

OS LAB DSE 3161

CONTENTS

LAB NO.	TITLE	PAGE NO.	REMARKS
	COURSE OBJECTIVES AND OUTCOMES	i	
	EVALUATION PLAN	ii	
	INSTRUCTIONS TO THE STUDENTS	iii	
1	BASIC LINUX COMMANDS	1	
2	SHELL SCRIPTING – 1	24	
3	SHELL SCRIPTING – 2	31	
4	LINUX SYSTEM CALLS	39	
5	THREAD PROGRAMMING	53	
6	CPU SCHEDULING ALGORITHMS	59	
7	INTERPROCESS COMMUNICATION	63	
8	PROCESS SYNCHRONIZATION	83	
9	DEADLOCK ALGORITHMS	92	
10	MEMORY MANAGEMENT	98	
11	PAGE REPLACEMENT ALGORITHMS	103	
12	DISK SCHEDULING ALGORITHMS	107	
	REFERENCES	112	

Objectives

- Illustrate and explore the basic commands, shell scripting and system calls related to Linux operating system.
- Learn process management concepts which include scheduling algorithms, and inter process communication.
- Understand the working of memory management schemes, disk scheduling algorithms, and page replacement algorithms through simulation.

Course Outcomes

At the end of this course, students will have the

- Ability to execute Linux commands, shell scripting using appropriate Linux system calls.
- Ability to design thread programming, simulate process management and inter process communication techniques.
- Ability to implement the memory management, disk scheduling and page replacement algorithms.

Evaluation Plan

- Internal Assessment Marks: 60%
 - ✓ Continuous evaluation component (for each experiment)
 - ✓ The assessment will depend on punctuality, program execution, maintaining the observation note and answering the questions in viva voce
 - ✓ Total marks of the 12 experiments will sum up to 60
- End semester assessment of 2 hour duration: 40 Marks

INSTRUCTIONS TO THE STUDENTS

Pre-Lab Session Instructions

1. Carry Class notes, Lab Manual and the required stationery to every lab session.
2. Be in time.
3. Sign in the log register provided.
4. Adhere to the rules and maintain lab decorum.

In-Lab Session Instructions

- Work on allotted exercises given in Lab Manual.
- Prescribed textbooks and class notes can be kept ready for reference if required.

General Instructions for the exercises in Lab

- Shell scripts and programs should meet the following criteria:
 - Shell scripts and programs should be interactive with appropriate prompt messages, error messages if any, and descriptive messages for outputs.
 - Shell scripts and programs are **properly indented and comments** should be given whenever it is required.
- In case a student misses a lab class, he/she must ensure that the experiment is completed.
- Place mobile phones face down and on silent.

WEEK 1

BASIC LINUX and Vi COMMANDS

Objectives:

- In this lab, student will be able to:
 - Understand and execute basic commands.

Lab Exercises:

1. Basic Linux Commands: Test the function of each of the following:

Online help: man, help

Who, whoami, more, less, head, tail

Standard output: echo

Date

clear

Directory related commands: pwd, mkdir, cd, rmdir, mvdir

File related commands: ls, cat, more, cp, mv, rm, touch

2. Write shell commands for the following.
 - i) Create 11 subdirectories in your home directory under OSxxx.
 - ii) Create subdirectory **TestFolder** under **Week1**
 - iii) Create 3 different empty files in the TestFolder.
 - iv) Add different contents in each of them using *vi editor* (say *f1*, *f2* and *f3*)
 - v) Move back to Week1 folder and copy *f1* to this location.
 - vi) Create one more file with the command listing the **number of users** and **number of files**. Check content of this file.
 - vii) Use command to list all the files which starts with either 'a' or 'A'.
 - viii) Create a newfile called **month_names** and use sort command to sort them.
 - ix) Display the first five and last five lines of a given file.
 - x) Redirect the output of commands 'pwd', 'date' and 'ls' in succession to a file.
 - xi) Exercise for vi editor commands: (*Refer to next page*)

Looking into the Linux kernel

The core of the Linux system is the kernel. The kernel controls all of the hardware and software on the computer system, allocating hardware when necessary, and executing software when required.

If you've been following the Linux world at all, no doubt you've heard the name Linus Torvalds. Linus is the person responsible for creating the first Linux kernel software while he was a student at the University of Helsinki. He intended it to be a copy of the Unix system, at the time a popular operating system used at many universities.

After developing the Linux kernel, Linus released it to the Internet community and solicited suggestions for improving it. This simple process started a revolution in the world of computer operating systems.

Soon Linus was receiving suggestions from students as well as professional programmers from around the world.

For the above exercise (**Ex. xi**) perform the following editor operations:

- a) Remove blank lines wherever the *** is found
- b) Replace all occurrences of the word "Linux" with "Ubuntu"
- c) Save and return to command line prompt
- d) Use command to view "Linux.txt" at the command prompt without using vi
- e) Rename "Linux.txt" with the name "Ubuntu.txt"
- f) Make a copy of the same text file in the name "Linux1.txt" at command prompt