IV. UNIX AND SHELL PROGRAMMING LAB SYLLABUS

UNIX AND SHELL PROGRAMMING LABORATORY	
Course Code: CIL37	**Credits:** 0:0:1
Pre – requisites: Nil	**Contact Hours:** 14P
Course Coordinator: Shubha Vibhu Malige	

COURSE CONTENTS

List of Lab Programs

Vi/Vim editor operations

1. Introduction to Unix and Shelling Programming, Vi/Vim editor:

Consider the following content and edit using Vi/Vim editor:

"The simplest way to understand how AI and ML relate to each other is: AI is the broader concept of enabling a machine or system to sense, reason, act, or adapt like a human. ML is an application of AI that allows machines to extract knowledge from data and learn from it autonomously.

Artificial Learning is the capability of a computer system to mimic human cognitive functions such as learning and problem-solving. Through AI, a computer system uses math and logic to simulate the reasoning that people use to learn from new information and make decisions

Cyber security is primarily about people, processes, and technologies working together to encompass the full range of threat reduction, vulnerability reduction, deterrence, international engagement, incident response, resiliency, and recovery policies and activities, including computer network operations, information assurance, law enforcement, etc."

a. Execute the following operations:

i. Open the file in vi/vim editor and type the above given content and save the file by name test1.txt, continue working in the opened file.
ii. Navigate through the 10^{th} line and go to save mode and write the 10^{th} line to another file named test2.txt and continue working with test1.txt
iii. Find the Keyword AI and replace AI with Artificial Learning using interactive substitution and save the current file.
iv. Delete 5^{th} line to a buffer (a), save the current file. Now switch to the file test2.txt, move to a desired location and copy the text. Toggle between previous file and current file.

v. Navigate to the first line in test.txt file, and go to 15th character towards right and move 3 lines down and 2 characters left and replace the character with M and save the current file and exit.

vi. In the test2.txt file, move to 3 words forward in the current line and take the cursor to 2 words back, delete the word in the cursor position and navigate yourself to move to the line extremes, save and escape to the shell.

vii. In the test1.txt, join first 3 lines (Join the current line with 2 lines).

viii. Delete the 7th line in current file and exit the vi editor by saving and quitting.

b. Correct the below given c program in Vi/Vim editor using vi commands and execute the corrected c program.

Sample.c(Before Correction—with errors)	Sample.c(After Correction--Right)

#include<stio.h>
#include errno.h
int test(int * message)
{
print("Errno is %8d',errno)
exit;
	#include<stdio.h>
	#include <errno.h>
	void test(char * message)
	{
	print("Errno is %8d\n",errno);
	exit(1);
	}

Shell Scripting

2. Consider the below scenarios and execute the given shell scripts.

"*Ramaiah College is having 10 departments (Say, CS, IS, AI, ML, Cyber Security, EC, Mechanical, EEE, DS, Civil) with UG and PG programs, and in each of the program student details, course details are maintained in 10 different files (such as Student Details, Course details, Curriculum, Exam, Marks, Research Activity, NBA, Placement Activities, Library Details, Extra Activities….)*."

a. Develop a shell script for the above scenario to create 10 levels of folders for the departments and inside each level(department) of folder, create 10 files in each department for maintaining student details. Display the entire hierarchy on the standard output by using tree command.

b. Develop a shell script that accepts above created filename as argument and dis-

play its creation time and permissions of the file, on the standard output.

3. a. Develop a shell script that takes a valid directory name as an argument and recursively descend all the sub-directories, finds the maximum length of any file in that hierarchy, and store the output in a file.

b. Create a shell script to find a file with particular name, (show separate outputs for both the conditions)

- if that file exists then rename the existing file and create an empty file with that name.
- if that file does not exist then create a new empty file.
- If both the conditions done together.

c. Set a cronjob for above developed scripts, that will be execute after every 30 minutes.

d. Illustrating the shell variables in a shell script.

4. a. Build a shell script to display the system space used. If it is greater than 80%, display as Low system Space and list the files having size greater than 1GB. Set up a cron job for the above developed script to execute every Monday morning 10AM.

b. Write a shell program to count number of words, characters, white spaces, and special symbols in each text and display the output on standard output. Set a cronjob to execute above script every 3rd day of week morning 9 AM.

AWK

5. a. Develop an awkscript that accepts date argument in the form of dd-mm-yy and display it in the form month, day, and year. The script should check the validity of the argument and in the case of error, display a suitable message.

   b. Develop an awkscript to delete duplicated line from a text file. The order of the original lines must remain unchanged.

   c. Set up a cron job for the above developed scripts to execute every other day evening 4 PM.

Sed and Find

6. a. Type the below given text and save the file as 1.txt using Vi/Vim editor. Perform the below given operations.

"Python is a very popular language.
Python is easy to use. Python is easy to learn.
Python is a cross-platform language
HTML is a Markup Language
Python Programming Language
C Programming Language
Shell Programming
Perl Programming Language
Bash"

i. Replace all instances of a *Python* in a second line of 1.txt with *Perl*.

ii. Replace the last occurrence of *Programming* with *Scripting* only of a match, not other instances.

iii. Create a text file in the path /MSRIT/CSE/UG/Python.txt. Replace full path with just the filename no directory (such as Python.txt) and display it on standard output.

iv. Add string before and after the matching pattern using '\1'. In the above given text, navigate yourself to last line, you can find *Bash* keyword, Add *Learn* before *Bash* and *Programming* after *Bash* keyword.

b. Perform the following execution using find command

i. Find all the files in a current directory, whose permissions are 0777.

ii. Assign a sticky bit to all the files in a current directory.

iii. Find Directories with full permissions (777) and by using chmod command change the permissions by assigning read, write and execute permissions to

owner and only read & execute to group and others.

iv. Find last 20 days modified files, accessed files.

v. Find all the files which are modified in last 1 hour.

C Programs by using UNIX File System Calls

7. a. Develop a C program to emulate the UNIX ls –li command, which lists all the attributes of the files in a specified directory.
   b. Write a C program to remove empty files from the given directory using system calls.

8. a. Write a C program to read n characters from a file and append them back to the same file using dup2 function.

   b. Write a C program to list all files in a current directory using systemcalls
.

9. a. Create a C program to simulate the copy command in Unix (cp command)

   b. Develop a C program to simulate the ls (list) Unix command using system calls.

10. a. Understanding File Descriptors and Building a C program to create two processes P1 and P2. P1 takes a string and passes it to P2. P2 concatenates the received string with another string without using string function and sends it back to P1 for printing, send the output to standard output.

    b. Create a C program to simulate Grep Unix Command using system calls.

11. **XML Integration**

Consider the student details given below and create an XML file and save it as sample.xml.

Student Details: -

SI. No.	Student Name	USN	Department Name
1.	Alex	1RITCS001	CSE
2.	Smith	1RITDS040	DS
3.	Saliena	1RITCV051	Civil
4.	Elizabeth	1RITME011	Mechanical

Department Details: -

SI. No.	Department Name	DepartmentId	Total no of enrolled students

1.	CSE	01	170
2.	DS	02	188
3.	Civil	03	160
4.	Mechanical	04	150

a. From the above XML file, separate only student details and redirect the output to a file.
b. Replacing the tag name from name to Dept name and change globally in a shell script.

c. Read the tag value of USN and redirect the output to standard output as well as redirect to a file.

12.**Docker**

a. Docker installation and set up
b. Create a shell script to pass arguments and run in a container using Docker.
c. Creating docker file
d. Building docker image, creating a container and running the shell scripts.
e. Displaying and running docker image.

***Note: Practical sessions will be based on the contents.**

Suggested Learning Resources

Text Books:

1) Introduction to UNIX & SHELL programming, M.G. Venkatesh Murthy, Pearson Education.
2) Unix concepts and applications, Fourth Edition, Sumitabha Das, TMH.
3) Unix for programmers and users, 3rd edition, Gaham Glass & K. Ables, Pearson education.

Course Outcomes:

At the end of the course students will be able to:

1. Understand the basics of Unix concepts and commands, dockers, containers and system calls of Unix.
2. Apply the commands to incorporate changes to the file system.
3. Implement the Shell scripts based on the real time scenarios.

Course Assessment and Evaluation:

Parameter	**Marks**
CIE Test	20
Lab Record Writing + Viva+ program execution	30
Total	**50**
Final Exam will be conducted for 50 marks (SEE)	