain zooming and clipping positioning techniques. 10
- 6 Explain different pointing and positioning devices. 10

UNIT-IV

7. Discuss 3-D transformations and explain different 3-D geometric transformations. 10
- 8.(a) Explain the concept of projection. What do you mean by parallel projection? 10
(b) Give brief description of hidden line and surfaces. Discuss anyone algorithm used for removal of hidden line and surfaces. 10

**Interactive Computer Graphics
Paper-CSE-404**

[Time Allowed:3 Hours]

[Maximum Marks:100]

Note: Attempt five questions in all, selecting at least one question from each unit. All questions carry equal marks.

UNIT-I

- 1.(a) Compare and contrast organization of raster scan and random scan display systems. 10
(b) Explain the use of color lookup tables. 10
- 2.(a) What is function of frame buffer and display process or in an interactive computer graphics? Consider raster system with resolution: 1280x1024 What size frame buffer (in bytes) is needed for the system to store 24 bits per pixel? 10
(b) Explain the constructions and working beam penetration CRT. 10

UNIT-II

- 3.(a) Write Bresenham 's Line drawing algorithm for a line whose slope is between 0° and 45° . 10
(b) Use the algorithm given in part (a) to scan convert a line from pixel coordinate (1,1) to (5,5). 10
4. Explain window-to-view point coordinate transformation. 20

UNIT-III

5. Explain the working of the following input devices: (a) Light Pen 10
(b) Track Ball
- 6.(a) Write down the algorithm for the Cohen-Sutherland line clipping. Also explain its working. 10
(b) Explain the following position in techniques: Rubber-Band and Panning.

UNIT-IV

- 7.(a) Explain the following 3D transformation along with their matrix representations: Rotation and Translation. 10
(b) What is oblique parallel projection? How is it different from perspective projection. 10
- 8.(a) How are hidden surface calculations? simplified in area subdivision algorithm? 10
(b) Explain the Phong specular reflection model of shading. 10

CSE-415N						
Object Oriented Software Engineering						
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	0	0	75	25	100	3 Hrs.
Purpose	To provide the thorough knowledge to use the concepts and their design attributes for Object Oriented Software Engineering approaches and platforms to solve real time problems.					
Course Outcomes (CO)						
CO1	To understand the differences between procedural design and object-oriented design.					
CO2	To understand the Concept of inheritance, polymorphism, dynamic binding and Generic structures					
CO3	Object-oriented program design based on template model of C++.					
CO4	for creating efficient codes, exception handling Mechanisms use in C++					
CO5	Perform object oriented programming to develop solutions to problems demonstrating usage of control structures, modularity, I/O. and other standard language constructs.					
CO6	To learn how to design C++ classes for code reuse.					

Unit - I

An Overview of Object-Oriented system Development, Objects Basis, Class Hierarchy, Inheritance, Polymorphism, Object Relationships and Associations, Aggregations and Object Containment, Object Persistence, Meta-Classes, Object Oriented Systems Development Life Cycle: Software Development Process, Object Oriented Systems Development: A Use-Case Driven Approach.

Unit - II

Object Oriented Methodologies: Rumbaugh Methodology, Jacobson Methodology, Booch Methodology, Patterns, Frameworks, The Unified approach, Unified Modeling Language (UML)

Unit - III

Object Oriented Analysis Process, Use Case Driven Object Oriented Analysis, Use Case Model, Object Analysis: Classification, Classification Theory, Approaches for identifying classes, Responsibilities and Collaborators, Identifying Object Relationships, Attributes and Methods: Associations, Super-Sub Class relationships, A-Part-of-Relationships-Aggregation, Class Responsibilities, Object Responsibilities.

Unit - IV

Object Oriented Design process and Design Axioms, Corollaries, Design Patterns, Designing Classes: Object Oriented Design Philosophy, UML Object Constraint Language, Designing Classes: The Process, Class Visibility, Refining Attributes, Designing Methods and Protocols, Packages and Managing classes, View Layer: Designing Interface objects, Designing View layer Classes, Macro and Micro Level Interface Design Process.

Text Books:

1. Ali Bahrami, Object Oriented Systems Development, McGraw Hill Publishing Company Limited, New Delhi, 2013.
2. Rumbaugh *et al.*, Object Oriented Modeling and Design, PHI, 2006.
3. Robert Laganière and Timothy C. Lethbridge, Object-Oriented Software Engineering: Practical Software Development, McGraw-Hill Publishing Company Limited, New Delhi, Sixth Print 2008.

Reference Books:

1. Ivar Jacobson, Magnus Christerson, Patrick Jonsson, Gunnar Overgaard, Object-oriented Software Engineering: A Use Case Driven Approach, Pearson Education, New Delhi, Seventh Edition Reprint, 2009.
2. David C. Kung, Object-Oriented Software Engineering: An Agile Unified Methodology, McGraw-Hill Publishing Company Limited, New Delhi, 2013.
3. Bernd Bruegge, Allen H. Dutoit, Object-Oriented Software Engineering Using UML, Patterns, and Java: Pearson New International, Third Edition, 2013.

LESSON PLAN

Week	Lect.	Theory
1	L1	Design Objects
	L2	Class Hierarchy, Inheritance
2	L3	Polymorphism,
	L4	Object relationships and associations
	L5	Aggregations and Object Containment
3	L6	Object Persistence
	L7	Meta classes
	L8	Object oriented systems development life cycle
4	L9	Software development process
	L10	Object oriented systems development: a use case driven approach
	L11	Object oriented systems development: a use case driven approach
5	L12	Object modeling techniques as software as software engineering methodology
	L13	Object modeling techniques as software as software engineering methodology
	L 14	Rumbaugh methodology
6	L 15	Jacobson methodology
	L 16	Booch methodology
	L 17	Patterns
7	L 18	Frameworks
	L 19	Unified Modeling language
	L20	Analysis Process
8	L 21	Use-Case Driven Object Oriented Analysis
	L 22	Use-Case Model
	L23	Object Classification Theory
9	L 24	Different Approaches for identifying classes
	L 25	classes, responsibilities and collaborators
	L26	classes, responsibilities and collaborators
10	L 27	identifying Object Relationships
	L 28	attributes and Methods
	L29	super-sub class relationship
11	L30	Apart of Relationship-Aggregation
	L 31	Class Responsibilities and Object Responsibilities
	L 32	Class Responsibilities and Object Responsibilities
12	L33	Object Classification Theory
	L34	Object Oriented design process, corollaries
	L35	design axioms, design patterns
13	L 36	object oriented design philosophies
	L 37	UML Object Constraint Language
	L 38	Designing Classes: The Process,

14	L39	Class Visibility, Refining Attributes
	L40	Designing Methods and Protocols
	L41	Packages and managing classes
15	L42	Designing interface objects
	L43	View layer interface design
	L44	Macro level interface design process.
16	L45	Micro level interface design process

Roll No.

Total Pages : 03

BT-7/D-19

37149

OBJECT ORIENTED SOFTWARE
ENGINEERING
CSE-415N (Option I)

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Section.

Section I

1. (a) Explore the different phases of Object-oriented System development life-cycle. **8**
- (b) Explain the role and their usage mechanism of object persistence and Meta classes in Object-oriented design system. **7**
2. (a) What type of transformations are used to develop software process ? Explain the limitations of using Waterfall approach in Software development process. **8**
- (b) Discuss Inheritance and Polymorphism in Object-oriented Software Engineering. **7**

Section II

3. (a) What is UML ? Explain different views supported by UML. 8
- (b) What is basic role of using Patterns ? Explain Generative and Non-generative Patterns. 7
4. Explain the comparison between Rumbaugh, Booch and Jacobson Methodology. 15

Section III

5. (a) Explain the role of classes, responsibilities and collaborators process in Object oriented analysis. 8
- (b) What guidelines are used to identify a-part-of relationship ? How to eliminate unnecessary association ? 7
6. (a) Why Use-case model is used in project planning, development and documentation of systems ? 8
- (b) Write a detailed note on Attributes and Methods. 7

L-37149

2

Section IV

7. (a) What is significance of Occam's razor ? How Occam's razor relate to object oriented design ? 8
- (b) Explain the significance of various Micro-level user interface design process. 7
8. (a) Explain designing concept and principles ? How Interface objects are designed ? 8
- (b) Discuss various Design axioms and Design Patterns. 7

OBJECT ORIENTED SOFTWARE
ENGINEERING
CSE-415N

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) Discuss the following by taking suitable examples :
 - (i) Object relationships and Associations 4
 - (ii) Aggregations and Object Containments. 4
- (b) Discuss various ways by which the lifetime of an object can be extended. 7

2. (a) Explain the role of object containment and object persistence to identify and classes. 8
- (b) Identify the design pattern and quality indicators which are required to build high quality software.

Unit II

3. (a) What is the basic role of using patterns ? Explain the following with respect to the patterns :
- (i) Generative and Non-generative Patterns. 8
 - (ii) Antipatterns and Pattern Thumbnail. 8
- (b) How object-oriented business engineering and object-oriented software engineering approaches are used to design efficient and time critical systems ? 7
4. (a) Why model constrains and stereotypes are used in UML ? Explain the modus operandi of various stereotypes notations. 8
- (b) Identify the role of activity diagram and sequence diagram in UML. 7

Unit III

5. (a) What are the various steps that are used on object-oriented analysis ? Explain using unified process approach. 8
- (b) Explain various roles which are associated with object responsibilities to access and pass accurate methods and messages. 7

6. (a) Why use case model is used in project development, planning and documentation of systems ? Explain the following in context of use cases :
- (i) Guidelines for finding use cases
 - (ii) Dividing use cases into packages
 - (iii) Naming a use case. 10
- (b) Why uses and extends associations are useful in use-case modeling ? 5

Unit IV

7. (a) What is the task of design ? Why do we need analysis ? 5
- (b) What is the relationship between coupling and cohesion ? 5
- (c) Explain design patterns by taking a suitable example. 5
8. (a) Define Corollaries. Explain six design corollaries which are used in the objected-oriented design process and deign axioms, 8
- (b) Explain the significance of various Micro-level user interface design process. 7

BT-8/M-138807

OBJECT ORIENTED SOFTWARE ENGINEERING

Paper-CSE-472

Time Allowed: 3Hours

MaximumMarks:75

Note: Attempt five questions in all, selecting at least one question from each Unit. All questions carry equal marks.

UNIT-I

- 1.(a) How is Object-oriented Software Development different from traditional Software development? 10,5
(b) Describe the following terms: Object persistence. And Meta classes.
2. Explain the following terms with the help of suitable examples: 15
a) Inheritance b) Polymorphism c) Object Relationships and Associations.

UNIT-II

- 3.(a) Give a brief description of the characteristics of Object-oriented modeling. 10
(b) Write a short note on UML. 5
4. Compare and contrast the Object-oriented methodology of Booch, Rumbaugh and Jacobson in detail. 15

UNIT-III

- 5.(a) What is a Use-case model? Why is the use case modeling useful in analysis? 10
(b) Explain the different approaches for identifying classes. 5
6. Explain the following terms: a) Class responsibilities b) Object responsibilities.
c) Aggregation

UNIT-IV

- 7.(a) Write in brief about Object-oriented design Philosophy. 8
(b) Explain Class visibility with suitable example. 7
- 8.(a) Write a short note on Macro and Micro level interface design process. 7
(b) Explain the Object-oriented design in methods and protocols. 4
(c) Write a short note on Design patterns. 4

CSE-425N	Expert Systems					
Lecture	Tutorial	Practical	Major Test	Minor Test	Total	Time
3	0	0	75	25	100	3 Hrs.
Purpose	In this course the student will learn the methodologies used to transfer the knowledge of a human expert into an intelligent program that can be used to solve real-time problems.					
Course Outcomes(CO)						
CO1	Concepts of expert systems will be inserted.					
CO2	Students become capable to design expert system using knowledge-based software tools.					
CO3	Enable students for knowledge (acquisition) and specify it clearly (validation).					
CO4	To design knowledge structure along with production planning, quality control and other subsystems of an industrial organization.					
CO5	To insert the characteristics of a feasible expert system.					
CO6	To use various knowledge representation methods and expert system structures from the industrial engineering point of view.					

Unit-I

Introduction to AI programming languages, Blind search strategies, Breadth first – Depth first – Heuristic search techniques Hill Climbing – Best first – A Algorithms AO* algorithm – game trees, Min-max algorithms, game playing – Alpha beta pruning.

Knowledge representation issues predicate logic – logic programming Semantic nets- frames and inheritance, constraint propagation; Representing Knowledge using rules, Rules based deduction systems.

Unit-II

Introduction to Expert Systems, Architecture of expert system, Representation and organization of knowledge, Basics characteristics, and types of problems handled by expert systems.

Expert System Tools: Techniques of knowledge representations in expert systems, knowledge engineering, System-building aids, support facilities, stages in the development of expert systems.

Unit-III

Building an Expert System: Expert system development, Selection of tool, Acquiring Knowledge, Building process.

Unit-IV

Problems with Expert Systems: Difficulties, common pitfalls in planning, dealing with domain expert, difficulties during development.

Text Books

1. Elain Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw-Hill, New Delhi, 2008.
2. Waterman D.A., “A Guide to Expert Systems”, Addison Wesley Longman, 1985.

Reference Books

1. Stuart Russel and other Peter Norvig, “Artificial Intelligence – A Modern Approach”, Prentice Hall, 1995.
2. Patrick Henry Winston, “Artificial Intelligence”, Addison Wesley, 1979.
3. Patterson, Artificial Intelligence & Expert System, Prentice Hall India, 1999.
4. Hayes-Roth, Lenat and Waterman: Building Expert Systems, Addison Wesley, 1983.
5. Weiss S.M. and Kulikowski C.A., “A Practical Guide to Designing Expert Systems”, Rowman & Allanheld, New Jersey, 2011.

LESSON PLAN

WEEK	LECTURE	THEORY
1ST	1ST	Introduction to AI programming languages
	2ND	Blind search strategies, Breadth first – Depth first
	3RD	Heuristic search techniques Hill Climbing – Best first
2ND	4TH	A Algorithm, AO* algorithm
	5TH	Game trees, Min-max algorithms, game playing – Alpha beta pruning.
	6TH	Knowledge representation issues predicate logic – logic programming
3RD	7TH	-DO-
	8TH	Semantic nets- frames and inheritance
	9TH	constraint propagation;
4TH	10TH	Representing Knowledge using rules
	11TH	Rules based deduction systems
	12TH	REVISION OF UNIT -I
5TH	13TH	Introduction to Expert Systems
	14TH	Architecture of expert system
	15TH	Representation and organization of knowledge
6TH	16TH	-DO-
	17TH	Basics characteristics, and types of problems handled by expert systems
	18TH	-DO-
7TH	19TH	Techniques of knowledge representations in expert systems
	20TH	-DO-
	21ST	-DO-
8TH	22ND	knowledge engineering
	23RD	System-building aids
	24TH	support facilities
9TH	25TH	stages in the development of expert systems
	26TH	-DO-
	27TH	REVISION OF UNIT -II
10TH	28TH	Expert system development
	29TH	-DO-
	30TH	-DO-
11TH	31ST	Selection of tool
	32ND	Acquiring Knowledge
	33RD	-DO-
12TH	34TH	Building process
	35TH	REVISION OF UNIT -III
	36TH	Difficulties, common pitfalls in planning
13TH	37TH	-DO-
	38TH	-DO-
	39TH	dealing with domain expert
14TH	40TH	-DO-
	41ST	-DO-
	42ND	difficulties during development
15TH	43RD	-DO-
	44TH	-DO-
	45TH	REVISION OF UNIT-IV

TUTORIALSHEET-1

1. Define Expert System. What are their components?
2. What are the various features of Expert System?
3. Describe the types of problems handled by Expert System?
4. What are the characteristics of Expert System?
5. A case study for development of PUFF system using EMYCIN.
6. A case study of PROSPECTOR System.

TUTORIALSHEET-2

1. What is the difference b/w data processing and knowledge engineering?
2. How is the knowledge represented in Expert systems?
3. Describe the system building aids.
4. What are the stages in the development of Expert System?
5. Describe knowledge Engineering.
6. Describe the role of knowledge engineering in the development of a hybrid knowledge based medical information system.

TUTORIALSHEET-3

1. What are the necessary requirements for Expert System development?
2. Explain the various characteristics that make the use of Expert System appropriate.
3. What are the different types of tools available for Expert System building?
4. Explain the message passing in object oriented programs.
5. Explain forward chaining.
6. Development of a Real-Time Knowledge-Based System at Eli Lilly using Knowledge Acquisition

TUTORIALSHEET-4

1. What are the various limitations of Expert System building tools?
2. Explain the various pitfalls while choosing an appropriate problem.
3. What types of problems come while choosing the domain expert?
4. Explain the various pitfalls while choosing the resources for building the system?
5. What are the various pitfalls during system development?
6. Use of Multiple Domain Experts in the Design and Development of an Expert System for Disaster Recovery Planning.

Roll No.

Total Pages : 0

BT-7/D-19

37154

EXPERT SYSTEMS
CSE-425N

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) Discuss some of the potential problems, when hill climbing search is used ? Give example of each. 7
- (b) What are alpha and beta cutoffs for minmax algorithm ? Explain the role of alpha and beta values. 8
2. (a) Elaborate a decision tree architecture of an expert system. 7
- (b) Identify and describe two good application areas for an expert system with a university environment. 8

Unit II

3. (a) What basic operations, must a program perform in order to access specific knowledge ? 7
- (b) Write a note on conceptual dependency for knowledge acquisition ? 8
4. (a) What is knowledge based architecture of an expert system ? What are its advantages ? 7
- (b) Give an example of the use of metaknowledge in an expert system inference. 8

Unit III

5. Discuss the following expert system development stages :
 - (a) Decision about development
 - (b) Tools for design
 - (c) Implementation. 5,5,5
6. (a) Discuss utility of expert system in control and prediction application areas. 8
- (b) What are the problems of present days expert systems ? 7

L-37154

2

Unit IV

7. Discuss the role of the following persons in expert system development :
 - (a) Domain Expert 7
 - (b) Updation, review and changeover. 8
8. Enlist advantages of using an expert system in analysis, design and planning areas. 5,5,5

Roll No.

Total Pages : 03

BT-7/D-18

37154

EXPERT SYSTEMS

CSE-425N

Time : Three Hours]

[Maximum Marks : 75

Note : Attempt *Five* questions in all, selecting at least *one* question from each Unit. All questions carry equal marks.

Unit I

1. (a) When is it advisable to use depth-first search over breadth-first search and vice-versa ? Discuss.
(b) Differentiate between declarative frames and procedural frames using suitable examples.
2. (a) What is mini-max search ? Discuss the use of alpha and beta cut offs to refine the mini-max search using suitable examples.
(b) What are the desirable characteristics of a knowledge representation scheme ? Write a note on representing the knowledge using rules.

Unit II

3. (a) Discuss the factors motivating the use of Expert Systems.
(b) What are the advantages of keeping the knowledge base separate from inference mechanism in Expert Systems ? Discuss.
4. (a) What are the different techniques of representing the knowledge in Expert Systems ? Discuss.
(b) What are the types of problems handled by Expert Systems ? Discuss.

Unit III

5. (a) What are the different tools required for Expert System development ? Discuss.
(b) What are the different stages in Expert system development ? Discuss.
6. Expert knowledge is generally subconscious in nature. So how is it acquired to populate the knowledge base of Expert Systems ? Discuss.

Unit IV

7. Write a detailed note on common pitfalls in planning for expert system development.
8. What are the common inconsistencies observed in rule-based knowledge representation ? Discuss.

BT-81M-138805

EXPERTSYSTEMS

Paper-CSE-446

TimeAllowed:3Hours

MaximumMarks75

Note:Attempt five questions in all,selecting at least one question from eachUnit

UNIT-I

- 1.(a) Explain expert systems and write their important applications. 2,6,3,4
(b) What is the need of knowledge for developing an expert system? How is knowledge organized?

2. Write down the architecture of an expert system, explaining each component.
How do expert systems differ from Production systems? 12,3

UNIT-II

- 3.(a) In the development of an expert system we need knowledge which needs to be organized. Explain.
(b) Describe briefly various methods of knowledge representation, highlighting their merits and limitations. 4,11
- 4.(a) Write about the various stages of developing an expert system, showing their interactive nature.
(b) Explain Knowledge Engineering and Paradigm shift in the domain of expert systems. 9,3,3

UNIT- III

5. What are various methods of acquiring knowledge for the expert system with special reference to
a) On-site observation b) Problem analysis c) Interviewing the domain expert. 4,4,7
6. Write a complete analysis of developing an expert system in terms of
a) Selection of expert system tool
b) Knowledge consistency
c) Reliability. 8,4,3

UNIT--IV

7. Write down various pitfalls in the planning and development of an expert system. 15
8. Write explanatory notes on
a) Future of expert systems
b) Applications of expert systems. 7, 8

CSE-405N	Computer Graphics Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
0	0	2	40	60	100	3Hrs.
Purpose	To Design and implement various Line and Circle Drawing Algorithms.					
Course Outcomes (CO)						
CO1	To Implement basic algorithms related to Line & Circle Drawing.					
CO2	Implement various Line & Circle Drawing Algorithms.					
CO3	Hands on experiments on 2-D transformations.					
CO4	Conceptual implementation of Clipping and other drawing algorithms.					
CO5	Implementation of boundary fill and Polygon clipping algorithm.					
CO6	To draw rectangle using Bresenham's and DDA Algorithm					

List of Practicals:

1. Write a program to implement DDA line drawing algorithm.
2. Write a program to implement Bresenham's line drawing algorithm.
3. Implement the Bresenham's circle drawing algorithm.
4. Write a program to draw a decagon whose all vertices are connected with every other vertex using lines.
5. Write a program to move an object using the concepts of 2-D transformations.
6. Write a program to implement the midpoint circle drawing algorithm using any Object Oriented Programming Language like Python, C++, Java.
7. Implement the line clipping algorithm using any Object Oriented Programming Language like Python, C++, Java.
8. Implement boundary fill algorithm using any Object Oriented Programming Language like Python, C++, Java.
9. Implement the depth buffer algorithm using any Object oriented language like Python, C++, Java.
10. Perform the Polygon Clipping Algorithm using any Object oriented language like Python, C++, Java.
11. Draw a Rectangle using Bresenham's and DDA Algorithm using any Object oriented language like Python, C++, Java.

Note: At least 5 to 10 more exercises are to be given by the teacher concerned.

LESSON PLAN

Week	Theory	
	Lecture Day	Topic
1	1st	Introduction to the subject
2	2nd	Write a program to implement DDA line drawing algorithm.
3	3rd	Write a program to implement Bresenham's line drawing algorithm.
4	4th	Implement the Bresenham's circle drawing algorithm.
5	5th	Write a program to draw a decagon whose all vertices are connected with every other vertex using lines.
6	6th	Write a program to move an object using the concepts of 2-D transformations.
7	7th	Write a program to implement the midpoint circle drawing algorithm using any Object Oriented Programming Language like Python, C++, Java.
8	8th	Implement the line clipping algorithm using any Object Oriented Programming Language like Python, C++, Java.
9	9th	Implement boundary fill algorithm using any Object Oriented Programming Language like Python, C++, Java.
10	10th	Implement the depth buffer algorithm using any Object oriented language like Python, C++, Java.
11	11th	Perform the Polygon Clipping Algorithm using any Object oriented language like Python, C++, Java.
12	12th	Perform the Polygon Clipping Algorithm using any Object oriented language like Python, C++, Java.
13	13th	Draw a Rectangle using Bresenham's and DDA Algorithm using any Object oriented language like Python, C++, Java.
14	14th	Draw a Rectangle using Bresenham's and DDA Algorithm using any Object oriented language like Python, C++, Java.
15	15th	Internal Viva

CSE-409N	Unix & Linux Programming Lab					
Lecture	Tutorial	Practical	Minor Test	Practical	Total	Time
0	0	2	40	60	100	3 Hrs.
Purpose	To provide experimental knowledge of Unix & Linux Programs					
Course Outcomes (CO)						
CO1	Installation of Linux OS					
CO2	To implement commands in vi editor					
CO3	Shell Programming					
CO4	Programming using awk					
CO5	Programming using PERL					
CO6	To implement programs in C environment					

List of Practicals

1. Familiarize with Unix/Linux logging/logout and simple commands.
2. Familiarize with vi editor.
3. Using Bash shell develops simple shell programs.
4. Develop advanced shell programs using grep, fgrep&egrep.
5. Compile and debug various C programs using different options.
Content defined chunking, frequency based chunking, delta/Xdelta, Rabin Fingerprint Generator, Parallel Compression compress.
6. Learning of installation and upgradation of Linux operating system.
7. Install Linux on a PC having some other previously installed operating system. All operating systems should be usable.
8. As supervisor create and maintain user accounts, learn package installation, taking backups, creation of scripts for file and user management, creation of startup and shutdown scripts using at, cron etc.

Note: At least 5 to 10 more exercises are to be given by the teacher concerned.