

		2. UDP Client, UDP Server		
VI	Case study on designing network topology	A case study to design and configure any organization network eg. College network or campus network, using any packet tracer or network topology design software based on infrastructure requirements, servers and clients, traffic consideration and application requirements.	06	L06

Text Books:

1. Computer Network: Top Down approach, Behrouz Forouzan, Firoz Mossharraf. MGH
2. Packet analysis with Wire shark, Anish Nath, PACKT publishing

Reference Books:

1. NS2.34 Manual
2. Introduction to Network Simulator NS2, 2nd Edition, Teerawat Issariyakul, Ekram Hossain, Springer

Term Work:

Term Work shall consist of at least 10 to 12 practical's based on the above list. Also Term work Journal must include at least 2 assignments.

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments) + 5 Marks (Attendance)

Oral Exam: An Oral exam will be held based on the above syllabus.

Course Code	Course Name	Theory	Practical	Tutorial	Theory	TW/Practical	Tutorial	Total
ITL402	Unix Lab	--	2	--	--	1	--	1

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Oral & Practical	Total
		Internal assessment			End Sem. Exam			
		Test 1	Test 2	Avg. of two Tests				

ITL402	Unix Lab	--	--	--	--	25	25	50
--------	----------	----	----	----	----	----	----	----

Lab Objectives: Students will try:

1. To introduce Basic Unix general purpose Commands
2. To learn network Unix commands.
3. To learn C programming in Unix editor environment.
4. To learn shell script and sed concepts.
5. To learn file management and permission advance commands.
6. To learn awk, grap, perl scripts.

Lab Outcomes: Student will be able to:

1. Identify the basic Unix general purpose commands.
2. Apply and change the ownership and file permissions using advance Unix commands.
3. Use the awk, grep, perl scripts.
4. Implement shell scripts and sed.
5. Apply basic of administrative task.
6. Apply networking Unix commands.

Prerequisite: C Programming Language and Operating System

Hardware requirement:

PC i3 and above.

Software requirement:

Unix, Editor, Bash shell, Bourne shell and C shell.

Detailed syllabus:

Sr. No.	Module	Detailed Content	Hours	LO Mapping
0	Prerequisite	Programming syntax, Installation of Unix, concepts of operating system	02	--
I	Basic Commands	A brief history of UNIX, Unix Architecture. Logging into (and out of) UNIX systems, Changing your password, General format of UNIX commands. a) Installation of Unix operating system. b) User management in Unix. c) Study of Unix general purpose	04	LO1

		utility command list obtained from (cd, cp, ps, ls, mv, rm, mkdir, rmdir, man, who, cat, echo, more, date, time, kill, history, chmod, passwd, who am i, who, time, bc, history, clear, man, lost, chown, finger, pwd, cal, logout, shutdown) commands.		
II	Advance Commands	<p>a) Study of Unix networking commands (ifconfig, ping, traceroute, netstat, nslookup, whois, hostname, tcpdump).</p> <p>b) Study of Unix file system (tree structure).</p> <p>c) Study of .bashrc, /etc/bashrc and Environment variables.</p> <p>d) Study File and directory permissions.</p> <p>e) Study of Editor Vi/other editor.</p> <p>f) Study of Bash shell, Bourne shell and C shell in Unix operating system.</p>	04	LO1 LO2 LO5 LO6
III	Basic System administrative task	<p>Process management</p> <p>Memory management</p> <p>File system management</p> <p>User management</p>	04	LO1 LO2 LO5
IV	Shell scripts	<p>a) Write a shell script program to display list of user currently logged in.</p> <p>b) Write a shell script program to display “HELLO WORLD”.</p> <p>c) Write a shell script program to develop a scientific calculator.</p> <p>d) Write a shell Script program to check whether the given number is</p>	04	LO1 LO4

		<p>even or odd.</p> <p>e) Shell script Program to search whether element is present is in the list or not.</p>		
V	Shell scripts and sed	<p>a) Shell script program to check whether given file is a directory or not.</p> <p>b) Shell script program to count number of files in a Directory.</p> <p>c) Shell script program to copy contents of one file to another.</p> <p>d) Create directory, write contents on that and Copy to a suitable location in your home directory.</p> <p>e) Use a pipeline and command substitution to set the length of a line in file to a variable.</p> <p>f) Write a program using sed command to print duplicated lines of Input.</p>	06	<p>LO1</p> <p>LO4</p>
VI	grep, awk, perl scripts	<p>a) Write a grep/egrep script to find the number of words character, words and lines in a file.</p> <p>b) Write an awk script to develop a Fibonacci series.</p> <p>c) Write a perl script to compute the power of a given number.</p> <p>d) Write an awk script to display the pattern of given string or number.</p> <p>e) Write a perl script to check a number is prime or not.</p> <p>f) Write an egrep script to display</p>	04	<p>LO1</p> <p>LO2</p> <p>LO3</p>

		list of files in the directory.		
--	--	---------------------------------	--	--

Text Books:

1. Unix, concepts and applications by Sumitabha Das, McGraw-Hill
2. Mastering Shell Scripting, Randal. K. Michael , Second Edition, Wiley Publication

References:

1. Unix Shell Programming by Yashwant Kanetkar
2. Unix shell programming by forozun

Term Work:

Term Work shall consist of at least 10 to 12 practical's based on the above list. Also Term work Journal must include at least 2 assignments.

Term Work Marks: 25 Marks (Total marks) = 15 Marks (Experiment) + 5 Marks (Assignments) + 5 Marks (Attendance)

Oral & Practical Exam: An Oral & Practical exam will be held based on the above syllabus.

Course Code	Course Name	Theory	Practical	Tutorial	Theory	TW/Practical	Tutorial	Total
ITL403	Microprocessor Programming Lab	--	2	--	--	1	--	1

Course Code	Course Name	Examination Scheme						
		Theory Marks				Term Work	Oral	Total
		Internal assessment			End Sem. Exam			
		Test1	Test 2	Avg. of two Tests				
ITL403	Microprocessor Programming Lab	--	--	--	--	25	25	50

Lab Objectives: Students will try to:

1. Learn assembling and disassembling of PC.
2. Get hands on experience with Assembly Language Programming.
3. Study interfacing of peripheral devices with 8086 microprocessor.
4. Understand techniques for faster execution of instructions and improve speed of operation and performance of microprocessors.
5. Learn fundamentals of designing embedded systems
6. Write and debug programs in TASM/MASM/hardware kits

Lab Outcomes: Students will be able to :