

**MC213 UNIX PROGRAMMING LAB**

**Objective of the Course:**

- Enable students to learn various unix utilities and shell scripting

**Recommended Systems/Software Requirements:**

- Intel based desktop PC with minimum of 166 MHZ or faster processor with atleast 64 MB RAM and 100 MB free disk space LAN Connected
- Any flavour of Unix / Linux

**Session-1**

- a)Log into the system
- b)Use vi editor to create a file called myfile.txt which contains some text.
- c)correct typing errors during creation.
- d)Save the file
- e)logout of the system

**Session-2**

- a)Log into the system
- b)open the file created in session 1(vi,cat,touch)
- c)Add some text (cp,mv,rm,mkdir,rmdir,ls)
- d)Change some text
- e>Delete some text
- f)Save the Changes
- g)Logout of the system

**session-3**

Filters:(Text processing utilities)

Wc,od,cmp,comm.,diff,head,tail,cut,paste,sort,grep,uniq Disk&backup utilities

Du,df,tar,cpio,ps,who

**session-4**

1. Write a shell script to generate a multiplication table.
2. Write a shell script that copies multiple files to a directory.
3. Write a shell script which counts the number of lines and words present in a given file.
4. Write a shell script which displays the list of all files in the given directory.
5. Write a shell script(small calculator) that adds, subtracts, multiplies and divides the given two integers. There are two division options: one returns the quotient and the other returns remainder. The script requires 3 arguments: The operation to be used and two integer numbers. The options are add(-a), subtract(-s), multiply(-m), quotient(-c) and remainder(-r).
6. Write a shell script to reverse the rows and columns of a matrix

### **Session-5**

1. Write a C program that counts the number of blanks in a text file. using standard I/O using system calls. Imp
  - a) using standard I/O
  - b) using system calls.
2. Implement in C the following Unix commands using system calls.
  - a) cat
  - b) ls
  - c) mv
3. Write a program that takes one or more file/directory names as command line input and reports the following information on the file:
  - a) File type.
  - b) Number of links.
  - c) Time of last access.
  - d) Read, Write and Execute permissions.
4. Write a C program that illustrates how to execute two commands concurrently with a command pipe.
5. Write a C program that illustrates the creation of child process using fork system call.
6. Write a C program that displays the real time of a day every 60 seconds.
7. Write a C program that illustrates file locking using semaphores.
8. Write a C program that implements a producer-consumer system with two processes.(using semaphores)
9. Write a C program that illustrates inter process communication using shared memory system calls.
10. Write a C program that illustrates the following:
  - a) Creating a message queue.
  - b) Writing to a message queue.
  - c) Reading from a message queue

### **Text Books:**

1. Unix and shell Programming Behrouz A. Forouzan, Richard F. Gilberg.Thomson
2. Advanced Programming in the UNIX environment W.R.Stevens

### **Reference Books:**

1. Unix internals,the new frontiers Uresh vahalia.
2. The C Odyssey UNIX Meeta Gandhi