

Ref No:

SKIT, BANGALORE



COURSE PLAN

Academic Year AUG 2019

Program:	B E – COMPUTER SCIENCE & ENGINEERING
Semester :	7
Course Code:	15CS744
Course Title:	UNIX SYSTEM PROGRAMMING
Credit / L-T-P:	3 / 3-0-0
Total Contact Hours:	40
Course Plan Author:	MAMATHA T S

Academic Evaluation and Monitoring Cell

No. 29, Chimney hills, Hesaraghatta Road, Chikkabanavara  
BANGALORE-560099, KARNATAKA , INDIA  
Phone / Fax :+91-08023721315/23721477  
[www.skit.org.i](http://www.skit.org.i)

## Table of Contents

UNIX SYSTEM PROGRAMMING.....	1
A. COURSE INFORMATION.....	3
1. Course Overview.....	3
2. Course Content.....	3
3. Course Material.....	4
4. Course Prerequisites.....	4
5. Content for Placement, Profession, HE and GATE.....	5
B. OBE PARAMETERS.....	5
1. Course Outcomes.....	5
2. Course Applications.....	6
3. Mapping And Justification.....	6
4. Articulation Matrix.....	7
5. Curricular Gap and Content.....	8
6. Content Beyond Syllabus.....	8
C. COURSE ASSESSMENT.....	9
1. Course Coverage.....	9
2. Continuous Internal Assessment (CIA).....	9
D1. TEACHING PLAN - 1.....	9
Module - 1.....	9
Introduction.....	9
Module - 2.....	10
E1. CIA EXAM – 1.....	11
a. Model Question Paper - 1.....	11
b. Assignment -1.....	12
D2. TEACHING PLAN - 2.....	13
Module - 3.....	13
Module - 4.....	14
E2. CIA EXAM – 2.....	15
a. Model Question Paper - 2.....	15
b. Assignment – 2.....	15
D3. TEACHING PLAN - 3.....	16
Module - 5.....	16
E3. CIA EXAM – 3.....	17
a. Model Question Paper - 3.....	17
b. Assignment – 3.....	17
F. EXAM PREPARATION.....	18
1. University Model Question Paper.....	18
2. SEE Important Questions.....	19
G. Content to Course Outcomes.....	20
1. TLPA Parameters.....	20
2. Concepts and Outcomes:.....	22

Note : Remove "Table of Content" before including in CP Book  
 Each Course Plan shall be printed and made into a book with cover page  
 Blooms Level in all sections match with A.2, only if you plan to teach / learn at higher levels

## A. COURSE INFORMATION

### 1. Course Overview

Degree:	B.E	Program:	CS
Semester :	VII	Academic Year:	2019-20
Course Title:	UNIX SYSTEM PROGRAMMING	Course Code:	15CS744
Credit / L-T-P:	3/3-0-0	SEE Duration:	180 Minutes
Total Contact Hours:	40	SEE Marks:	80 Marks
CIA Marks:	20	Assignment	1 / Module
Course Plan Author:	Mamatha T S	Sign	Dt:
Checked By:		Sign	Dt:
CO Targets	CIA Target : 90 %	SEE Target:	80%

**Note:** Define CIA and SEE % targets based on previous performance.

### 2. Course Content

Content / Syllabus of the course as prescribed by University or designed by institute. Identify 2 concepts per module as in G.

Module	Content	Teaching Hours	Identified Module Concepts	Blooms Learning Levels
1	The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++,The POSIX Standards,The POSIX.1 FIPS Standard, The X/Open Standards.	05	POSIX standards	L3
	UNIX and POSIX APIs:The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.	03	API characteristics	L2
2	File Types, The UNIX and POSIX File System,UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files.Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.	04	Kernel support for files	L2
	General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs.	04	API for file	L3
3	Introduction, main function, Process Termination,Command-Line Arguments,Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions.UNIX Kernel Support for Processes.Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and GroupProcess Accounting, User Identification,Process Times, I/O Redirection.	05	Process management	L4
	Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups.	03	Process relationship	L4
4	Introduction, Daemon Characteristics, Coding Rules,error Logging, Client-Server Model.The UNIX Kernel Support for signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function,The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lbTimers.	05	Signal handling techniques	L3
	Daemon Characteristics, Coding Rules,error Logging, Client-Server Model.	03	Daemon characteristics	L3
5	Overview of IPC Methods, Pipes, popen, pclose Functions, Co processes ,FIFOs, System V IPC, Message Queues, Semaphores. Shared Memory	04	IPC mechanisms	L4
	Client-Server Properties, Stream Pipes, Passing File	04	Client server	L3

Descriptors, An Open Server-Version 1, Client-Server Connection Functions.		communication	
--	--	---------------	--

### 3. Course Material

Books & other material as recommended by university (A, B) and additional resources used by course teacher (C).

1. Understanding: Concept simulation / video ; one per concept ; to understand the concepts ; 15 – 30 minutes
2. Design: Simulation and design tools used – software tools used ; Free / open source
3. Research: Recent developments on the concepts – publications in journals; conferences etc.

Modules	Details	Chapters in book	Availability
<b>A</b>	<b>Text books (Title, Authors, Edition, Publisher, Year.)</b>	-	-
1,2,4,5	1) Unix System Programming Using C++ - Terrence Chan, PHI, 1999.	1,2,3,4,5,6,7,8,9,10,11,12,13	In Dept/ in library
3	2) Advanced Programming in the UNIX Environment - W.Richard Stevens, Stephen A. Rago, 3rd Edition, Pearson Education / PHI, 2005.	15,16,17	In Dept/ in library
<b>B</b>	<b>Reference books (Title, Authors, Edition, Publisher, Year.)</b>	-	-
1,2,3,4,5	1. Advanced Unix Programming- Marc J. Rochkind, 2nd Edition, Pearson Education, 2005.	-	In Lib
1,2,3,4,5	2. The Design of the UNIX Operating System - Maurice.J.Bach, Pearson Education / PHI, 1987.	-	Not Available
1,2,3,4,5	3. Unix Internals - Uresh Vahalia, Pearson Education, 2001.	-	In lib
1,2,3,4,5	4. Roberto Tamassia, Michael H Goldwasser, Michael T Goodrich, "Data Structures and Algorithms in Python", 1st Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978- 8126562176	-	In lib
<b>C</b>	<b>Example programs links</b>		
	<a href="http://vtuac.in/library/cse/sem7/15cs744/">http://vtuac.in/library/cse/sem7/15cs744/</a>		
	<a href="https://www.vturesource.com/vtu-question-papers/CS/2010/10CS62/Unix-System-Programming">https://www.vturesource.com/vtu-question-papers/CS/2010/10CS62/Unix-System-Programming</a>		
	<a href="https://www.gurugg.com/must-know-linux-commands.html">https://www.gurugg.com/must-know-linux-commands.html</a>		
	<a href="http://cms.gat.ac.in/course/info.php?id=584">http://cms.gat.ac.in/course/info.php?id=584</a>		
<b>D</b>	<b>Software Tools for implementation</b>	-	-
<b>E</b>	<b>Recent Developments for Research</b>	-	-
<b>F</b>	<b>Others (Web, Video, Simulation, Notes etc.)</b>	-	-

### 4. Course Prerequisites

Refer to GL01. If prerequisites are not taught earlier, GAP in curriculum needs to be addressed. Include in Remarks and implement in B.5.

Students must have learnt the following Courses / Topics with described Content . . .

Modules	Course Code	Course Name	Topic / Description	Sem	Remarks	Blooms Level
---------	-------------	-------------	---------------------	-----	---------	--------------

1	15cs64	Operating system	1/operating system concepts /Knowledge of concepts operating system	6		L2
2	15cs35	Unix and shell programming	1/ UNIX architecture/Knowledge of architecture of UNIX operating system	3		L2
3	15cs35	Unix and shell programming	2&3/Unix commands/knowledge of UNIX shell commands	3		L3

## 5. Content for Placement, Profession, HE and GATE

The content is not included in this course, but required to meet industry & profession requirements and help students for Placement, GATE, Higher Education, Entrepreneurship, etc. Identifying Area / Content requires experts consultation in the area.

Topics included are like, a. Advanced Topics, b. Recent Developments, c. Certificate Courses, d. Course Projects, e. New Software Tools, f. GATE Topics, g. NPTEL Videos, h. Swayam videos etc.

Mod ules	Topic / Description	Area	Remarks	Blooms Level
2	File locks	placement	Gap seminar	L3
5	Sockets	placement	Gap presentation	L3

## B. OBE PARAMETERS

### 1. Course Outcomes

Expected learning outcomes of the course, which will be mapped to POs. Identify a max of 2 Concepts per Module. Write 1 CO per Concept.

Mod ules	Course Code.#	Course Outcome At the end of the course, student should be able to . . .	Teach. Hours	Concept	Instr Method	Assessme nt Method	Blooms' Level
1	15CS744.1	Use runtime & compile time limits in UNIX platform	05	POSIX standards	Demonstrate programs	Student presentation of programs	L3 Apply
1	15CS744.2	Understand API characteristics using POSIX standard	03	API characteristics	Reading, discussion	Student presentation Question & answers	L2 Understand
2	15CS744.3	Understand file structure in UNIX operating system.	04	Kernel support for files	Presentation	Question and answers assignment	L2 Understand
2	15CS744.4	Apply file manipulation system calls for different types of files.	04	API for file	Demonstrate programs	Student presentation	L3 Apply
3	15CS744.5	Analyze process control primitives for different applications in multiuser environment	05	Process management	Demonstrate programs	Student presentation of programs	L4 Analyze
3	15CS744.6	Identify relationship between group of processes for job control	03	Process relationship	Demonstrate programs	assignment unit test	L4 Analyze
4	15CS744.7	Apply interrupt methods for	05	Signal	Demonstrate	Seminar	L3

		handling asynchronous events		handling techniques	strate programs	assignments	Apply
4	15CS744.8	Understand daemon characteristics for coding rules	03	Daemon characteristics	Reading, discussion	Question & answers unit test	L3 Apply
5	15CS744.9	Distinguish message queues semaphores & shared memory across machine boundaries	04	IPC mechanisms	Demonstrate programs in lab	Student presentation of programs in groups	L4 Analyze
5	15CS744.10	Discover communication between client server using pipes & sockets	04	Client server communication	Lecture / Demonstrate programs	Student presentation seminars viva	L3 Apply
	-	<b>Total</b>	<b>40</b>	-	-	-	<b>L2-L4</b>

## 2. Course Applications

Write 1 or 2 applications per CO.

Students should be able to employ / apply the course learnings to ...

Modules	Application Area Compiled from Module Applications.	CO	Level
1	Implementation of programs to check limits in UNIX operating system	CO1	L3
1	Use of POSIX in Real-time Systems, Assessing its Effectiveness and Performance	CO2	L2
2	Organization of file in the file system of operating system	CO3	L2
2	Applying of read and write lock on files.	CO4	L3
3	Create, resume, suspend & kill the process in multitasking environment.	CO5	L4
3	Switching among multiple jobs in multiuser & multitasking environment.	CO6	L4
4	Communication between program and operating system	CO7	L3
4	Operating system bootstrapping	CO8	L3
5	Client server communication in distributed computing	CO9	L4
5	Remote procedure calls, web, mail servers.	CO10	L3

## 3. Mapping And Justification

CO - PO Mapping with mapping Level along with justification for each CO-PO pair.

To attain competency required (as defined in POs) in a specified area and the knowledge & ability required to accomplish it.

Mod ules	Mapping CO	Mapping PO	Mapping Level	Justification for each CO-PO pair	Lev el
-	<b>CO</b>	<b>PO</b>	-	<b>'Area': 'Competency' and 'Knowledge' for specified 'Accomplishment'</b>	-
1	CO1	PO1	3	Knowledge of POSIX limits is required to implement programs	L3
		PO2	3	Analyze programs in UNIX operating system using compile time & run time limits	L3
		PO3	3	Design new program using the knowledge of compile time & run time limits	L3
		P012	3	Learning in the context of technology changes in UNIX versions	L3
	CO2	PO1	2	Knowledge of API characteristics is required to implement different system calls	L2
		PO2	2	Analyze the different system calls using the knowledge of API characteristics	L2
		PO12	2	Learning in the context of technology changes in UNIX versions	L2
2	CO3	PO1	2	The knowledge of Unix file types are used to solve complex engineering problems.	L2

		PO12	3	Life long learning is required to explore new UNIX tools	L2
	CO4	PO1	3	The knowledge of Unix file APIs are used to solve complex engineering problems.	L3
		PO2	3	Analyze programs written using different file API's	L3
		PO3	3	Design new programs using the knowledge of file API's	L3
		PO12	3	Life long learning is required to explore new UNIX tools	L3
3	CO5	PO1	3	knowledge of APIs for Processes control is applied to solve complex engineering problems.	L4
		PO2	3	Analyze programs written for multiuser operating system	L4
		PO3	3	Design different programs using the knowledge of process API's	L4
		PO4	3	Investigate & interpretation of new programs can do done using process system calls	L4
		PO12	3	Learning in the context of technology changes in UNIX versions	L4
	CO6	PO1	3	Knowledge of process relationship is required to implement job control feature in multiuser environment	L4
		PO2	4	Analyze the relationship between parent & child process	L4
		PO3	4	Design programs which shows the relationship between parent & child process	L4
		PO4	4	Investigate different programs written to handle job control feature	L4
		PO12	4	Learning in the context of technology changes in UNIX versions	L4
4	CO7	PO1	4	Knowledge of interrupt & signals is required to perform interrupt handling	L3
		PO2	4	In order to identify, formulate and analyse engineering problems students make use of knowledge of signal and daemon Process functions.	L3
		PO3	4	Design programs to handle synchronous & asynchronous events	L3
		PO12	4	Learning in the context of technology changes in UNIX versions	L3
	CO8	PO1	4	The knowledge of the given specifications of daemon Process functions to demonstrate signal handling are analyzed are examined to solve complex engineering problems	L3
		PO2	4	Analyze different daemon coding rules	L3
		PO3	4	Design different programs using daemon coding rules	L3
		PO12	4	Learning in the context of technology changes in UNIX versions	L3
5	CO9	PO1	4	Knowledge of inter process communication is required to implement communication across different system	L4
		PO2	4	Design programs using different IPC mechanisms	L4
		PO3	4	In order to design solutions for complex engineering problems and design system components or processes for open ended engineering problems considering health and safety risks students can make use of Inter process communication using IPC Methods..	L4
		PO4	4	Investigation of inter process communication among different system. The knowledge of Inter process communication using IPC Methods are used to provide valid conclusions.	L4
		PO12	4	Learning in the context of technology changes in UNIX versions	L4
	CO10	PO1	3	Knowledge of pipes & sockets is required to implement client server communication	L3
		PO2	3	Analyze different client server communication	L3
		PO3	3	Design client server communication using IPC system calls	L3
		PO12	3	Learning in the context of technology changes in UNIX versions	L3

#### 4. Articulation Matrix

CO – PO Mapping with mapping level for each CO-PO pair, with course average attainment.

-	-	Course Outcomes	Program Outcomes															-	
			PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PS O1	PS O2	PS O3		
Mod ules	CO.#	<b>At the end of the course student should be able to ...</b>																	
1	15CS744.1	Use runtime & compile time limits in UNIX platform	2.4	2.4	2	0.6	-	-	-	-	-	-	-	-	2.4				L3

1	15CS744.2	Understand API characteristics using POSIX standard	2.4	2.4			-	-	-	-	-	-	-	2.4			L3
2	15CS744.3	Understand file structure in UNIX operating system.	2.4	2.4			-	-	-	-	-	-	-	2.4			L4
2	15CS744.4	Apply file manipulation system calls for different types of files.	2.4	2.4	2		-	-	-	-	-	-	-	2.4			L4
3	15CS744.5	Analyze process control primitives for different applications in multiuser environment	2.4	2.4	2	0.6	-	-	-	-	-	-	-	2.4			L4
3	15CS744.6	Identify relationship between group of processes for job control	2.4	2.4	2	0.6	-	-	-	-	-	-	-	2.4			L4
4	15CS744.7	Apply interrupt methods for handling asynchronous events	2.4	2.4	2		-	-	-	-	-	-	-	2.4			L4
4	15CS744.8	Understand daemon characteristics for coding rules	2.4	2.4	2		-	-	-	-	-	-	-	2.4			L4
5	15CS744.9	Distinguish message queues semaphores & shared memory across machine boundaries	2.4	2.4	2	0.6	-	-	-	-	-	-	-	2.4			L4
5	15CS744.10	Discover communication between client server using pipes & sockets	2.4	2.4	2		-	-	-	-	-	-	-	2.4			L4
-	<b>CS664PC</b>	<b>Average attainment (1, 2, or 3)</b>	<b>2.4</b>	<b>2.4</b>	<b>2</b>	<b>0.6</b>	-	-	-	-	-	-	-	<b>2.4</b>			<b>L2-L4</b>
-	<i>PO, PSO</i>	<i>1.Engineering Knowledge; 2.Problem Analysis; 3.Design / Development of Solutions; 4.Conduct Investigations of Complex Problems; 5.Modern Tool Usage; 6.The Engineer and Society; 7.Environment and Sustainability; 8.Ethics; 9.Individual and Teamwork; 10.Communication; 11.Project Management and Finance; 12.Life-long Learning; S1.Software Engineering; S2.Data Base Management; S3.Web Design</i>															

## 5. Curricular Gap and Content

Topics & contents not covered (from A.4), but essential for the course to address POs and PSOs.

Modules	Gap Topic	Actions Planned	Schedule Planned	Resources Person	PO Mapping
1	Unix features & architecture	Extra classes		Concerned faculty	
2	File locking mechanism	Extra classes		Concerned faculty	
5	Socket API's	Extra classes		Concerned faculty	

## 6. Content Beyond Syllabus

Topics & contents required (from A.5) not addressed, but help students for Placement, GATE, Higher Education, Entrepreneurship, etc.

Modules	Gap Topic	Area	Actions Planned	Schedule Planned	Resources Person	PO Mapping
1	Hands on Examples Programs using POSIX compile time & run time limits	placement	Planned for hands on session	Conducting hands on sessions(one hour per week)	Concerned faculty	
2	Hands on Examples Programs using UNIX & POSIX files	placement	Planned for hands on session	Conducting hands on sessions(one hour per week)	Concerned faculty	
3	Hands on Examples	placement	Planned for	Conducting	Concerned	



	Programs using UNIX process API's		hands on session	hands on sessions(one hour per week)	faculty	
4	Hands on Example Programs using signals	placement	Planned for hands on session	Conducting hands on sessions(one hour per week)	Concerned faculty	
5	Hands on Example programs using IPC	placement	Planned for hands on session	Conducting hands on sessions(one hour per week)	Concerned faculty	

## C. COURSE ASSESSMENT

### 1. Course Coverage

Assessment of learning outcomes for Internal and end semester evaluation. Distinct assignment for each student. 1 Assignment per chapter per student. 1 seminar per test per student.

Modules	Title	Teach. Hours	No. of question in Exam						CO	Levels
			CIA-1	CIA-2	CIA-3	Asg	Extra Asg	SEE		
1	Introduction	08	2	-	-	1	1	2	CO1, CO2	L3,L2
2	UNIX Files and APIs	08	2	-	-	1	1	2	CO3, CO4	L2, L3
3	UNIX Processes and Process Control	08	-	2	-	1	1	2	CO5, CO6	L4, L4
4	Signals and Daemon Processes	08	-	2	-	1	1	2	CO7, CO8	L3, L3
5	Interprocess Communication	08	-	-	4	1	1	2	CO9, CO10	L4, L3
-	<b>Total</b>	<b>40</b>	<b>4</b>	<b>4</b>	<b>4</b>	<b>5</b>	<b>5</b>	<b>10</b>	<b>-</b>	<b>-</b>

### 2. Continuous Internal Assessment (CIA)

Assessment of learning outcomes for Internal exams. Blooms Level in last column shall match with A.2.

Modules	Evaluation	Weightage in Marks	CO	Levels
1, 2	CIA Exam - 1	15	CO1, CO2, CO3, CO4	L3, L3, L3, L3
3, 4	CIA Exam - 2	15	CO5, CO6, CO7, CO8	L4, L4, L4, L4
5	CIA Exam - 3	15	CO9, CO10	L4, L4
1, 2	Assignment - 1	05	CO1, CO2, CO3, CO4	L3, L3, L3, L3
3, 4	Assignment - 2	05	CO5, CO6, CO7, CO8	L4, L4, L4, L4
5	Assignment - 3	05	CO9, CO10	L4, L4
1, 2	Seminar - 1		-	-
3, 4	Seminar - 2		-	-
5	Seminar - 3		-	-
1, 2	Quiz - 1		-	-
3, 4	Quiz - 2		-	-
5	Quiz - 3		-	-
1 - 5	Other Activities - UNIT TEST	-	CO9, CO10	L4, L4
	<b>Final CIA Marks</b>	<b>20</b>	<b>-</b>	<b>-</b>

## D1. TEACHING PLAN - 1

## Module - 1

Title:	Introduction	Appr Time:	8 Hrs
<b>a</b>	<b>Course Outcomes</b>	-	<b>Blooms Level</b>
-	The student should be able to:	-	
1	Use runtime & compile time limits in UNIX platform	CO1	L3
2	Understand API characteristics using POSIX standard	CO2	L2
<b>b</b>	<b>Course Schedule</b>	-	-
<b>Class No</b>	<b>Module Content Covered</b>	<b>CO</b>	<b>Level</b>
1	UNIX and ANSI Standards:	CO1	L2
2	The ANSI C Standard, The ANSI/ISO C++ Standards,	CO1	L2
3	Difference between ANSI C and C++	CO1	L2
4	The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards.	CO1	L3
5	UNIX and POSIX APIs:	CO2	L2
6	The POSIX APIs	CO2	L2
7	The UNIX and POSIX Development Environment,	CO2	L2
8	API Common Characteristics	CO2	L2
<b>c</b>	<b>Application Areas</b>	<b>CO</b>	<b>Level</b>
1	Implementation of programs to check limits in UNIX operating system	CO1	L3
2	Implementation of programs using test macros in UNIX operating system	CO1	L3
<b>d</b>	<b>Review Questions</b>	-	-
1	Bring out the importance of UNIX operating system.	CO1	L2
2	What is POSIX standard ,ANSI C standard , ANSI/ISO C++ standard?	CO1	L2
3	Difference between ANSI C & C++.	CO1	L2
4	Explain the different subsets of posix standard.	CO1	L2
5	API basic concepts.	CO2	L1
6	Define an API?	CO2	L1
7	General API characteristics?	CO2	L2
8	API error names & errno.	CO2	L2
9	What is errno?	CO2	L2
10	Describe the error status codes.	CO2	L1
		CO2	L2
<b>E</b>	<b>Experiences</b>	-	-
1		-	-
2			
3			
4		CO3	L3
5			

## Module - 2

Title:	UNIX Files and APIs	Appr Time:	8 Hrs
<b>A</b>	<b>Course Outcomes</b>	-	<b>Blooms Level</b>
-	The student should be able to:	-	
1	Understand file structure in UNIX operating system.	CO3	L2
2	Apply file manipulation system calls for different types of files.	CO4	L3
<b>B</b>	<b>Course Schedule</b>	-	-
<b>Class No</b>	<b>Module Content Covered</b>	<b>CO</b>	<b>Level</b>
09	File Types, The UNIX and POSIX File System,	CO3	L2
10	The UNIX and POSIX File Attributes, Inodes in UNIX System V,	CO3	L2

11	Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors,	CO3	L2
12	Directory Files, Hard and Symbolic Links.	CO3	L2
13	UNIX File APIs: General File APIs,	CO4	L3
14	File and Record Locking, Directory File API	CO4	L3
15	Device File APIs,	CO4	L3
16	FIFO File APIs, Symbolic Link File APIs.	CO4	L3
<b>C</b>	<b>Application Areas</b>	<b>CO</b>	<b>Level</b>
1	Organization of file in the file system of operating system	CO3	L2
2	Applying read and write lock on files.	CO4	L4
<b>D</b>	<b>Review Questions</b>	-	-
11	Define file. List & explain different types of file in UNIX operating system.	CO3	L2
12	Explain the concept of file attributes, inodes along with examples.	CO3	L2
13	When do we use API?	CO3	L2
14	Write the diagram for UNIX kernel support for files .	CO3	L2
15	Compare soft links & hard links.	CO4	L2
16	List and explain general file APIs with prototypes.	CO4	L3
17	Compare read lock & write lock with examples.	CO4	L3
18	What is a directory file API?	CO4	L2
19	Explain the sequence of events that occur when a process calls the close function to close an opened file?	CO3	L2
20	Advantages of locking the file?	CO4	L3
21	Why advisory lock is considered safe?what are the drawbacks of advisory lock?	CO4	L2
22	Explain symbolic link file APIs?	CO3	L2
<b>E</b>	<b>Experiences</b>	-	-
1			
2			
3			
4			
5			

## E1. CIA EXAM – 1

### a. Model Question Paper - 1

Crs Code:	15cs744	Sem:	VII	Marks:	30	Time:	75 minutes	
Course:	UNIX SYSTEM PROGRAMMING							
-	-	<b>Note: Answer any 2 questions, each carry equal marks.</b>				<b>Marks</b>	<b>CO</b>	<b>Level</b>
1	a	What is posix standard? Explain the different subset of posix standard.				6	CO1	L2
	b	List any 6 error status code along with its meanings				6	CO2	L2
	c	Differentiate between ANSI C and C++				3	CO1	L3
		<b>OR</b>						
2	a	Write a c or c++ program posix complement program to check following limits: i)number of clock ticks ii)Maximum number of child processes iii)Maximum path length				6	CO1	L3
	b	Write structure of program to filter out non-posix compliant codes from user program				5	CO1	L3
	c	Explain the common characteristics of API.				4	CO2	L2

3	a	Describe the UNIX Kernel support for files	5	CO3	L2
	b	Explain directory file and device file APIs?	5	CO4	L3
	c	Differentiate between the stream pointer and file descriptor?	5	CO3	L2
<b>OR</b>					
4	a	List the important uses of fcntl API. Give its prototype description	6	CO4	L3
	b	Explain the different file types available in UNIX or POSIX system.	5	CO3	L2
	c	Differentiate between soft link & hard link with examples.	4	CO4	L3

## b. Assignment -1

Note: A distinct assignment to be assigned to each student.

Model Assignment Questions							
Crs Code:	15CS744	Sem:	VII	Marks:	5 / 5	Time:	90 – 120 minutes
Course:	UNIX SYSTEM PROGRAMMING						
Note: Each student to answer 2-3 assignments. Each assignment carries equal mark.							
SNo	USN	Assignment Description	Marks	CO	Level		
1		Discuss the difference between ANSI C & C++.		CO1	L2		
2		What do you understand by the term feature test macros? List all the test macros along with its meanings.		CO1	L2		
3		Write C/C++ program to display POSIX version.		CO1	L3		
4		Mention any 5 compile time limits.		CO1	L2		
5		Differentiate between K & R C & ANSI C		CO1	L2		
6		Write a C/C++ POSIX compliant program to check the following limits. a) Number of child processes. b) Maximum path length c) Maximum path length d) Maximum number of open files per process.		CO1	L2		
7		Write a C/C++ program to emulate ln command in UNIX.		CO4	L3		
8		Write C or C++ program to check the following compile time along with its minimum value. a)supplemental groups b)maximum number of links of a file. c)maximum number of simulate nous asynchronous I/O. d)real signals		CO1	L3		
9		Explain different file types available in UNIX operating system.		CO3	L2		
10		List all the file attributes along with their meaning.		CO3	L2		
11		What is API?how it is different from C library functions?		CO3	L2		
12		Why calling an API is more time consuming than calling on user function.		CO3	L2		
13		Differentiate C stream pointer and file descriptor .		CO4	L3		
14		Differentiate symbolic link and hard link.		CO4	L2		
15		Which of the file attributes can't be changed and why?		CO4	L2		
16		List the commands needed to change the following attributes. i) file size ii) user ID iii) Last access & modification time iv) hard link count		CO3	L3		
17		What is inode to its file? Why are the inodes unique only within a file system? How does OS map the inode to its filename?		CO3	L3		
18		Explain UNIX kernel support for file with a neat diagram.		CO3	L2		
19		Explain the following general file APIs i)open() ii)fcntl() iii)l seek		CO4	L3		

--	--	--	--	--	--

## D2. TEACHING PLAN - 2

### Module – 3

<b>Title:</b>	Unix processes and process control	<b>Appr Time:</b>	<b>8 Hrs</b>
<b>A</b>	<b>Course Outcomes</b>	-	<b>Blooms Level</b>
-	The student should be able to:	-	
1	Analyze process control primitives for different applications in multiuser environment	CO5	L4
2	Identify relationship between group of processes for job control	CO6	L4
<b>b</b>	<b>Course Schedule</b>		
<b>Class No</b>	<b>Module Content Covered</b>	<b>CO</b>	<b>Level</b>
1	The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments,	CO5	L2
2	Environment List, Memory Layout of a C Program,	CO5	L3
3	Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions,	CO5	L3
4	UNIX Kernel Support for Processes. Process Control: Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions,	CO5	L4
5	Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection.	CO5	L3
6	Process Relationships: Introduction, Terminal Logins, Network Logins,	CO6	L4
7	Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions,	CO6	L3
8	Job Control, Shell Execution of Programs, Orphaned Process Groups.	CO6	L4
<b>c</b>	<b>Application Areas</b>	<b>CO</b>	<b>Level</b>
1	Create, resume, suspend & kill the process in multitasking environment.	CO5	L4
2	Switching among multiple jobs in multiuser & multitasking environment.	CO6	L4
<b>d</b>	<b>Review Questions</b>	-	-
1	Explain the following system calls: i)fork ii)vfork iii)exit iv)wait.	CO5	L4
2	Giving the prototype explain different variants of exec system call	CO5	L4
3	What is race condition? Write a program in C/C++ to illustrate a race condition.	CO5	L3
4	How UNIX operating system keeps process accounting?	CO5	L2
5	What is job control? Summarize the job control features with the help of a figure.	CO6	L4
6	With a neat block schematic, explain the terminal login process in BSD Unix. What is a session? Explain how you create a session using appropriate shell commands.	CO6	L4
7	Explain the following: i)wait ii)waitpid	CO5	L3
8	With a neat diagram, explain the memory layout of c program. In which segments are the automatic variables and dynamically created objects are stored?	CO5	L3
9	Write a short note on command-line arguments?	CO5	L2
10	Explain the three functions for memory allocation and alternate memory allocators?	CO5	L3
11	Explain getrlimit and setrlimit functions?	CO5	L3

12	Explain the data structure of parent and child processes after fork?	CO5	L3
<b>e</b>	<b>Experiences</b>	-	-
1			
2			
3			
4			
5			

## Module – 4

<b>Title:</b>	Signals and Daemon processes	<b>Appr Time:</b>	8 Hrs
<b>a</b>	<b>Course Outcomes</b>	-	<b>Blooms Level</b>
-	The student should be able to:	-	
1	Apply interrupt methods for handling asynchronous events	CO7	L3
2	Understand daemon characteristics for coding rules	CO8	L3
<b>b</b>	<b>Course Schedule</b>		
<b>Class No</b>	<b>Module Content Covered</b>	<b>CO</b>	<b>Level</b>
1	Signals: The UNIX Kernel Support for Signals,signal,	CO7	L3
2	Signal Mask, sigaction,	CO7	L3
3	the SIGCHLD Signal and the waitpid Function,	CO7	L3
4	The sigsetjmp and siglongjmp Functions,	CO7	L3
5	Kill, Alarm, Interval Timers, POSIX.1b Timers.	CO7	L3
6	Daemon Processes: Introduction,	CO8	L2
7	Daemon Characteristics, Coding Rules,	CO8	L3
8	Error Logging, Client-Server Model.	CO8	L3
<b>c</b>	<b>Application Areas</b>	<b>CO</b>	<b>Level</b>
1	Communication between program and operating system	CO8	L3
2	Operating system bootstrapping	CO7	L3
<b>d</b>	<b>Review Questions</b>	-	-
1	What is a signal? Discuss any five POSIX defined signals?	CO7	L2
2	What is a daemon? Discuss the basic coding rules.	CO8	L2
3	Explain the terms i)signal ii)signal mask	CO7	L2
4	What are daemon processes? Enlist their characteristics. Also write a program to transform a normal user process into a daemon process. Explain every step in the program.	CO8	L3
5	Briefly explain the kill() API and alarm() API?	CO7	L3
6	List the timer manipulation APIs in POSIX.1b	CO7	L3
7	Discuss daemon characteristics?	CO8	L2
8	Explain the coding rules for daemon process.	CO8	L2
9	What is error logging?with a neat block schematic discuss the error login facility in BSD.	CO8	L2
10	Explain the sigaction() function by giving the prototype and discuss its features?	CO7	L2
11	Briefly explain SIGCHLD Signal and the waitpid API?	CO7	L3
<b>e</b>	<b>Experiences</b>	-	-
1			
2			
3			
4			

5			
---	--	--	--

## E2. CIA EXAM – 2

### a. Model Question Paper - 2

Crs Code:	15CS744	Sem:	VII	Marks:	20	Time:	75 minutes	
Course:	UNIX SYSTEM PROGRAMMING							
-	-	<b>Note: Answer any 2 questions, each carry equal marks.</b>				<b>Marks</b>	<b>CO</b>	<b>Level</b>
1	a	Describe the UNIX Kernel support for process. Show the related data structures.				6	CO5	L3
	b	What is race condition? Write a program in C/C++ to illustrate a race condition.				5	CO5	L3
	c	Explain the following: i) network login ii) terminal login				4	CO6	L3
		OR						
2	a	What are the different ways in which a process can terminate? With a neat block schematic, explain how a process is launched and terminates clearly indicating the role of C-startup routine and the exit handlers.				8	CO5	L4
	b	What is job control? Summarize the job control features with the help of a figure.				7	CO6	L3
		OR						
3	a	What is a signal? Discuss any five POSIX defined signals?				5	CO7	L2
	b	What is a daemon? Discuss the basic coding rules.				5	CO8	L2
	c	Briefly explain SIGCHLD Signal and the waitpid API?				5	CO7	L3
		OR						
4	a	What are daemon processes? Enlist their characteristics.				6	CO8	L2
	b	Explain the sigaction() function by giving the prototype and discuss its features?				5	CO7	L3
	c	Discuss daemon characteristics?				4	CO8	L2

### b. Assignment – 2

Note: A distinct assignment to be assigned to each student.

<b>Model Assignment Questions</b>								
Crs Code:	15CS744	Sem:	VII	Marks:	5 / 5	Time:	90 – 120 minutes	
Course:	UNIX SYSTEM PROGRAMMING							
Note: Each student to answer 2-3 assignments. Each assignment carries equal mark.								
<b>SNo</b>	<b>USN</b>	<b>Assignment Description</b>				<b>Marks</b>	<b>CO</b>	<b>Level</b>
1		What is a signal? Discuss any five POSIX defined signals?				6	CO7	L2
2		What is a daemon? Discuss the basic coding rules.				5	CO8	L2
3		Explain the terms i)signal ii)signal mask				6	CO7	L3
4		What are daemon processes? Enlist their characteristics. Also write a program to transform a normal user process into a daemon process. Explain every step in the program.				8	CO8	L3
5		Explain the sigaction() function by giving the prototype and discuss its features?				5	CO7	L3
6		Briefly explain the kill() API and alarm() API?				6	CO7	L3
7		List the timer manipulation APIs in POSIX.1b				6	CO7	L3
8		What is error logging?with a neat block schematic discuss the error login facility in BSD.				7	CO8	L2
9		Briefly explain SIGCHLD Signal and the waitpid API?				6	CO7	L3
10		Discuss daemon characteristics?				3	CO8	L2
11		With an example explain the use of setjmp and longjmp functions				6	CO5	L3
12		Describe the UNIX Kernel support for process. Show the related data structures				8	CO5	L2



13		Bring out the importance of locking files. What is the drawback of advisory lock? Explain in brief.	8	CO6	L3
14		Explain the following system calls: i)fork ii)vfork iii)exit iv)wait.	9	CO5	L4
15		What is job control? Summarize the job control features with the help of a figure.	7	CO6	L3
16		How UNIX operating system keeps process accounting?	5	CO5	L3
17		What is race condition? Write a program in C/C++ to illustrate a race condition.	6	CO5	L3
18		Giving the prototype explain different variant of exec system call	6	CO5	L4

### D3. TEACHING PLAN - 3

#### Module - 5

<b>Title:</b>	Interprocess communication	<b>Appr Time:</b>	8 Hrs
<b>a</b>	<b>Course Outcomes</b>	-	<b>Blooms Level</b>
-	The student should be able to:	-	
1	Distinguish message queues semaphores & shared memory across machine boundaries	CO9	L4
2	Discover communication between client server using pipes & sockets	CO10	L3
<b>b</b>	<b>Course Schedule</b>		
<b>Class No</b>	<b>Module Content Covered</b>	<b>CO</b>	<b>Level</b>
1	Overview of IPC Methods, Pipes, popen,	CO9	L3
2	pclose Functions, Coprocesses,	CO9	L3
3	FIFOs, System V IPC, Message Queues	CO9	L4
4	Semaphores Shared Memory,	CO9	L4
5	Client-Server Properties,	CO10	L2
6	Stream Pipes, Passing File Descriptors,	CO10	L3
7	An Open Server-Version 1	CO10	L3
8	Client-Server Connection Functions	CO10	L3
<b>c</b>	<b>Application Areas</b>	<b>CO</b>	<b>Level</b>
1	Client server communication in distributed computing	CO10	L4
2	Remote procedure calls,web,mail servers.	CO9	L3
<b>d</b>	<b>Review Questions</b>	-	-
1	What are pipes? Write C++ program to send data from parent to child over a pipe.	CO9	L3
2	Write a program to implement popen and pclose system calls	CO9	L3
3	Explain the concept of shared memory with an example C/C++ program.	CO9	L3
4	Explain timing comparison of semaphores versus record locking?	CO9	L4
5	Explain the concept of shared memory with an example C/C++ program.	CO9	L3
6	Explain passing file descriptors over STREAMS-based pipes?	CO9	L3
7	What is a STREAMS-Based pipe?	CO10	L2
8	Explain how to setup connd to make unique connections?	CO10	L3
9	What is a socket? Discuss how to it create and destroy a socket?	CO10	L3
<b>e</b>	<b>Experiences</b>	-	-
1			
2			
3			
4			
5			



### E3. CIA EXAM – 3

#### a. Model Question Paper - 3

Crs Code:	15CS744	Sem:	VII	Marks:	30	Time:	75 minutes	
Course:	Unix system programming							
-	-	<b>Note: Answer any 2 questions, each carry equal marks.</b>				<b>Marks</b>	<b>CO</b>	<b>Level</b>
1	a	What are pipes? What are their limitations? Write a c program that sends "hello world" message to the child process through the pipe. The child on receiving this message should display it on the standard output.				6	CO9	L3
	b	What is a STREAMS-Based pipe?				5	CO10	L2
	c	Discuss the applications of FIFO.				4	CO9	L2
<b>OR</b>								
2	a	Explain different APIs used with message queues?				7	CO9	L4
	b	What do you mean by passing file descriptors between processes? Explain.				5	CO10	L2
	c	What is a socket? Discuss how it create and destroy a socket?				5	CO10	L3
	d							
3	a	Explain the concept of shared memory with an example C/C++ program				7	CO9	L3
	b	Explain how to setup conlld to make unique connections?				5	CO10	L4
	c	What are semaphores. What is their purpose.				3	CO9	L
	d							L2
<b>OR</b>								
4	a	Explain the different client server connections functions with examples				8	CO10	L3
	b	Explain passing file descriptors over STRAMS-based pipes?				7	CO9	L3
	c							
	d							

#### b. Assignment – 3

Note: A distinct assignment to be assigned to each student.

<b>Model Assignment Questions</b>								
Crs Code:	15CS744	Sem:	VII	Marks:	5 / 5	Time:	90 – 120 minutes	
Course:	UNIX SYSTEM PROGRAMMING							
Note: Each student to answer 2-3 assignments. Each assignment carries equal mark.								
<b>SNo</b>	<b>USN</b>	<b>Assignment Description</b>				<b>Marks</b>	<b>CO</b>	<b>Level</b>
1		With a neat block schematic, explain how FIFO can be used to implement client-server communication model.				6	CO9	L2
2		What are the three different ways in which the client and server processes are can get access to same IPC structures? List the APIs with their argument details that are used to create, control, send and receive messages from a message queue.				9	CO9	L4
3		What are semaphores? What is their purpose? List and explain the APIs used to create and control the semaphores.				7	CO9	L4
4		What are the different system calls available to create and manipulate semaphores? Explain.				6	CO9	L3
5		What are the limitations of pipes.				4	CO9	L2
6		What are the different system calls available to create and manipulate semaphores? Explain.				6	CO9	L3
7		Write a short note on message queues?				5	CO9	L2
8		Explain different APIs used with message queues?				5	CO9	L2
9		Discuss the applications of FIFO				4	CO9	L2
10		What are the advantages and disadvantages of IPC?				6	CO9	L2
11		What are pipes? What are their limitations? Write a c				8	CO9	L3

		program that sends "hello world" message to the child process through the pipe. The child on receiving this message should display it on the standard output.			
12		With a neat block schematic, explain how FIFO can be used to implement client-server communication model.	8	CO9	L3
13		Write a short notes on client sever properties.	5	CO10	L2
14		What do you mean by passing file descriptors between processes?	6	CO10	L3
15		What is a STREAMS-Based pipe?	6	CO10	L3
16		Explain open server,version 1?	5	CO10	L2

## F. EXAM PREPARATION

### 1. University Model Question Paper

Course:	Unix system programming				Month / Year	Dec /2018		
Crs Code:	15Cs744	Sem:	VII	Marks:	80	Time:	180 minutes	
-	<b>Note</b>	Answer all FIVE full questions. All questions carry equal marks.				<b>Marks</b>	<b>CO</b>	<b>Level</b>
1	a	Write a c++ program to list the actual values of the following system configuration limits on a given UNIX OS. i) Maximum no. of child processes that can be created. ii) Maximum no. of files that can be opened simultaneously. iii) Maximum no. of message queues that can be accessed.				7	CO1	L3
	b	Write C++ program to display POSIX VERSION.				3	CO1	L3
	c	List any six values of the global variable errno along with their meanings.				6	CO2	L2
		<b>OR</b>						
-	a	List the differences between ANSI C and K & R Explain				6	CO1	L2
	b	Write a c++ program to list the actual values of the following system configuration limits on a given unix OS. i)Maximum no. of child processes that can be created. ii)Maximum no. of files that can be opened simultaneously. iii)Maximum no. of message queues that can be accessed.				7	CO1	L3
	c	Explain the common characteristics of API.				3	CO1	L2
		<b>OR</b>						
2	a	Explain the different file types available in UNIX or POSIX system.				5	CO3	L2
	b	Describe the UNIX Kernel support for files with neat diagram.				6	CO3	L2
	c	Explain directory file and device file APIs?				5	CO4	L3
		<b>OR</b>						
-	a	What are APIs? When do you use them? Why are the API more time consuming than the library function?				5	CO4	L3
	b	List all the file attributes along with their meanings. Which of these attributes can t be changed and why? List the commands needed to change the following file attributes. i) file size;ii) User ID; iii) Last access and modification time; iv) hard link count.				7	CO3	L2
	c	Write a program to implement ls -l command				4	CO4	L3
		<b>OR</b>						
3	a	Write an explanatory note on environment variables. Also write a C/C++ program that outputs the contents of its environment list.				5	CO5	L4
	b	Describe the UNIX Kernel support for process. Show the related data structures				6	CO5	L2
	c	What is race condition? Mention & explain routines to avoid race condition.				5	CO5	L3
	d							L3

		<b>OR</b>			
-	a	With a neat block schematic, explain the terminal login/network login process in BSD Unix. What is a session? Explain how you create a session using appropriate shell commands.	6	CO6	L2
	b	Explain how the shells execute programs?	4	CO6	L2
	c	Explain the following system calls: i)fork ii)vfork iii)exit iv)wait	6	CO5	L4
4	a	What is a signal? Discuss any five POSIX defined signals?	6	CO7	L2
	b	Explain the terms i)signal ii)signal mask	2	CO7	L2
	c	What are daemon processes? Enlist their characteristics. Also write a program to transform a normal user process into a daemon process. Explain every step in the program.	8	CO8	L3
		<b>OR</b>			
-	a	What is error logging?with a neat block schematic discuss the error login facility in BSD.	7	CO8	L2
	b	Briefly explain the kill() API and alarm() API?	4	CO7	L3
	c	Mention the different sources of signals. Write a program to setup signals handlers for SIGINT & SIGACRAM signals.	5	CO7	L2
					L3
					L3
5	a	What are pipes? What are their limitations? Write a C/C++ program that sends "hello world" message to the child process through the pipe. The child on receiving this message should display it on the standard output.	8	CO9	L3
	b	Explain STREAMS-Based pipe?	5	CO10	L2
	c	Define the following: i) semaphores ii) message queues iii) shared memory	3	CO9	L1
	d				
		<b>OR</b>			
	a	Which is the fastest form of IPC? Explain.	5	CO9	L4
	b	Explain client server communication using FIFO with neat diagram.	6	CO10	L3
	c	Briefly explain the client server functions.	5	CO10	L3
	d				

## 2. SEE Important Questions

Course:	Unix system programming			Month / Year	Dec /2018		
Crs Code:	15Cs744	Sem:	VII	Marks:	80	Time:	180 minutes
	<b>Note</b>	Answer all FIVE full questions. All questions carry equal marks.				-	-
Mo dule	Qno.	Important Question	Marks	CO	Year		
1	1	Bring out the importance of standardizing the UNIX operating system. List the differences between ANSI C and K & R C.	8	CO1	2009		
	2	List all the five feature test macros along with their meanings.	6	CO1	2009		
	3	What are the API common characteristics? List any six values of the global variable errno along with their meanings.	10	CO2	2016		
	4	Explain POSIX standards with different subsets of POSIX. Write C++ program to display POSIX VERSION.	8	CO1	2017		
	5	Write a c++ program to list the actual values of the following system configuration limits on a given UNIX OS. i) Maximum no. of child processes that can be created. ii) Maximum no. of files that can be opened simultaneously. iii) Maximum no. of message queues that can be accessed.	7	CO1	2017		
	6	Write structure of program to filter out non-posix compliant codes from user program	5	CO1	2017		
2	1	Explain the commands to create different file types supported by UNIX.	6	CO3	2016		

	2	Explain UNIX kernel support for files with neat diagram.	8	CO3	2016
	3	Explain explain the prototype of the following APIs i) open ii) lseek iii) fsat iv) chmod	8	CO3	2018
	4	What is the advantage of locking files? Explain mandatory & advisory locks? Why advisory lock is considered safe? What are the drawbacks of advisory lock?	7	CO4	2017
	5	Explain symbolic link file APIs?	8	CO4	2016
3	1	Write an explanatory note on environment variables. Also write a C/C++ program that Outputs the contents of its environment list.	6	CO5	2009
	2	What is race condition? Write a program for generating race condition?	8	CO5	2016
	3	Explain in detail the family of exec functions.	12	CO5	2016
	4	Explain the memory layout of C program with neat diagram?	7	CO5	2018
	5	Explain fork and vfork system calls. How fork call differs from vfork? Write program to demonstrate fork & vfork system calls.	10	CO5	2017
	6	What is job control? What are the three forms of support from operating system required for job control	4	CO6	2017
4	1	What is signal? Discuss any five POSIX defined signals. Explain how to set up a signal handler.	10	CO7	2017
	2	Explain program how to setup a signal handler.	6	CO7	2017
	3	Write C/C++ program to show the use of alarm.	6	CO7	2016
	4	What is daemon process? Explain daemon characteristics & relation to session & process groups.	10	CO8	2016
	5	Explain coding coding rules for daemon process?	5	CO8	2016
5	1	What are three different ways in which client sever process can get access to same IPC structure? Explain different prototype of APIs that support these structure	10	CO9	2018
	2	What is FIFO? Explain how it is used in IPC. Discuss with an example, the client server communication using FIFOs.	10	CO9	2017
	3	What are pipes? Write a C++ program to send data from parent to child over a pipe.	10	CO9	2016
	4	Briefly explain client server functions.	6	CO10	2016
	5	What are stream pipes? Explain passing file descriptors.	7	CO10	2015
	6	What is a socket. Describe the socket API.	5	CO10	20

## G. Content to Course Outcomes

### 1. TLPA Parameters

**Table 1: TLPA – 15CS744**

Module-#	Course Content or Syllabus (Split module content into 2 parts which have similar concepts)	Content Teaching Hours	Blooms' Learning Levels for Content	Final Blooms' Level	Identified Action Verbs for Learning	Instruction on Methods for Learning	Assessment Methods to Measure Learning
A	B	C	D	E	F	G	H
1	The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++,The POSIX Standards,The POSIX.1 FIPS Standard, The X/Open Standards.	05	- L1 - L2 -L3	L3	understand Demonstrate	Demonstrate programs	Student presentation of programs
1	UNIX and POSIX APIs:The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.	03	- L1 - L2	L2	understand	Demonstrate programs	Student presentation of programs
2	File Types, The UNIX and POSIX File System,UNIX and POSIX File Attributes,	04	- L1 - L2	L2	Understand	Reading, discussi	Question & answers

	Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files. Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.					on Hands on sessions	Quiz
2	General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs.	04	- L1 - L2 -L3	L3	Understand and Demonstrate	Reading, discussion Hands on sessions	Student presentation of programs
3	Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions. UNIX Kernel Support for Processes. Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group Process Accounting, User Identification, Process Times, I/O Redirection.	05	- L1 - L2 -L3 -L4	L4	Apply Analyze	Presentation Hands on sessions	Question and answers assignment
3	Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups.	03	- L2 - L3	L4	Understand and Implement	Presentation Hands on sessions	Question and answers assignment
4	Introduction, Daemon Characteristics, Coding Rules, error Logging, Client-Server Model. The UNIX Kernel Support for signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lbtimers.	05	- L2 - L3 -	L3	Apply	Demonstrate programs Hands on sessions	Student presentation Quiz
4	Daemon Characteristics, Coding Rules, error Logging, Client-Server Model.	03	- L2 - L3	L3	Apply	Demonstrate programs Hands on sessions	Student presentation Quiz
5	Overview of IPC Methods, Pipes, popen, pclose Functions, Co processes, FIFOs, System V IPC, Message Queues, Semaphores. Shared Memory	04	- L2 - L3 -L4	L4	Apply Examine	Demonstrate programs Hands on sessions	Student presentation of programs Quiz
5	Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server-Version 1, Client-Server Connection Functions.	04	- L2 - L3	L3	Apply	Demonstrate programs Hands on sessions	Student presentation of programs Quiz

## 2. Concepts and Outcomes:

**Table 1: Concept to Outcome – 15CS744**

Module #	Learning or Outcome from study of the Content or Syllabus	Identified Concepts from Content	Final Concept	Concept Justification (What all Learning Happened from the study of Content / Syllabus. A short word for learning or outcome)	CO Components (1.Action Verb, 2.Knowledge, 3.Condition / Methodology, 4.Benchmark)	Course Outcome  <b>Student Should be able to ...</b>
<i>A</i>	<i>I</i>	<i>J</i>	<i>K</i>	<i>L</i>	<i>M</i>	<i>N</i>
1	-UNIX & ANSI standards - ANSI C standards -ANSI/ISO C++ standards -POSIX.1 FIPS standard -POSIX standard	-POSIX standards -ANSI standards	POSIX standards	Implement simple programs using POSIX runtime & compile time limits	Demonstrate compile time & run time limits UNIX operating system	Use runtime & compile time limits in UNIX platform
1	-POSIX API's -API common characteristics	-API characteristics	API characteristics	Understand API characteristics	Understand API characteristics POSIX & UNIX standards	Understand API characteristics using POSIX standard
2	-file types - UNIX file system - file attributes -inodes in unix system -API to files -kernel support for files	-files -file attributes -inodes in kernel -file API's	Kernel support for files	Understand kernel support for files in unix operating system	Understand file API's POSIX standard	Understand file structure in UNIX operating system.
2	-General file API's -file & record locking -hard link & soft link	-file API's -file locks	API for file	Application program interface for files	Demonstrate Different file API's Unix operating system POSIX standard	Apply file manipulation system calls for different types of files.
3	-unix process environment -main processes -process termination -memory layout of C program -kernel support for process	-unix processes -process control	Process management	Analyze different process control API's	Analyze process control API's UNIX environment	Analyze process control primitives for different applications in multiuser environment
3	-terminal login -network login	-process relationships	Process relationship	Analyze the relationship between process	Analyze process relationships UNIX environment	Identify relationship between group of processes for job control

	-process groups -session -job control					
4	-unix kernel support for signals -signal mask -sigaction -kill -alarm	-signals -kernel support for handling signals	Signal handling techniques	Apply different signal handling API's to handle signals	Apply signal API's UNIX environment	Apply interrupt methods for handling asynchronous events
4	-Daemon characteristics -coding rules -error logging -client server model	-daemon characteristics	Daemon characteristics	Understand daemon characteristics	Understand characteristics UNIX environment	Understand daemon characteristics for coding rules
5	-IPC methods -pipes -popen -pclose -co processes	-IPC methods -co processes	IPC mechanisms	Inter processes communication techniques	Distinguish IPC mechanism UNIX environment	Distinguish message queues semaphores & shared memory across machine
	-client server properties -stream pipes -client server connection function	-client server communication	Client server communication	Client server communication API's	Discover client server communication UNIX environment	Discover communication between client server using pipes & sockets