CMR ENGINEERING COLLEGE
UGC AUTONOMOUS
Approved by AICTE-New Delhi | Affiliated to JNTUH | Accredited by NAAC & NBA
Department of Computer Science & Engineering (Artificial Intelligence & Machine Learning)

AY 2025-2026

LIFT Planner (Lab Improvement for Future Trends)

On

Operating Systems Lab

II YEAR - SEM I

HOD **PRINCIPAL**

S. No	CONTENT
1	OBJECTIVES AND RELEVANCE
2	SCOPE
3	PREREQUISITES
4	SYLLABUS AS PER JNTUH
5	LEAD EXPERIMENTS
6	LAB SCHEDULE
7	SUGGESTED BOOKS
8	WEBSITES (USEFUL LINKS)
9	EXPERT DETAILS
10	MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS
11	INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12	SHADOW ENGINEERING
13	CALIBIRATION , TESTING AND INSPECTION
14	MAINTAINANCE AND TROUBLESHOOTING
15	ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL
16	VIRTUAL LABS REFERENCE LINKS

1. OBJECTIVES AND RELEVANCE

Upon successful completion of this Lab the student will be able to:

- ✓ To able to know about design aspects of operating system concepts through simulation.
- ✓ To learn the principles of IPC Mechanisms.
- ✓ To practice the handling, I/O System calls of UNIX/LINUX Operating systems.
- ✓ Introduce basic Unix commands, system call interface.

2. SCOPE

This lab subject is cover with Scheduling Algorithms, deadlock management and handling of I/O System calls of UNIX/LINUX Operating systems

3. PREREQUISITES

This subject recommends continuous practice flexibility of C programming.

4. Autonomous Syllabus-R20

1. Write C programs to simulate the following CPU Scheduling algorithms a) FCFS b) SJF c) Round Robin d) priority

2. Write C programs to using the I/O system calls ofUNIX /LINUX operating sys - tem (open, read, write, close, fcntl, seek, stat, opendir, readdir)

3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.

4. Write a C program to implement the Producer – Consumer problem using semaphores us- ing UNIX/LINUX system calls.

5. Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs c) Message Queues d) Shared Memory

6. Write C programs to simulate the following memory management techniques a) Paging b) Segmentation

7. Write C programs to simulate Page replacement policies a) FCFS b) LRU c) Optimal

LEAD PROGRAMS

1. Write a C program to simulate page replacement algorithms
   a) LRU
   b) Optimal Page Replacement Algorithm
2. Write a C program to simulate the concept of Dining-Philosophers problem.

LAB EXPERIMENTS:

Week No.	Name of the Program	No. of Hours required	UNIT NUMBER
1	Introduction & Basic Commands	1	I
2	1.Write C programs to simulate the following CPU Scheduling algorithms a) FCFS b) SJF	1	II
3	c) Round Robin d) priority	1	II
4	2.Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir)	1	II
5	3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.	2	III
6	4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls	1	III
7	5.Write C programs to illustrate the following IPC mechanisms a) Pipes b) FIFOs	1	III
8	c) Message Queues d) Shared Memory	1	III
9	6.Write C programs to simulate the following memory management techniques	2	IV

	a) Paging b) Segmentation		
10	LEAD Experiments	3	
	Total no of HOURS required to complete syllabus	**14**	

EXPERIMENT NO.1

Introduction of CPU scheduling algorithms – FCFS, SJF, Round Robin, Priority.

OBJECTIVE:

The main objective is to understand clearly about all the process scheduling algorithms.

PRE REQUISITES:

Basic scheduling process technique and communication techniques

DESCRIPTION:

1. Introduction to process techniques.
2. Introduction to scheduling techniques.
3. Implementation of scheduling algorithms.

APPLICATIONS:

1. These mechanisms can be practically applied in various Operating Systems.
2. These techniques are useful to improve the throughput.

EXPERIMENT NO.2

Introduction of I/O System calls of UNIX/LINUX

OBJECTIVE:

The main objective is to understand clearly about all the basic I/O system calls in UNIX/LINUX.

PRE REQUISITES:

Basic of system calls.

DESCRIPTION:

1. Introduction to I/O system calls.
2. Implementation of open, close, read, write, etc.

APPLICATIONS:

1. These mechanisms can be practically applied in various Operating Systems.
2. These techniques are useful to improve the throughput.

EXPERIMENT NO. 3

Deadlocks

OBJECTIVE:

The main objective is to understand clearly about Deadlocks technique

PREREQUISITES:

Basic knowledge of Deadlock characteristics.

DESCRIPTION:

1. Introduction to Deadlocks.
2. Deadlock detection
3. Deadlock Avoidance
4. Data Recovery from Deadlocks.
5. Deadlock prevention.

APPLICATION:

1. It gives the correct idea of Deadlocks.
2. It helps the student to know about Deadlock detection, Deadlock avoidance & prevention techniques.

EXPERIMENT NO.4 & 5

Introduction to Producer Consumer problem & Pipes, Inter-processor communication and semaphores

OBJECTIVE:

The main objective is to understand clearly about IPC Mechanisms.

PRE REQUISITES:

Basic communication techniques

DESCRIPTION:

1. Introduction to communication techniques.
2. Implementation of inter process communication with pipes
3. Implementation of semaphores

APPLICATIONS:

3. These mechanisms can be practically applied in various Operating Systems.
4. These techniques are useful to improve the throughput.
5. These techniques are used to solve communication problems and Producer - Consumer problem

EXPERIMENT NO. 6

Introduction of paging techniques and page replacement algorithms

OBJECTIVE:

The main objective is to understand clearly about all the page replacement algorithms and

Allocation techniques.

PRE REQUISITES:

Basic paging techniques.

DESCRIPTION:

1. Introduction to paging techniques.
2. Introduction to page replacement and allocation technique.
3. Implementation of page replacement algorithms.

APPLICATIONS:

1. These mechanisms can be practically applied in various secondary storage techniques such as disk storage.

LEAD EXPERIMENTS

1. OBJECTIVE

1. Simulate the following Page Replacement Algorithms

LRU (Least Recently Used)

DESCRIPTION:

Least Recently Used (LRU) page replacement algorithm works on the concept that the pages that are heavily used in previous instructions are likely to be used heavily in next instructions. And the page that are used very less are likely to be used less in future. Whenever a page fault occurs, the page that is least recently used is removed from the memory frames. Page fault occurs when a referenced page in not found in the memory frames.

Program for LRU Page Replacement Algorithm

```
#include<stdio.h>

int findLRU(int time[], int n){
	int i, minimum = time[0], pos = 0;

	for(i = 1; i < n; ++i){
		if(time[i] < minimum){
			minimum = time[i];
			pos = i;
		}
	}

	return pos;
}

int main()
{
   int no_of_frames, no_of_pages, frames[10], pages[30], counter = 0, time[10], flag1, flag2, i, j,
pos, faults = 0;
	printf("Enter number of frames: ");
	scanf("%d", &no_of_frames);

	printf("Enter number of pages: ");
	scanf("%d", &no_of_pages);

	printf("Enter reference string: ");

   for(i = 0; i < no_of_pages; ++i){
	scanf("%d", &pages[i]);
   }

	for(i = 0; i < no_of_frames; ++i){
	frames[i] = -1;
   }

   for(i = 0; i < no_of_pages; ++i){
	flag1 = flag2 = 0;

	for(j = 0; j < no_of_frames; ++j){
		if(frames[j] == pages[i]){
			counter++;
			time[j] = counter;
				flag1 = flag2 = 1;
				break;
```

```
                        }
        }

        if(flag1 == 0){
                        for(j = 0; j < no_of_frames; ++j){
                        if(frames[j] == -1){
                                counter++;
                                faults++;
                                frames[j] = pages[i];
                                time[j] = counter;
                                flag2 = 1;
                                break;
                        }
                }
        }

        if(flag2 == 0){
                pos = findLRU(time, no_of_frames);
                counter++;
                faults++;
                frames[pos] = pages[i];
                time[pos] = counter;
        }

        printf("\n");

        for(j = 0; j < no_of_frames; ++j){
                printf("%d\t", frames[j]);
        }
        }

        printf("\n\nTotal Page Faults = %d", faults);

    return 0;
}
```

OUTPUT

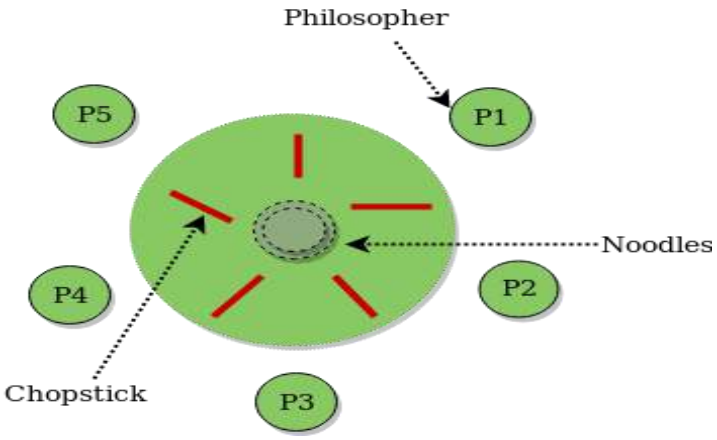
```
Enter number of frames: 3
Enter number of pages: 6
Enter reference string: 5 7 5 6 7 3


5 -1 -1
5 7 -1
5 7 -1
5 7 6
5 7 6
3 7 6
```

Total Page Faults = 4

2. Dining Philosopher Problem Using Semaphores

DESCRIPTION:

The Dining Philosopher Problem – The Dining Philosopher Problem states that K philosophers seated around a circular table with one chopstick between each pair of philosophers. There is one chopstick between each philosopher. A philosopher may eat if he can pickup the two chopsticks adjacent to him. One chopstick may be picked up by any one of its adjacent followers but not both.



Semaphore Solution to Dining Philosopher –
Each philosopher is represented by the following pseudocode:

```
process P[i]
 while true do
   {  THINK;
      PICKUP(CHOPSTICK[i], CHOPSTICK[i+1 mod 5]);
      EAT;
      PUTDOWN(CHOPSTICK[i], CHOPSTICK[i+1 mod 5])
   }
```

Code

```
#include <pthread.h>

#include <semaphore.h>
```

```
#include <stdio.h>


#define N 5
#define THINKING 2
#define HUNGRY 1
#define EATING 0
#define LEFT (phnum + 4) % N
#define RIGHT (phnum + 1) % N
int state[N];
int phil[N] = { 0, 1, 2, 3, 4 };
sem_t mutex;
sem_t S[N];
void test(int phnum)
{
    if (state[phnum] == HUNGRY
        && state[LEFT] != EATING
        && state[RIGHT] != EATING) {
        // state that eating
        state[phnum] = EATING;
        sleep(2);
```

```
        printf("Philosopher %d takes fork %d and %d\n",
                        phnum + 1, LEFT + 1, phnum + 1);
        printf("Philosopher %d is Eating\n", phnum + 1);
        // sem_post(&S[phnum]) has no effect
        // during takefork
        // used to wake up hungry philosophers
        // during putfork
        sem_post(&S[phnum]);
    }
}
// take up chopsticks
void take_fork(int phnum)
{
    sem_wait(&mutex);
    // state that hungry
    state[phnum] = HUNGRY;
    printf("Philosopher %d is Hungry\n", phnum + 1);
    // eat if neighbours are not eating
    test(phnum);
    sem_post(&mutex);
```

```
    // if unable to eat wait to be signalled
    sem_wait(&S[phnum]);
    sleep(1);
}
// put down chopsticks
void put_fork(int phnum)
{
    sem_wait(&mutex);
    // state that thinking
    state[phnum] = THINKING;
    printf("Philosopher %d putting fork %d and %d down\n",
           phnum + 1, LEFT + 1, phnum + 1);
    printf("Philosopher %d is thinking\n", phnum + 1);
    test(LEFT);
    test(RIGHT);
    sem_post(&mutex);
}
void* philospher(void* num)
{
    while (1) {
```

```
        int* i = num;
        sleep(1);
        take_fork(*i);
        sleep(0);
        put_fork(*i);
    }
}
int main()
{
    int i;
    pthread_t thread_id[N];
    // initialize the semaphores
    sem_init(&mutex, 0, 1);
    for (i = 0; i < N; i++)
     sem_init(&S[i], 0, 0);
    for (i = 0; i < N; i++) {
        // create philosopher processes
        pthread_create(&thread_id[i], NULL,
                       philospher, &phil[i]);
```

```
        printf("Philosopher %d is thinking\n", i + 1);
    }
    for (i = 0; i < N; i++)
        pthread_join(thread_id[i], NULL);
}
```

A) LAB SCHEDULE:

CYCLE 1:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8
B1	Demo	Exp.1	Exp.1	Exp.2	Exp.3	Exp.4	Test	Exp.5
B2	Demo	Exp.1	Exp.1	Exp.2	Exp.3	Exp.4	Test	Exp.5

Batches	week-9	week-10	week-11	week-12	week-13	week-14
B1	Exp.5	Exp.6	Exp.6	Test	LEAD 1 &2	Lab internal
B2	Exp.5	Exp.6	Exp.6	Test	LEAD 1 & 2	Lab internal

(B) VIVA SCHEDULE: The viva schedule should be planned prior to the lab experiment.

CYCLE 1: **ROUND -** 1

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10	week-11	week-12	week-13	week-14
B1,B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	Viva	Viva	viva	viva

ROUND - 2

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10
SG1	viva									
SG2			viva							
SG3					viva					
SG4							viva			
SG5									viva	

*SG: Selected Group with a maximum of 6 or 12 students

LAB EXTERNAL:			
S.NO	Write up	Final Evaluation	Viva
1	**1. Aim 2. Procedure 3. Program 4. Expected output.**	**Based on correctness of the program and Results**	**Based on understanding of Experiment and theoretical questions in the related subjects**
Marks	**25**	**25**	**10**
Total Marks-60			

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams is as follows:

LAB INTERNAL:							
Day to Day Evalution-10					**Internal Exam-20**		
Uniform	**Observation & Record**	**Performance Of the Experiment**	**Result**	**Viva**	**Write up**	**Execution & Results**	**Viva**
Marks-2	**Marks-2**	**Marks-2**	**Marks-2**	**Marks-2**	**Marks-5**	**Marks-5**	**Marks-10**
Total Marks-30							

5. SUGGESTED BOOKS:

1.Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley

2. Advanced programming in the Unix environment, W.R.Stevens, Pearson education.

6. WEBSITES

1. https://en.wikipedia.org/wiki/Operating_system
2. https://nptel.ac.in/courses/106/108/106108101/#

7. EXPERTS DETAILS

The Expert Details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally. There are a few others known as well.

INTERNATIONAL

1.Martin C. Rinard, MIT Computer Science and Artificial Intelligence Laboratory ,The Stata Center, Building 32-G82832 Vassar Street, Cambridge, MA 02139 Email: rinard@lcs.mit.edu .

2. Abraham Silberschatz, Professor and Chair of Computer Science at Yale University.

3.Andrew S. Tanenbaum, American computer scientist and professor of computer science,

NATIONAL

1. Matthew Jacob T. Ph.D. (SERC) Professor. Performance aspects of Computer Architecture, Operating Systems, Parallel Processing. Email: mjt@serc.iisc.ernet.in.

2. Mrs. A. Kalaivani MCA., M. Phil Asst. Professor, NIITM, NEW DELHI.

REGIONAL

1. Dr Raja shekar, professor,CSE VNRVJIET Hyderabad, Email: rajasekarm9@gmail.com.

9849296429

2. Dr prashanth Rao Professor, IT Anurag

university,Hyderabad,Email:prashanthrao@cvsr.ac.in,9490232922

8. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D:

PROPOSAL 1: Scheduling Techniques

To implement this simplified process management scheme using 2 different scheduling techniques: prioritized round robin and shortest remaining time.

Your project is to implement this simplified process management scheme using 2 different scheduling techniques: prioritized round robin and shortest remaining time. You also need to keep track of various statistcal data for each scheduling technique and job: total time each process was in each state, number of jobs run, total elasped time, total elapsed time per job (throughput), and longest and shortest time taken for any job in the system.

The system will have the following behavior. You must have a way to keep time intervals (clock ticks) during the execution of the system. A simple approach to this problem might be to use a loop and for each iteration, increment the clock. Seven things should be done in the scheduling loop during each cycle, the first six of which pretty much map directly to the transitions of the diagramabove. For this simulation jobs will do no "real" work; i.e. your simulation will implement what happens when the CPU invokes the operating system process scheduler. You can think of this project as a function call inside the operating system. A function call that updates all job states and sets up the next job to run, so that when the function call completes, the CPU has next job and can begin to run it.

Since our simulated jobs are doing no "real work", we have a little problem. If they are not actually executing code, they cannot request I/O. If they cannot request I/O, we can't transition from *running* to *sleeping on I/O*. And since there is no real I/O going on, I/O can never complete, so we cannot transition from *sleeping on I/O* to a *ready to run*. This is a problem. But we have a solution. When a job is in the *running* state you will have a function that randomly decides whether or not the job has requested I/O. When a job is in the "waiting for I/O" state, you will have a function that randomly decides whether or not I/O has completed for that job.

PROPOSAL 2: A "Slower" File System

Objectives

There are two objectives to this assignment:

1. To understand how file systems work, specifically the directory hierarchy and storage management.
2. To understand some of the performance issues file systems deal with.

Overview

In this project, we will build a user-level library, libFS, that implements a good portion of a file system. Your file system will be built inside of a library that applications can link with to access files and directories. Your library will in turn link with a layer that implements a "disk"; we provide this library, LibDisk, which you must use.

LibFS Specification

We'll start by describing the LibFS API to the file system. There are three parts to the API: two generic file system calls, a set of calls that deal with file access, and a set of calls that deal with directories, and

Applications (e.g., your own test applications, and certainly our test applications) will link with LibFS in order to test out your file system.

Your library will be tested on how functions and also particularly on how it handles errors. When an error occurs (each possible error is specified below in the API definition), your library should set the global variable **osErrno** to the error described in the API definition below and return the proper error code. This way, applications that link with your library have a way to see what happened when the error occurred.

9. PROPOSALS INDUSTRIAL VISIT

Relevant IV (Industrial Visit) (VIP= Visit of Industrial Proposals)

If students visited different software development companies like TCS, WIPRO, INFOSYS.

S.No	Type of industry	Nature of industry	Date of visit (Tentative Date)	No. of students participated	Year/branch	Remarks

1	**MNC**	**software**		**20 to 30**	**CSE**	
2	**MNC**	**software**		**20 to 30**	**CSE**	

10. SHADOW ENGINEERING

GUIDELINES FOR SHADOW ENGINEERING (VIP)

INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)

OBJECTIVES OF SHADOW ENGINEERING:

1. To uplift the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

TABLE 2: INDUSTRIAL TRAINING (Shadow Engineering)
(Career Visit Approval)

S.No	Name of the Course	Nature of industry	Duration of Training	Authority	Date of Training/Certificate No.	Remarks
1	**Operating Systems**	**Software**	**10 hours**			

11. Calibration, Testing & Inspection (Method & Tools to be used)

Calibration

There is no need of any equipment to be calibrated.

Testing & Inspection:

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment

Linux OS Installation steps :(Red Hat Linux GUI)

Step 1 – Insert the Red Hat Linux DVD into the DVD-drive of your computer. As soon as the following screen pops up, press 'Enter' to install Red Hat Enterprise Linux (RHEL) through GUI mode.

Step 2- RHEL installer would then prompt you to conduct a check as to whether the CD media from which you're installing is functioning correctly or not. Choose 'Skip', press enter and the installation would begin.

Step 3- Next, we need to select the language – English or any other language as per your preference, and then press 'Next'.

Step 4- In this step, the RHEL installer would ask you about the appropriate type of keyboard for the system. We take the 'US English' keyboard; you can pick any other option depending on the type of your keyboard. Then press 'Next' to move to the next step.

Step 5- Next, the installer would ask for an 'installation number' if you wish to install full set of Red Hat functionalities. Enter the installation number and press 'OK' if you have an officially licensed installation number (for corporate clients that buy Red Hat's backup support and full features).

Step 6- The Red Hat installer would then require you to create partitions in your computer's hard disk for the installation. You can do it in four ways but the simplest way is to select 'Use free space on selected drives and create default layout' as this option will not affect any other OS residing in your system.

Step 7- In this step you must create the required system partitions and mount points such as '/boot', '/home', 'swap' etc which are required for the Linux's proper functioning.

Step 8- This step pertains to the default OS that will be loaded by the GRUB loader.

Step 9- This step pertains to the network settings of the Linux system that you are going to install. You can select the Ethernet devices through which the system would communicate with other devices in the network.

Step 10- The next step is to adjust the system clock to your particular time zone. Select your time zone and then click 'Next'.

Step 11 – This is a very important step that deals with the root (super-user) password for the system. Type the password and confirm it and then click next.

Step 12 – The RHEL installer would then prompt you about if you wish to install some extra 'Software Development' or 'Web Server' features. By default, keep it at 'Customize later' and press 'Next'.

Step 13- This next step will initiate the installation of Red Hat Linux, press 'Next' to begin the process.

Step 14- Upon the completion of installation you should the following screen. Press Reboot and you'd be ready to use your newly installed Red Hat Linux OS.

Commands in Linux for creating, running, testing C programs

Creating program:

vi program name .c

vi commands:

for saving the commands is: esc: wq

for editing the command is: esc: i

for compiling the command is :gcc programname.c –o programname

for running the command is: ./programname

12. MAINTAINANCE AND TROUBLESHOOTING:

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

S.No.	Software	Date of Maintenance	Type of Activity	Remarks
1	**Operating System**		**Soft wares installed(PUTTY)**	

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

13. ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).

The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge
4. Ensure Effective environmental conditions (temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipments from dust, dirt etc.
5. Ensure Calibration of instruments/equipment (Only NABL accredited authorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team.
10. Auditors should submit the detailed report of each lab duly signed by them to the Principal.
11. Each lab should maintain all the bills/invoices of each instrument or equipment in a separate file.
12. All the stock registers either consumable or non consumable should be updated whenever any purchases of consumables or equipment takes place.
13. All the safety precautions should be properly displayed in front of each lab.
14. All the LEAD experiments should be maintained separately in a record / recorded in a separate folder.
15. Based on the Pre Assessment report submitted by the auditor, corrective actions should be carried out by each lab in charge and that must be forwarded to concerned HOD and Principal.

16. VIRTUAL LABS REFERENCE LINKS

No virtual labs are available.

AY 2025-2026

LIFT Planner (Lab Improvement for Future Trends)

On

JAVA PROGRAMMING LAB

II YEAR - I SEM

HOD **PRINCIPAL**

JAVA PROGRAMMING LAB

Sl. No	CONTENT
1	OBJECTIVES AND RELEVANCE
2	SCOPE
3	PREREQUISITES
4	SYLLABUS AS PER JNTUH
5	LEAD EXPERIMENTS
6	LAB SCHEDULE
7	SUGGESTED BOOKS
8	WEBSITES (USEFUL LINKS)
9	EXPERT DETAILS
10	MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS
11	INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12	SHADOW ENGINEERING
13	CALIBIRATION , TESTING AND INSPECTION
14	MAINTAINANCE AND TROUBLESHOOTING
15	ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL
16	VIRTUAL LABS REFERENCE LINKS

1. OBJECTIVES AND RELEVANCE

- To write programs using abstract classes.
- To write programs for solving real world problems using java collection frame work.
- To write multithreaded programs.
- To write GUI programs using swing controls in Java.
- To introduce java compiler and eclipse platform.
- To impart hands on experience with java programming.

2. SCOPE

This lab subject is cover with all object oriented principles, multithreading, GUI programming using Swings and Java Collection frame work which increases student's ability to develop applications for a range of problems using object-oriented programming techniques.

Example like as

- How to solve real world problems using java collection frame?
- How to design and develop application related to multitasking and multiprocessing?
- How to ensure reusability and extensibility of modules?
- How to develop GUI programs using swing controls in Java ?

3. PREREQUISITES

a. Basic coding knowledge in C and C++ Programming

4. SYLLABUS

1. Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.

2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.

3. a) Develop an applet in Java that displays a simple message.
   b) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked.

3. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

4. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.

5. Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the above list. Display the contents of the list after deletion.

6. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with "Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no message shown.

7. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

8. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.

9. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

10. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).

11. Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.

12. Write a Java program to list all the files in a directory including the files present in all its sub directories.

13. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order

14. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.

5. LEAD EXPERIMENTS

LEAD 1: JAVA Program to change the host name to its specific IP address

LEAD 2: JAVA program to validate an email address

LEAD 3: JAVA Program to print date in different formats

6. LAB SCHEDULE

Week No.	Name of the Program	No. of Lab sessions	Text Books	**Unit number**	**Mode of Assessment**
1	Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop	1	T1	Unit -I	Viva&Execution
2	Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.	1	T1	UNIT-V	Viva&Execution
3	a) Develop an applet in Java that	1	T1	UNIT I &UNIT -V	Viva&Execution

	displays a simple message. Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named "Compute" is clicked				
4	Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.	1	T1	UNIT –III AND UNIT -V	Viva&Execution
5	Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.	1	T1	UNIT -III	Viva&Execution
6	Write a Java program for the following: Create a doubly linked list of elements. Delete a given element from the above list. Display the contents of the list after deletion.	1	T1	UNIT -IV	Viva&Execution
7	Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with	1	T1	UNIT -V	Viva&Execution

	"Stop" or "Ready" or "Go" should appear above the buttons in selected color. Initially, there is no message shown				
8	Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.	1	T1	UNIT –I AND UNIT-II	Viva&Execution
9	Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.	1	T1	UNIT-V	Viva&Execution
10	Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).	1	T1	UNIT-V	Viva&Execution
11	Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).	1	T1	UNIT -II	Viva&Execution
12	Write a Java program that	1	T1	UNIT -III	Viva&Execution

	correctly implements the producer – consumer problem using the concept of interthread communication.				
13	Write a Java program to list all the files in a directory including the files present in all its subdirectories.	1	T1	UNIT-II	Viva&Execution
14	Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order	1	T1	UNIT-IV	Viva&Execution
15	Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.	1	T1	UNIT-IV	Viva&Execution
	Total no of Lab sessions required to complete syllabus	**15**			

CYCLE 1:

Batches	**week-1**	**week-2**	**week-3**	**week-4**	**week-5**	**week-6**	**week-7**	**week-8**
B1	Basic programs & Exp1	Exp.2	Exp.3	Exp4	Exp.5	Exp 6 &LEAD 1	Exp.7	Exp 8
B2	Basic programs & Exp1	Exp.2	Exp.3	Exp4	Exp.5	Exp 6 &LEAD 1	Exp.7	Exp 8
B3	Basic programs & Exp1	Exp.2	Exp.3	Exp4	Exp.5	Exp 6 &LEAD 1	Exp.7	Exp 8
B4	Basic programs & Exp1	Exp.2	Exp.3	Exp4	Exp.5	Exp 6 &LEAD 1	Exp.7	Exp 8

Batches	week-9	week-10	week-11	week-12	week-13	week-14	week-15
B1	Exp 9&LEAD 2	Exp.10	Exp11	Exp12 &LEAD 3	Exp 13	Exp 14	Exp 15
B2	Exp 9&LEAD 2	Exp.10	Exp11	Exp12 &LEAD 3	Exp 13	Exp 14	Exp 15
B3	Exp 9&LEAD 2	Exp.10	Exp11	Exp12 &LEAD 3	Exp 13	Exp 14	Exp 15
B4	Exp 9&LEAD 2	Exp.10	Exp11	Exp12 &LEAD 3	Exp 13	Exp 14	Exp 15

(B) VIVA SCHEDULE:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10	week-11	week-12	week-13	week-14	week-15
B1,B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva	viva
B3,B4	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva	viva
B1,B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	Viva	Viva
B3,B4	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva	viva

ROUND - 2

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10
SG1	viva									
SG2			viva							
SG3					viva					
SG4							viva			
SG5									viva	

*SG: Selected Group with a maximum of 6 or 12 students

(C) Scheme of Evaluation:

Java programming

The scheme of evaluation for internal and external exams is as follows:

LAB INTERNAL:									
Day to Day Evalution-10							Internal Exam-20		
Uniform	Observation	Record	Performance Of the Experiment	Result	Diagrams/ Figures	Viva	Aim/ Write up	Execution &Results	Result & Conclusions
Marks-2	Marks-2	Marks-2	Marks-2	Marks-2	Marks- 2	Marks - 10	Marks-3	Marks-3	Marks-2
Total Marks-30									

LAB EXTERNAL			
S.NO	Write up	Final Evaluation	Viva
1	1. Aim 2. Procedure 3. Program 4. Expected output.	Based on correctness of the program and Results	Based on understanding of Experiment and theoretical questions in the related subjects
Marks	25	25	10
Total Marks-60			

7. SUGGESTED BOOKS

a. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.

b. Thinking in Java, Bruce Eckel, Pearson Education.

c. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.

d. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

8. WEBSITES (USEFUL LINKS)

a. http://www.btechsmartclass.com/java/java-programs-list.html

b. https://www.tutorialspoint.com/java/index.htm

c. https://www.tutorialspoint.com/java/index.htm

d. https://onlinecourses.nptel.ac.in/noc21_cs03/preview

9. EXPERT DETAILS

- **International**
  - **James Gosling**, Engineer at Amazon Web Services (AWS) and the original developer of Java, Blog: **nighthacks.com/roller/jag**
- **National**
  - **Mr. Ranganath Vadapalli,** Java full stack developer, JP Morgan Chase, Email - vadapalliranganath@gmail.com, Contact No – 8884057329, Expert in Java, java8, Spring Boot, Angular and React JS.
- **Local**
  - **Dr. Mandru dheena babu,** Professor, HOD, Dept of IT, MREC, 9491369993.
  - **Dr.Aravid,professor**,Dean of Accreditations,HITAM **college.**7019933070

10. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS

Student Management System

This project is also a simple Student Management System. In this project you will be learning how to add new students to the database, how to generate a 5 digit unique student ID for each student, how to enroll students in the given courses. Also, you will be implementing the following operations enroll, view balance, pay tuition fees, show status, etc. The status will show all the details of the student including name, id, courses enrolled and balance. This is one of the best projects to implement the OOPS concepts.

11. INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)

Students visit schedule different software development MNC companies.

S.No	Type of industry	Nature of industry	Date of visit (Tentative Date)	No. of students participated	Year/branch	remarks
1	MNC	software	April month	20 to 30	CSE – II Year	
2	MNC	software	June	20 to 30	CSE – II Year	

12. SHADOW ENGINEERING

- To uplift the knowledge of the students related to laboratories.
- To improve the industry-college interactions.
- To create industry like environment for all the students in order to make future Assignment.

13. CALIBIRATION, TESTING AND INSPECTION

Calibration

There is no need of any physical equipment to be calibrated but we use Eclipse IDE to control the errors.

Testing & Inspection:

We use J Unit to perform testing Java programs

14. MAINTAINANCE AND TROUBLESHOOTING

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

S.No.	Software	Date of Maintenance	Type of Activity	Remarks
1	Operating System		Software installed	
2	JDK		Installation	
3	Eclipse		Installation	

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

15. ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).

The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge
4. Ensure Effective environmental conditions (temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipments from dust, dirt etc.
5. Ensure Calibration of instruments/equipment (Only NABL accredited authorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team. Auditors should submit the detailed report of each lab duly signed by them to the Principal.

16. VIRTUAL LABS REFERENCE LINKS

a. No virtual labs are available.

CMR ENGINEERING COLLEGE
UGC AUTONOMOUS
Approved by AICTE-New Delhi | Affiliated to JNTUH | Accredited by NAAC & NBA
Department of Computer Science & Engineering (Artificial Intelligence & Machine Learning)

AY 2025-2026

LIFT Planner (Lab Improvement for Future Trends)

On

PYTHON PROGRAMMING LAB

III YEAR - I SEM

HOD **PRINCIPAL**

S.No	CONTENT
1	OBJECTIVES AND RELEVANCE
2	SCOPE
3	PREREQUISITES
4	SYLLABUS AS PER AUTONOMUS
5	LEAD EXPERIMENTS
6	LAB SCHEDULE
7	SUGGESTED BOOKS
8	WEBSITES(USEFULLINKS)
9	EXPERT DETAILS
10	MAPPINGOFLABWITHPROJECT/CONSULTANCY/R&D PROPOSALS
11	INDUSTRIAL VISITS(IIP–INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12	SHADOW ENGINEERING
13	CALIBIRATION,TESTING AND INSPECTION
14	MAINTAINANCE AND TROUBLE SHOOTING
15	ASSESSMENT AND ACCREDITATION PROCEDURE AS PERNABL
16	VIRTUAL LABS REFERENCELINKS

1. OBJECTIVESANDRELEVANCE

Up on successful completion of this Lab the student will be able to:

- ✓ Use Analyze the core concepts of basic programming design with function susing Python.
- ✓ Analyze in-depth data and information processing techniques.
- ✓ Analyze basic concepts of scripting and contributions of scripting languages.
- ✓ Explore built –in objects of python.
- ✓ CreatepracticalandcontemporaryapplicationssuchasTCP/IPnetworkprogramming, Web applications, discrete event simulations.

2. Scope:

The practical/exercises in this action are psychomotor domain Learning Outcomes(i.e. subcomponents of the COs), to be developed and assessed to lead to the attainment of the competency.

- ✓ Understand principles of Python.
- ✓ Ability to understand Python code, develop medium-difficulty applications in Python.
- ✓ Understandtheprosandconsonscriptinglanguagesvs.classicalprogramming languages.
- ✓ Use Python data structure & develop Python programs by defining functions & calling them.
- ✓ Understand object oriented programming

3. PREREQUISITES

1. To analyze Python, one should have knowledge about basic programming concepts like:
2. Data types
3. Variables
4. Conditional statements
5. Loops
6. Functions

4. Autonomous Syllabus

7. Write a program to demonstrate different number data types in Python.
8. Write a program to perform different arithmetic operations on numbers in Python.
9. Write a program to create, concatenate, and print a string, and access sub-strings from a given string.
10. Write a Python script to print the current date in the following format:
    "Sun May 29 02:26:23 IST 2017"

11. Write a program to create, append, and remove items from lists in Python.

12. Write a program to demonstrate working with tuples in Python.

13. Write a program to demonstrate working with dictionaries in Python.

14. Write a Python program to find the largest of three numbers.

15. Write a Python program to convert temperatures to and from Celsius and Fahrenheit.

16. Formula: C/5 = (F - 32)/9

19. Write a Python program to construct the following pattern using a nested for loop:

Mark down

```
*
* *
* * *
* * * *
* * *
* *
*
```

20. Write a Python script that prints prime numbers less than 20.

21. Write a Python program to find the factorial of a number using recursion.

22. Write a program that accepts the lengths of three sides of a triangle as inputs.

- The program output should indicate whether or not the triangle is a right triangle.
- *(Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides.)*

23. Write a Python program to define a module to find Fibonacci numbers and import the module into another program.

24. Write a Python program to define a module and import a specific function in that module into another program.

25. Write a script named copyfile.py. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.

26. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.

27. Write a Python class to convert an integer to a Roman numeral.

28. Write a Python class to implement pow(x, n).

29. Write a Python class to reverse a string word by word.

Basic Programs

1. Python program to print "Hello Python" – Analyze basic print function.
2. Python program to do arithmetical operations – Analyze basic arithmetic operations.
3. Python program to find the area of a triangle – Analyze how to seek inputs from the user.
4. Python program to solve a quadratic equation – Analyze applying the formulae.
5. Python program to swap two variables – Analyze logical application on memory.
6. Python program to generate a random number – Analyze random() method.
7. Python program to convert kilometers to miles – Analyze simple conversions.
8. Python program to convert Celsius to Fahrenheit – Analyze simple conversions.
9. Python program to display a calendar – Analyze importing calendar and working with it.

5. LEAD Programs

1. Write a Python program to upload an Excel file and display its contents.
2. Write a Python program to convert speech to text and text to speech.

MAINLINKAGEOFPYTHONPROGRAMMINGTHEORYWITHLABEXPERIMENTS:

Wee kNo.	Name of the Program	No.of Hours required	UNIT NUMBER
1	Basic Programs	9	I
2	Write a program to demon strate different number data types in Python	1	II

3	Write a program to create, concatenate and print a string and accessing sub-string from a given string. Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST	1	I
4	Write a program to create,append,and remove lists in python. Write a program to demon strate working with tuples in python.	1	II
5	Write a program to demonstrate working with dictionaries in python. Write a python program to find largest of three numbers.	1	II
6	Write a Python program to convert temperatures to and from Celsius, Fahrenheit. [Formula:c/5=f-32/9]	1	II
7	Write a Python program to construct the following pattern, using anested for loop * * * * * * * * * * * * * * * * * * Write a Python script that prints prime numbers less than 20.	1	IV

8	Write a Python script that prints prime numbers less than 20.	1	I

	Write a python program to find factorial of a number using Recursion		
9	Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is aright triangle (Recallfrom the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides). Write a python program to define a module to find Fibonacci Numbers and import the module to another program.	1	I
10	Write a python program to define a module and import a specific function in that module to another program. Write a script named copyfile.py.Thisscript should prompt the user for the names of two text	1	III
11	Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order. Write a Python class to convert an integer to a roman numeral.	1	I
12	Write a Python class to implement pow(x, n) Write a Python class to reverse a string word byword	1	III
13	LEAD Experments	1	
	Total no of HOURS required to complete syllabus	**14**	

1. Python Code to Accept Voice Input and Display the Same on Output Console

□ Description:

Speech Recognition is an important feature in several applications such as home automation, artificial intelligence, etc.

□ Installation Required:

- Python Speech Recognition module:
  `pip install speechrecognition`
- PyAudio:
  For **Linux users**:
  `sudo apt-get install python3-pyaudio`
  For **Windows users**:
  `pip install pyaudio`
- Python `pyttsx3` module (for text-to-speech):
  `pip install pyttsx3`

□ Speech Input Using a Microphone and Translation of Speech to Text

Ambient Noise Adjustment:
Since surrounding noise varies, allow the program a second or two to adjust the energy threshold of recording, according to the external noise level.

Speech to Text Translation:
This is done using Google Speech Recognition. It requires an **active internet connection**.
(Offline alternatives like PocketSphinx exist but require complex installation.)

□ Translation of Speech to Text with Speech Output

- **Initialize the recognizer and engine:**

```
import speech_recognition as sr
import pyttsx3

# Initialize the recognizer
r = sr.Recognizer()

# Function to convert text to speech
def SpeakText(command):
    engine = pyttsx3.init()
    engine.say(command)
    engine.runAndWait()
```

- **Continuous Speech Recognition Loop:**

```
while True:
    try:
        with sr.Microphone() as source2:
            # Adjust for ambient noise
            r.adjust_for_ambient_noise(source2, duration=0.2)

            # Listen for the user's input
            audio2 = r.listen(source2)

            # Recognize speech using Google
```

```
            MyText = r.recognize_google(audio2)
            MyText = MyText.lower()

            print("Did you say: " + MyText)
            SpeakText(MyText)

    except sr.RequestError as e:
        print("Could not request results; {0}".format(e))

    except sr.UnknownValueError:
        print("Unknown error occurred")
```

Input Example:
Voice speech: "Hi buddy how are you"

Output:
Did you say: hi buddy how are you

✅2. Python Program to Upload an Excel File and Display

We can easily import an Excel file into Python using **Pandas**. For this, we use the `read_excel` function.

□ Import Required Module

```
import pandas as pd
```

□ Basic File Load

```
python
CopyEdit
df = pd.read_excel(r'Path\to\your\ExcelFile.xlsx')
print(df)
```

Note: For older Excel versions, use `.xls` instead of `.xlsx`.

□ To Load a Specific Sheet:

```
python
CopyEdit
df = pd.read_excel(r'Path\to\your\ExcelFile.xlsx', sheet_name='Sheet1')
print(df)
```

□ Steps to Import an Excel File into Python using Pandas

Step 1: Capture the File Path

Example path:
`C:\Users\AIML\Desktop\ProductList.xlsx`

Update the path in your code to reflect the actual location and filename, including the `.xlsx` extension.

Step 2: Apply the Python Code

Use the examples above and modify as per your file.

Step 3: Run the Code

To read `.xlsx` files, install the **openpyxl** module:

```
pip install openpyxl
```

6.LAB SCHEDULE:

CYCLE 1:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8
B1	Demo	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	test	Exp.6
B2	Demo	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	test	Exp.6
B3	Demo	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	test	Exp.6
B4	Demo	Exp.1	Exp.2	Exp.3	Exp.4	Exp.5	test	Exp.6

Batches	week-9	week-10	week-11	week-12	week-13	week-14
B1	Exp.7	Exp.8	Exp.9	test	LEAD1&2	Labinternal
B2	Exp.7	Exp.8	Exp.9	test	LEAD1&2	Labinternal
B3	Exp.7	Exp.8	Exp.9	test	LEAD1&2	Labinternal
B4	Exp.7	Exp.8	Exp.9	test	LEAD1&2	Labinternal

(B) VIVA SCHEDULE : The viva schedule should be planned prior to the lab experiment.

CYCLE1: **ROUND-**1

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10	week-11	week-12	week-13	week-14
B1,B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva
B3,B4	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva
B1,B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva
B3,B4	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva

ROUND-2

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10
SG1	viva									
SG2			viva							
SG3					viva					
SG4							viva			
SG5									viva	

*SG :Selected Group with a maximum of 5 or 10 students

(C) Scheme of Evaluation:

The scheme of evaluation for internal and external exams is as follows:

LAB INTERNAL:							
DaytoDayEvalution-20					**InternalExam-10**		
Uniform	**Observation& Record**	**Performance Of the Experiment**	**Result**	**Viva**	**Writeup**	**Execution &Results**	**Viva**
Marks-3	**Marks-3**	**Marks-3**	**Marks-3**	**Marks-3**	**Marks-4**	**Marks-3**	**Marks-3**
TotalMarks-30							

LAB EXTERNAL			
S.NO	**Write up**	**Final Evaluation**	**Viva**
1	1. Aim 2. Procedure 3. Program 4. Expected output.	Based on correctness of the program and Results	Based on understanding of Experiment and theoretical questions in the related subjects
Marks	**25**	**25**	**10**
Total Marks-60			

7. SUGGESTEDBOOKS:

1. OperatingSystemPrinciples-AbrahamSilberchatz,PeterB.Galvin,GregGagne7thEdition, John Wiley

2. Advanced programming in the Unix environment,W.R.Stevens, Pearsoneducation.

REFERENCEBOOKS:

1. MichaelHGoldwasser,DavidLetscher,"ObjectOrientedProgramminginPython",Prentice Hall, 1st Edition, 2007.

2. YashavantKanetkar,AdityaKanetkar,"LetusPython",BPBpublication,1st Edition,2019.

3. AshokKamthane,AmitKamthane,"ProgrammingandProblemSolvingwithPython", McGraw Hill Education (India) Private Limited, 2018

8.WEBSITES

1. https://en.wikipedia.org/wiki/Python_(programming_language)
2. https://onlinecourses.swayam2.ac.in/aic20_sp33/preview
3. https://onlinecourses.nptel.ac.in/noc22_cs70/preview

9.EXPERTS DETAILS

The Expert Details which have been mentioned below are only a few of the eminent one sknown Internationally, National lyand Locally.There area few others known as well.

INTERNATIONAL

GUIDO VAN ROSSUM

He's Put In Time AtA Number Of High-ProfileCompanies,BringingHis Unique Platform Expertise With Him Every where HeGoes.

Currently, He Splits His Time between Contributing to Python and Working at Drop box.

https://gvanrossum.github.io/

NATIONAL

1. Mrs.A.Kalaivani MCA.,M.PhilAsst.Professor,NIITM,NEW DELHI.

REGIONAL

1. Dr Rajashekar, professor, CSE

VNRVJIETHyderabad,Email:rajasekarm9@gmail.com.9849296429

2. Dr prashanth Rao Professor, IT Anurag

university,Hyderabad,Email:prashanthrao@cvsr.ac.in,9490232922

10. MAPPING OF LAB WITH PROJECT /CONSULTANCY/R&D:

PROPOSAL 1: Text based adventure game

Project-based learning not only helps in the logic building but it also enhances our problem-solving skill. There are different types of projects that you can create as a beginner in Programming.
A Text-Based Adventure Game is a type of game in which a player has to make choices (Yes / No) in every step of the game. Based on these choices, the story line changes, and a at last, weget to know that whether the player wins or loses the game.

PROPOSAL 2: Python Story generator

The **Random Story Generator** project aims to generate a unique story every time the user executes the code. A story is made up of a collection of sentences. We will use random phrases to construct these sentences, and thus generate different stories each time.

? How do we achieve this?

The answer is very simple:

- **We will first put the elements of the story in different lists.**
- **Then, we will use the random module to select random parts of the story collected in those lists.**
- **Finally, we will concatenate them to make a story.**

We make use of the random. choice() function.

Implementation Steps:

1. **Import the random module**, which is a built-in module in Python. So, there's no need to install it manually:

```
import random
```

2. **Define several lists of phrases**
   you may define more lists depending on your story structure. Here are eight commonly used lists:
   - sentence_starter – Gives an idea about the time of the event.
   - character – Describes the main character of the story.
   - time – Defines the exact day on which some incident occurred.
   - story_plot – Describes the plot of the story.
   - place – Indicates where the incident occurred.
   - second_character – Describes another character involved.
   - age – Specifies the age of the second character.

- work – Tells what the second character was doing.

3. **Use random. Choice () to select one item from each list**, and **concatenate them** to form a story.

Example:

python

CopyEdit

```
print(random.choice(sentence_starter) + " " +
    random.choice(character) + " " +
    random.choice(time) + " " +
    random.choice(story_plot) + " " +
    random.choice(place) + " " +
    random.choice(second_character) + " " +
    random.choice(age) + " " +
    random.choice(work))
```

This project enhances creativity, programming skills, and logical thinking using basic Python constructs.

PROPOSALS – INDUSTRIAL VISIT

Relevant IV (Industrial Visit)

VIP = Visit of Industrial Proposals

Industrial visits help students bridge the gap between theoretical knowledge and real-time industrial practices. It exposes them to real-world projects, tools, development cycles, and workflows.

 Suggested Companies for Industrial Visits:

- **Electronic Arts (EA)** – Renowned for game development.
- **Game Loft** – Mobile game production and design.
- **Prokarma** – IT solutions and services.
- **Cipher Cloud** – Cyber security and cloud data protection solutions.

These companies offer insights into software development, application lifecycle management, quality assurance, and cutting-edge technologies.

S.No	Type of industry	Nature of industry	Date of visit (Tentative Date)	No. of students participated	Year/branch	remarks
1						
2						

SHADOW ENGINEERING

GUIDE LINES FOR SHADOW ENGINEERING (VIP)

11. INDUSTRIAL VISITS (IIP–INNOVATIVEINDUSTRIALLEARNING PROGRAM)

OBJECTIVES OF SHADOW ENGINEERING:

1. To uplift the knowledge of the students related to laboratories.
2. To improve the industry-college in tractions.
3. Tocreateindustrylikeenvironmentforallthestudentsinordertomakefuture Assignment.
4. This program leads to matrixing with the students.

TABLE2: INDUSTRIAL TRAINING (Shadow Engg)

(Career Visit Approval)

S.No	Name of the urse	Nature of industry	Duration of aining	Authority	Date of Training/Certificate .	Remarks
1	Programming Python	Software				

13. Calibration, Testing & Inspection (Method &Tools to be used)

Calibration

There is no need of any equipment to be calibrated.

Testing & Inspection:

Installation: Aim of this concept is to make and maintain installation procedure for a new equipment or already existing equipment.

Introduction

Python is a widely used high-level programming language first launched in 1991. Since then, Python has been gaining popularity and is considered as one of the most popularand flexible server-side programming languages.

Unlike most Linux distributions, Windows does not come with the Python programming languagebydefault.However,youcaninstallPythoninanyversionofoperatingsystems.

Python Installation steps:(UbuntuLinuxGUI)

Step 1: Select Version of Python to Install.
Step2: Download Python Executable Installer.
Step 3: Run Executable Installer.
Step4:Verify Python Was Installed On Windows.
Step 5: Verify Pip Was Installed.
Step6:Add Python Path to Environment Variables(Optional)

14. MAINTAINANCEANDTROUBLESHOOTING:

Maintenance:

Maintenance and trouble shooting of each equip mention laboratory must follow the following guidelines:

Maintenance Schedules:

15. Preventive Maintenance Schedules of lab will be decided by lab in charge along with concernedHOD.Thedetailsofscheduleshouldberecordedinthefollowingtemplateof format.

S.No.	Software	Date of Maintenance	Type of Activity	Remarks
1	**Programming in Python**		**Software installed(python 3.10.5)**	

16. MaintenanceReportsshouldbedulysignedbyinchargesaswellasHODsandduly approved by Principal periodically.

TROUBLE SHOOTING SCHEDULES:

Apropos a list obemadefromeachlabbranchwise.Theproposalshouldcarryfollowingdetails related to specific lab.

S.No.,SoftwareTool,TypeofProblem(insufficientmemory,typeofversion,securityissues, authentication issues, Corrupt Software, proper configuration, Missing of Display, CRT not working), Expected Reasons .Trouble shooting exercises should be properly recorded in a separate format as mentioned below:

S.No.	Dateofrecording activity	Software Tool	Type of Trouble	Remedial Activity	Remarks

15. ASSESSMENTANDACCREDITATIONPROCEDUREASPERNABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).

The laboratory should carry out the following important tasks to wards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge
4. Ensure Effective environmental conditions (temperature, humidity, storage and placement)in the laboratories by implementing proper house keeping and cleaningof the equipments from dust, dirt etc.
5. EnsureCalibrationofinstruments/equipment (OnlyNABLaccreditedauthorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusivelyfor calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.

9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in amonthor3monthsorattheend of these mester.Theauditschedule will be decided by the Chairman and Principal of the CLT team.
10. Auditorsshouldsubmitthedetailedreportofeachlabdulysignedbythemtothe Principal.
11. Eachlabshouldmaintainallthebills/invoicesofeachinstrumentorequipmentina separate file.
12. All the stock registers either consumable or non consumable should be updated whenever any purchases of consumables or equipment takes place.
13. All the safety precautions should be properly displayed in front of each lab.
14. AlltheLEADexperimentsshouldbemaintainedseparatelyinarecord/recordedin a separate folder.
15. BasedonthePreAssessmentreportsubmittedbytheauditor,correctiveactionsshould be carried out by each lab in charge and that must be forwarded to concerned HODand Principal.

16. VIRTUALLABSREFERENCELINKS

No virtual labs are available.

CMR ENGINEERING COLLEGE
UGC AUTONOMOUS
Approved by AICTE-New Delhi | Affiliated to JNTUH | Accredited by NAAC & NBA
Department of Computer Science & Engineering (Artificial Intelligence & Machine Learning)

A.Y 2025-2026

LIFT Planner (Lab Improvement for Future Trends)

DATA ANALTICS LAB
(R Programming)

III YEAR I SEM

HOD **PRINCIPAL**

Sl. No	CONTENT
1	**OBJECTIVES AND RELEVANCE**
2	**SCOPE**
3	**PREREQUISITES**
4	**SYLLABUS AS PER JNTUH**
5	**LEAD EXPERIMENTS**
6	**LAB SCHEDULE**
7	**SUGGESTED BOOKS**
8	**WEBSITES (USEFUL LINKS)**
9	**EXPERT DETAILS**
10	**MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS**
11	**INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)**
12	**SHADOW ENGINEERING**
13	**CALIBIRATION , TESTING AND INSPECTION**
14	**MAINTAINANCE AND TROUBLESHOOTING**
15	**ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL**
16	**VIRTUAL LABS REFERENCE LINKS**

1. OBJECTIVESANDRELEVANCE

- The main objective of the lab course is to understand the fundamental concepts of data analytics. .
- Students can understand about principles and methods of statistical analysis.
- Students can come to know about various search methods and visualization techniques.

2. SCOPE

The scope of Data Analytics lab using R programming is student can be familiar with concepts of Analyzing the data and data preprocessing techniques.

3. PREREQUISITES

Student need to have the theoretical knowledge on basics of Data Communications, Programming language in C and Web Technologies.

4. CMREC SYLLABUS

1. Data Preprocessing
   a. Handling missing values
   b. Noise detection removal
   c. Identifying data redundancy and elimination
2. Implement any one imputation model
3. Implement Linear Regression
4. Implement Logistic Regression
5. Implement Decision Tree Induction for classification
6. Implement Random Forest Classifier
7. Implement ARIMA on Time Series data
8. Object segmentation using hierarchical based methods
9. Perform Visualization techniques (types of maps - Bar, Colum, Line, Scatter, 3D Cubes etc)
10. Perform Descriptive analytics on healthcare data
11. Perform Predictive analytics on Product Sales data
12. Apply Predictive analytics for Weather forecasting.

5. Lead Experiments

1. Write R program to build clustering model using K-mean algorithm
2. Visualize a correlation heatmap along with correlation coefficients on the map using the "value" column in the correlation matrix as text. Using geomtext( ) function annotations can be added on the heatmap and use "value" as labels.

6. LAB SCHEDULE

Week No.	Name of the Program	No. of Lab sessions	Text Books	Unit number	Mode of Assessment
1	Basic programs	1	T1		
2	Data Preprocessing a. Handling missing values b. Noise detection removal c. Identifying data redundancy and elimination	1	T1	Unit-I	Viva & Execution
3	Implement any one imputation model	1	T1	Unit-II	Viva & Execution
4	Implement Linear Regression	1	T1	Unit-III	Viva & Execution
5	Implement Logistic Regression	1	T1	Unit-III	Viva & Execution
6	Implement Decision Tree Induction for classification	1	T1	Unit-IV	Viva & Execution
7	Implement Random Forest Classifier	1	T1	Unit-IV	Viva & Execution
8	Implement ARIMA on Time Series data	1	T1	Unit-IV	Viva & Execution
9	Object segmentation using hierarchical based methods	1	T1	Unit-IV	Viva & Execution
10	Perform Visualization techniques (types of maps - Bar, Colum, Line, Scatter, 3D Cubes etc)	1	T1	Unit-IV	Viva & Execution
11	Perform Descriptive analytics on healthcare data.	1	T1	Unit-V	Viva & Execution
12	Perform Predictive analytics on Product Sales data	1	T1	Unit-V	Viva & Execution
13	Apply Predictive analytics for Weather forecasting	1	T1	Unit-V	Viva & Execution
14	Lead 1 & Lead 2	1	T1		Viva & Execution
	Total no of Lab sessions required to complete syllabus	14			

Linking of Data Analytics lab using (R programming)subject with Lab Experiments

Preamble: This lab covers the Data Analytics lab using R programming subject experiments in C language. The JNTU/CMREC has given 12 experiments in the DAR Lab syllabus.

UNIT-I:

EXPERIMENT NO: 1&2

1. Data Preprocessing

    a. Handling missing values

    b. Noise detection removal

    c. Identifying data redundancy and elimination

2. Implement any one imputation model.

OBJECTIVE:

The main objective of Data Preprocessing and imputation model is to handle the different types of data and preprocess the data which is useful for implementation.

APPLICATIONS:

The above experiments are used for Data Preprocessing checking techniques.

UNIT-II:

EXPERIMENT NO: 3, 4

3. Implement Linear Regression

4. Implement Logistic Regression

OBJECTIVE:

The main objective of Linear Regression is predict a continuous dependent variable based on one or more independent (predictor) variables by fitting a linear relationship

The main objective of Logistic Regression is to predict a categorical dependent variable (usually binary: 0 or 1) using one or more independent variables by modeling the probability of the outcome with a sigmoid (logistic) function**.**

APPLICATIONS:

Linear Regression Applications (Predicting continuous values):

1. **Real Estate**: Predicting house prices based on features like size, location, number of rooms, etc.
2. **Finance**: Forecasting stock prices, revenue, or sales based on historical trends.
3. **Healthcare**: Estimating blood pressure or cholesterol level based on age, weight, and lifestyle factors.
4. **Agriculture**: Predicting crop yield from rainfall, temperature, and soil conditions.
5. **Marketing**: Estimating customer lifetime value or future spending based on past purchases.

Logistic Regression Applications (Classifying categorical outcomes):

1. **Medical Diagnosis**: Predicting whether a patient has a disease (e.g., cancer, diabetes) or not.
2. **Email Filtering**: Classifying emails as spam or not spam.
3. **Banking and Finance**: Predicting loan default (will default or not).
4. **Customer Behavior**: Predicting whether a customer will churn (leave) or stay.
5. **Security**: Identifying fraudulent transactions or unauthorized access.

UNIT-III

EXPERIMENTNO: 5, 6

5. Implement Decision Tree Induction for classification
6. Implement Random Forest Classifier

OBJECTIVE:

The main objective of a Decision Tree is to create a model that predicts the value of a target variable by learning simple decision rules inferred from data features.

APPLICATIONS:

1. **Medical Diagnosis**: Predict diseases based on symptoms (e.g., flu vs. COVID-19).
2. **Credit Scoring**: Approve or deny loans based on income, credit history, etc.
3. **Customer Segmentation**: Group customers by purchasing behavior.
4. **Manufacturing**: Determine reasons for product defects or equipment failure.
5. **Education**: Predict student performance based on attendance, grades, etc.

OBJECTIVE:

The main objective of Random Forest is to improve the accuracy and robustness of predictions by combining multiple decision trees through ensemble learning.

APPLICATIONS:

1. **Fraud Detection:** Identify fraudulent transactions in banking and e-commerce.
2. **Healthcare: Classify** cancer types from gene expression or imaging data.
3. **E-commerce: Predict** product recommendations and customer churn.
4. **Environmental Science:** Predict air pollution levels or weather patterns.

UNIT-IV

EXPERIMENTNO: 7, 8

7. Implement ARIMA on Time Series data
8. Object segmentation using hierarchical based methods

Objectives of ARIMA:

1. **Forecasting Time Series Data**: Predict future values based on past observations.
2. **Modeling Temporal Dependencies**: Capture autocorrelations, trends, and seasonality in time-series data.

Applications:

1. **Finance:** Stock price prediction, risk assessment
2. **Retail:** Sales forecasting, inventory demand prediction
3. **Weather:** Temperature and rainfall forecasting
4. **Healthcare:** Forecasting disease incidence rates over time

Objectives of Object Segmentation:

1. **Identify and Classify Objects at the Pixel Level**: Distinguish different objects within an image.
2. **Localize Objects Accurately**: Provide precise outlines (masks) for each object.
3. **Enable Detailed Scene Understanding**: Understand not just what is in the image, but where it is and how it interacts.

Applications:

1. Autonomous Vehicles Lane, pedestrian, and vehicle detection
2. Medical Imaging Segmenting tumors, organs (e.g., in MRI/CT scans)
3. Agriculture Identifying and counting crops, detecting disease

UNIT-V

EXPERIMENTNO: 9

9. Perform Visualization techniques (types of maps - Bar, Colum, Line, Scatter, 3D Cubes Etc)

APPLICATIONS:

1. **Business Analytics:** Dashboards showing sales performance, customer behavior, KPIs
2. **Finance:** Stock market trends, portfolio analysis, risk visualization
3. **Healthcare:** Patient health records, disease progression, medical imaging
4. **Education:** Student performance tracking, course completion rates

UNIT-V

EXPERIMENTNO: 10, 11, 12

.

10. Perform Descriptive analytics on healthcare data
11. Perform Predictive analytics on Product Sales data
12. Apply Predictive analytics for Weather forecasting

Objectives of Descriptive Analytics: It aims to summarize and interpret historical data to understand past events and trends.

Objective of Predictive Analytics: It aims to forecast future outcomes using historical data and statistical or machine learning models.

Applications:

1. **Business** **:** Monthly sales reports, customer purchase patterns
2. **Healthcare** **:** Hospital readmission statistics, patient visit summaries
3. **Education** **:** Student attendance and grade analysis
4. **Finance** **:** Historical transaction summaries, profit/loss statements

6(a) LAB SCHEDULE: The lab schedule should be planned once in a week. The week wise scheduled experiment should be completed.

CYCLE1

Batches	week- 1	week-2	week-3	week-4	week-5	week- 6	Week-7	Week- 8
B1	Demo & Basic Programs	Exp1	Exp2	Exp3	Exp4	Exp5	Exp6	Exp7
B2	Demo & Basic Programs	Exp1	Exp2	Exp3	Exp4	Exp5	Exp6	Exp7

CYCLE2

Batches	week- 9	week-10	week-11	week-12	week-13	week- 14
B1	EXP8	Exp9	Exp10	Exp11	Exp12	LEAD 1 & 2
B2	EXP8	Exp9	Exp10	Exp11	Exp12	LEAD 1 & 2

6(b) VIVA SCHEDULE: The viva schedule should be planned prior starting to the lab experiment.

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10	week-11	week-12	week-13	week-14
B1, B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva
B1, B2	viva	viva	viva	viva	viva	viva	Viva	viva	viva	viva	viva	Viva	viva	viva

CYCLE 1: **ROUND–**1

Batches	**week-1**	**week-2**	**week-3**	**week-4**	**week-5**	**week-6**	**week-7**	**week-8**
B1,B2	viva		viva		viva		viva	
B1,B2		viva		viva		viva		viva

Batches	**week-1**	**week-2**	**week-3**	**week-4**	**week-5**	**week-6**	**week-7**	**week-8**
B1,B2	viva		viva		viva		viva	
B1,B2		viva		viva		viva		viva

*SG: Selected Group with a maximum of 6 or 12 students

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10
SG1	viva									
SG2			viva							
SG3					viva					
SG4							viva			
SG5									viva	

(C) Scheme of Evaluation: The scheme of evaluation for the lab internal and lab external as follows

LAB INTERNAL:									
Day to Day Evalution-10							**Internal Exam-20**		
Uniform	**Observation**	**Record**	**Performance Of the Experiment**	**Result**	**Diagrams/ Figures**	**Viva**	**Aim/ Write up**	**Execution &Results**	**Result & Conclusions**
Marks-2	**Marks-2**	**Marks-2**	**Marks-2**	**Marks- 2**	**Marks- 2**	**Marks - 10**	**Marks-3**	**Marks-3**	**Marks-2**
Total Marks-30									

LAB EXTERNAL			
S.NO	**Write up**	**Final Evaluation**	**Viva**
1	1. Aim 2. Procedure 3. Program 4. Expected output.	Based on correctness of the program and Results	Based on understanding of Experiment and theoretical questions in the related subjects
Marks	**25**	**25**	**10**
Total Marks-60			

7. SUGGESTEDBOOKS:

1. Student's Handbook for Associate Analytics – II, III.
2. Data Mining Concepts and Techniques, Han, Kamber, 3rd Edition, Morgan KaufmannPublishers.

8. WEBSITES (USEFULLINKS):

1. https://www.youtube.com/watch?v=_V8eKsto3Ug
2. https://www.youtube.com/user/joshstarmer
3. https://posit.co/resources/cheatsheets/
4. https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf
5. https://cran.r-project.org/manuals.html
6. https://cran.r-project.org/doc/manuals/r-release/R-data.pdf

9. EXPERTS DETAILS

The expert details which have been mentioned below are only afew of the eminent ones known Internationally, Nationally and Locally.

INTERNATIONAL

1. Hadley Wickham (USA)

Affiliation: Chief Scientist at Posit (formerly RStudio)
Expertise: Data science, tidyverse, ggplot2, dplyr
Famous Books:
1. R for Data Science (with Garrett Grolemund)

2. Advanced R

3. ggplot2: Elegant Graphics for Data Analysi

https://hadley.nz/

2. Garrett Grolemund (USA)

Affiliation: Data Scientist at Posit
Expertise: Teaching R, data wrangling, R Shiny
Famous Book:

- *Hands-On Programming with R*
- *R for Data Science* (co-authored with Hadley Wickham)

https://rstudio-education.github.io/

NATIONAL

1. Prof. Shalabh

Affiliation: IIT Kanpur
Expertise: Statistics, R Programming, Regression Analysis
Contributions:

- Instructor for R-based statistics courses on NPTEL
- Popular course: *Introduction to R Software*

https://nptel.ac.in/courses/111104100

2. Dr. G. Jay Kerns (USA–India Collaborator)

Book:*Introduction to Probability and Statistics Using R*

- Freely available and widely recommended for Indian universities

https://cran.r-project.org/doc/contrib/Kerns-prob.pdf

REGIONAL

1. Dr. Seema Acharya & Subhashini Chellappan

Book: *Data Analytics Using R*

Publisher: Wiley India

Highlights: Covers basics of R, data manipulation, statistics, visualization, and machine learning.

2. Kavya R.

Book: *Data Analytics Using R*

Publisher: Cengage India

Highlights: Indian-centric examples, exercises for students, aligned with university curriculum.

10. MAPPING OF LAB WITHPROJECT/CONSULTANCY/R&D:

PROPOSAL FOR PROJECT ACTIVITY:

Sales Forecasting Using Time Series Analysis

ABSTRACT

Sales forecasting is a critical component of strategic planning and decision-making in any business. This project aims to develop an accurate sales prediction model using time series analysis techniques such as ARIMA (Auto Regressive Integrated Moving Average), Exponential Smoothing, and Seasonal Decomposition. By analyzing historical sales data, the model identifies trends, seasonal patterns, and irregular fluctuations to forecast future sales. Accurate forecasting helps businesses manage inventory, optimize supply chains, and improve budgeting processes. The project also includes data preprocessing, visualization of sales trends, model evaluation, and error analysis to enhance forecast reliability. The ultimate goal is to provide a robust, data-driven approach for predicting sales that supports better business decisions and drives growth.

Hobby Projects:

1. Personal Finance Tracker
2. Netflix Viewing Trends
3. Analyze your watch history to find genre preferences and binge patterns.
4. Fitness Tracker Analysis
5. Use data from apps (e.g., Google Fit, Fit bit) to study steps, sleep, and heart rate trends.
6. Weather Pattern Analysis
7. Use open APIs to explore temperature and rainfall trends in your city.
8. Social Media Activity Insights

11. INDUSTRIAL VISITS(IIP–INNOVATIVEINDUSTRIALLEARNINGPROGRAM):

OBJECTIVES OF SHADOW ENGINEERING:

1. The program which uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future assignment.
4. This program leads to matrixing with the students.

12. SHADOW ENGINEERING

- To uplift the knowledge of the students related to laboratories.
- To improve the industry-college interactions.
- To create industry like environment for all the students in order to make future Assignment.

INDUSTRIAL TRAINING (Shadow Engg)

(Career Visit Approval)

S.no	Name of the Course	Nature of industry	Duration of Training	Authority	Date of Training/Certificate No.	Remarks
1	MNC	software	April month	20 to 30	CSE – II Year	
2	MNC	software	June	20 to 30	CSE – II Year	

13. CALIBIRATION, TESTING AND INSPECTION

Calibration

There is no need of any physical equipment to be calibrated but we use some datasets along with the R programming tool or R studio for implementation.

Testing & Inspection:

We use datasets to perform testing and implementation

14. MAINTAINANCE AND TROUBLESHOOTING:

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

1. Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format

S.No.	Software	Date of Maintenance	Type of Activity	Remarks
1	Operating System		Software installed	
2	R studio		Installation	

2. Maintenance Reports duly signed by in charges as well as HODs and duly approved by Principal periodically.

15. NABL (National Accreditation Board for Testing and Calibration Laboratories)

ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).
The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge

4. Ensure Effective environmental conditions (temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipments from dust, dirt etc.
5. Ensure Calibration of instruments/equipment (Only NABL accredited authorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team.
   Auditors should submit the detailed report of each lab duly signed by them to the Principal.

The following Data Analytics vendors are providing certification for accreditation

- Microsoft Power BI Data Analyst Associate (PL-300)
- Google Data Analytics Professional Certificate
- IBM Data Analyst Professional Certificate
- Microsoft Certified: Azure Data Scientist Associate

16. Virtual Lab's:

No virtual labs are available.

AY 2025-2026

LIFT Planner (Lab Improvement for Future Trends)

On

DEEP LEARNING LAB

IV -YEAR SEM-I

HOD **PRINCIPAL**

Sl. No	CONTENT
1	OBJECTIVES AND RELEVANCE
2	SCOPE
3	PREREQUISITES
4	SYLLABUS AS PER CMREC
5	LEAD EXPERIMENTS
6	LAB SCHEDULE
7	SUGGESTED BOOKS
8	WEBSITES (USEFUL LINKS)
9	EXPERT DETAILS
10	MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS
11	INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12	SHADOW ENGINEERING
13	CALIBIRATION , TESTING AND INSPECTION
14	MAINTAINANCE AND TROUBLESHOOTING
15	ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL
16	VIRTUAL LABS REFERENCE LINKS

1. **OBJECTIVES AND RELEVANCE**

- To implement ANN and Deep Learning concepts.

2. SCOPE

Deep Learning (DL) has a big scope in future and the biggest reason for it is that DL doesn't require any kind of feature engineering. Deep Learning extracts the features from the data itself instead of us giving it the features after extracting it from the data. This way it solves our biggest problem of feature engineering. Also, since features are learned by the model itself, it has a better probability of producing a model which is more generalized than the feature engineered models. These reasons alone are sufficient to prefer DL over other technologies. With the recent development in DL, now we have more state-of-the-art results on various tasks including Natural Language Processing, Language Translation, Automatic Speech Recognition, Multi Label Image Recognition/Classification, Language generation among others, AI has grown stronger in past 8 years with DL than it has developed in 20 years before that. So, if you are looking for a future in in Deep Learning, you are looking in right direction.

3. PRE REQUISITES

Any programming language Python/R-programming/Mat lab

4. SYLLABUS

1. Implementation of MP model
2. Implementation of feed forward neural network
3. Implementation of back propagation neural network
4. Implement all activation function of neural network for any pattern recognition application
5. Implement any one of Image Net or Google Net
6. Implement a system to recognize hand written character using CNN
7. Classify images appropriately using CNN
8. Implement LSTM Neural Network for Time Series Prediction

5. LEAD EXPERIMENTS

1. Applying the Deep Learning Models in the field of Natural Language Processing
2. Applying the Auto encoder algorithms for encoding the real-world data

6. **LAB SCHEDULE**

CYCLE 1:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8
B1	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B2	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B3	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B4	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B5	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B6	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B7	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B8	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3

Batches	week-9	week-10	week-11	week-12	week-13	week-14	week-15	week-16
B1	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B2	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B3	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B4	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B5	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B6	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B7	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B8	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD

CYCLE 2:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8
B9	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B10	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B11	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B12	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B13	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B14	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B15	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3

Batches	week-9	week-10	week-11	week-12	week-13	week-14	week-15	week-16
B9	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B10	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B11	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B12	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B13	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B14	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B15	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD

2. **(B) VIVA SCHEDULE:** The viva schedule should be planned prior starting to the lab experiment.

CYCLE 1: **ROUND -** 1

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10	week-11	week-12	week-13	week-14
B1,B2,B3,B4	viva				viva				viva					
B5, B6,B7, B8		viva				viva				viva				
B1,B2,B3,B4			viva				viva				viva			
B5,B6,B7,B8				viva				Viva				viva		

CYCLE 2: **ROUND –** 1

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10	week-11	week-12	week-13	week-14
B9,B10,B11,B12	viva				viva				viva					
B13,B14,B15		viva				viva				viva				
B9,B10,B11,B12			viva				viva				viva			
B13,B14,B15				viva				Viva				viva		

ROUND – 2

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8	week-9	week-10
SG1	viva									
SG2			viva							
SG3					viva					
SG4							Viva			
SG5									viva	

LAB INTERNAL:							
Day to Day Evalution-15					**Internal Exam-10**		
Uniform	**Observation & Record**	**Performance Of the Experiment**	**Result**	**Viva**	*SG: Selected Group with a maximum of 6 or 12 students **Write up**	**Execution & Results**	**Viva**
Marks-3	**Marks-3**	**Marks-3**	**Marks-3**	**Marks-5**	**Marks-5**	**Marks-4**	**Marks-3**
Total Marks-30							

SCHEME OF EVALUATION OF LABS

LAB EXTERNAL:			
S.NO		**Final Evaluation**	**Viva**
1	1. Aim 2. Procedure 3. Program 4. Expected output.	Based on correctness of the program and Results	Based on understanding of Experiment and theoretical questions in the related subjects
Marks	**25**	**25**	**10**
Total Marks-60			

7. SUGGESTED BOOKS:

TEXT BOOKS

1. Deep Learning: An MIT Press Book By IanGood fellow andYoshua Bengio and Aaron Courville.
2. Neural Networks and Learning Machines, Simon Haykin, 3rd Edition, Pearson Prentice Hall.

8. WEBSITES (USEFUL LINKS)

- https://www.geeksforgeeks.org/introduction-deep-learning/
- https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network
- https://www.javatpoint.com/deep-learning
- https://nptel.ac.in/courses/106106184
- https://www.simplilearn.com/tutorials/deep-learning-tutorial/what-is-deep-learning

9. List of Expert Details

The Expert Details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally. There are a few others known as well.

International: Universite de Montreal Pavillon Andre-Aisenstadt Phone: (514) 343-6111 ext. 1868

1. Aaron C. Courville, Departement dInformatique et de recherche operationnelle

   email: courvila@iro.umontreal.ca

   website: http://aaroncourville.wordpress.com

2. Yoshua Bengio, Professor of computer science, University of Montreal, Mila, IVADO, CIFAR,

   Phone: +393463904241,

   Email: yoshua.bengio@gmail.com

National:

1. Debarka Sengupta, Associate Professor, IIIT Delhi, Research areas: Computational genomics, Machine Learning.

Contact: Phone: 011-26907446,
Email: debarka@iiitd.ac.in,
Website: www.debarka.com

2. Anoop M. Namboodiri, Associate Professor, Ph.D (Michigan State University, USA),International Institute of Information Technology(IIIT-H), Hyderbad. Research Areas:Pattern recognition, machine learning, coputer vision, biometrics

Email: anoop@iiit.ac.in,

Contact : - (91) (40) 6653 1000 Ext:1326 Ext: 1326

3. Prof. Vineeth N Baasubramanian(IIIT Hyderabad), Computer Science and Engineering,

Phone:+914023016357,
Email: vineethnb@iith.ac.in

Regional:

1. Konda Reddy Mopuri Assistant Professor at the Department of Artificial Intelligence, Indian Institute of Technology Hyderabad. Office (214E) is located in the C-block.

2.Prof Krishna Mohan heads Visual Learning and Intelligence Lab (VIGIL) at IIT Hyderabad working in the research areas of video content analysis, computer vision, machine learning, deep learning. He is a recipient of the Excellence in Teaching Award**,**

Address : Room No. 103, Block - A
IIT Hyderabad
Kandi, Sangareddy-502285.
Telangana (State), India.

Phone : +91 - 94917 12312 (Mobile)
040- 2301 6021(Off)

10. MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS

1. Breast Cancer Prediction
2. Human Activity Recognition with Smart phones
3. Breast Cancer Prediction

11. INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)

Students visit schedule different software development MNC companies.

S.No	Type of industry	Nature of industry	Date of visit (Tentative Date)	No. of students participated	Year/branch	remarks
1						
2						

12. SHADOW ENGINEERING

1. The program which uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

13. CALIBIRATION, TESTING AND INSPECTION

Calibration:

Calibration: There is no need to calibrate any equipment for this concerned lab.
Installation: Installation Jupiter Notebook

14. MAINTAINANCE AND TROUBLESHOOTING

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

S.No.	Software	Date of Maintenance	Type of Activity	Remarks
1	**python**		**Software installed**	

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

15. ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).

The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge
4. Ensure Effective environmental conditions (temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipment s from dust, dirt etc.
5. Ensure Calibration of instruments/equipment (Only NABL accredited authorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.

7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team. Auditors should submit the detailed report of each lab duly signed by them to the Principal.

16. VIRTUAL LABS REFERENCE LINKS

No virtual labs are available.

CMR ENGINEERING COLLEGE
UGC AUTONOMOUS
Approved by AICTE-New Delhi | Affiliated to JNTUH | Accredited by NAAC & NBA
Department of Computer Science & Engineering (Artificial Intelligence & Machine Learning)

AY 2025-2026

LIFT Planner (Lab Improvement for Future Trends)

On

CLOUD COMPUTING LAB

IV -YEAR SEM-I

HOD **PRINCIPAL**

Sl. No	CONTENT
1	OBJECTIVES AND RELEVANCE
2	SCOPE
3	PREREQUISITES
4	SYLLABUS AS PER CMREC
5	LEAD EXPERIMENTS
6	LAB SCHEDULE
7	SUGGESTED BOOKS
8	WEBSITES (USEFUL LINKS)
9	EXPERT DETAILS
10	MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS
11	INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)
12	SHADOW ENGINEERING
13	CALIBIRATION, TESTING AND INSPECTION
14	MAINTAINANCE AND TROUBLESHOOTING
15	ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL
16	VIRTUAL LABS REFERENCE LINKS

1. **OBJECTIVES AND RELEVANCE**

- To implement distributed system models, different cloud service models, service-oriented architectures, cloud programming and software environments, resource management.

2. SCOPE

Cloud computing has revolutionized the way computing resources are delivered and consumed. Its scope spans across multiple domains, industries, and technologies. Business and Enterprise Applications Companies no longer need to invest heavily in hardware; instead, they can lease virtual infrastructure (IaaS), Software Development and Testing Seamless pipelines using tools like Jenkins, GitHub Actions, and Azure DevOps. Auto-scaling Environments Developers can scale resources dynamically during testing or deployment phases. Real-Time Analytics Cloud platforms (e.g., AWS Redshift, Google BigQuery) offer high-performance tools for data processing, Remote Classrooms Platforms like Google Classroom and Zoom rely on cloud services, Digital India & Smart Cities Indian initiatives leverage cloud for transparency and scalability.

3. PRE REQUISITES

Understanding IP addressing, DNS, DHCP, NAT, firewall rules, etc.

4. SYLLABUS

PART-A

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing.

PART-B

LIST OF EXPERIMENTS:

1. Install Virtualbox/VMware Workstation with different flavors of Linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
3. Create an Amazon EC2 instance and set up a web-server on the instance and associate an IP address with the instance.
4. Install Google App Engine. Create a hello world app and other simple web applications using python/java.

5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
8. Install Hadoop single node cluster and run simple applications like word count.
9. Create a database instance in the cloud using Amazon RDS.
10. Create a database instance in the cloud using Google Cloud SQL.

5. LEAD EXPERIMENTS

1. Creating and Configuring an Amazon EC2 Instance with Web Server
2. Install Hadoop Single Node Cluster and Run Word Count Application

6. LAB SCHEDULE

CYCLE 1:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8
B1	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B2	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B3	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B4	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B5	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B6	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B7	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B8	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3

Batches	week-9	week-10	week-11	week-12	week-13	week-14	week-15	week-16
B1	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B2	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B3	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B4	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B5	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B6	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B7	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B8	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD

CYCLE 2:

Batches	week-1	week-2	week-3	week-4	week-5	week-6	week-7	week-8
B9	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B10	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B11	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B12	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B13	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B14	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3
B15	Demo & Introduction	Exp.1	Exp.1	Exp.1	Exp.2	Exp.2	Exp.2	Exp.3

Batches	week-9	week-10	week-11	week-12	week-13	week-14	week-15	week-16
B9	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B10	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD

B11	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B12	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B13	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B14	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD
B15	Exp.3	Exp.3	Exp.4	Exp.4	Exp.4	LEAD	LEAD	LEAD

2. **(B) VIVA SCHEDULE:** The viva schedule should be planned prior starting to the lab experiment.

CYCLE 1: **ROUND -** 1

Batches	**week-1**	**week-2**	**week-3**	**week-4**	**week-5**	**week-6**	**week-7**	**week-8**	**week-9**	**week-10**	**week-11**	**week-12**	**week-13**	**week-14**
B1,B2,B3,B4	viva				viva				viva					
B5, B6,B7, B8		viva				viva				viva				
B1,B2,B3,B4			viva				viva				viva			
B5,B6,B7,B8				viva				Viva				viva		

CYCLE 2: **ROUND –** 1

Batches	**week-1**	**week-2**	**week-3**	**week-4**	**week-5**	**week-6**	**week-7**	**week-8**	**week-9**	**week-10**	**week-11**	**week-12**	**week-13**	**week-14**
B9,B10,B11,B12	viva				viva				viva					
B13,B14,B15		viva				viva				viva				
B9,B10,B11,B12			viva				viva				viva			
B13,B14,B15				viva				Viva				viva		

ROUND – 2

SCHEME OF EVALUATION OF LABS

LAB INTERNAL:									
Day to Day Evalution-10							**Internal Exam-20**		
Uniform	**Observation**	**Record**	**Performance Of the Experiment**	**Result**	**Diagrams /Figures**	**Viva**	**Aim/ Write up**	**Execution &Results**	**Result & Conclusions**
Marks-2	**Marks-2**	**Marks-2**	**Marks-2**	**Marks-2**	**Marks- 2**	**Marks-10**	**Marks - 3**	**Marks-3**	**Marks-2**
Total Marks-30									

Batches	**week-1**	**week-2**	**week-3**	**week-4**	**week-5**	**week-6**	**week-7**	**week-8**	**week-9**	**week-10**
SG1	viva									
SG2			viva							
SG3					viva					
SG4							Viva			
SG5									viva	

LAB EXTERNAL			
S.NO	**Write up**	**Final Evaluation**	**Viva**
1	1. Aim 2. Procedure 3. Program 4. Expected output.	Based on correctness	Based on understanding of Experiment and theoretical questions in the related subjects
Marks	**25**	**25**	**10**
Total Marks-60			

7. SUGGESTED BOOKS:

TEXT BOOKS

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014

REFERENCE BOOKS:

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej
   M. Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

8. WEBSITES (USEFUL LINKS)

- https://www.geeksforgeeks.org/cloud-computing/cloud-computing/
- https://www.techtarget.com/searchcloudcomputing/tip/Explore-edge-computing-services-in-the-cloud
- https://www.tpointtech.com/cloud-computing-definition
- https://nptel.ac.in/courses/106105167
- https://www.simplilearn.com/how-to-start-career-in-cloud-computing-article

9. List of Expert Details

The Expert Details which have been mentioned below are only a few of the eminent ones known Internationally, Nationally and Locally. There are a few others known as well.

International:

1. **Dr. Pranav Verma,** Associate Professor, Uppsala University, Sweden, Mail id- pranav.verma@angstrom.uu.se, Mobile No.- 9601406311.

2. **Mr. Abhishek Verma,** Data Scientist, Autodesk Inc San Francisco USA, Mail id – abhishek.verma@autodesk.com, mobile No.- (+1)-4124-626-1803

National:

1. **Dr. J. P. Patra,** Associate Professor, UDT CSVTU, Bhilai, Mail id- jppatra@gmail.com, Mobile No.-8319527740

2. **Dr. Toran Verma,** Associate Professor, UDT CSVTU, Bhilai, Mail id- vermatoran@gmail.com, Mobile No.-9770404044

Regional:

1. Anoop M. Namboodiri, Associate Professor, Ph.D (Michigan State University, USA), International Institute of Information Technology (IIIT-H), Hyderbad.
   Research Areas: Pattern recognition, machine learning, computer vision, biometrics
   Email: anoop@iiit.ac.in

2. Prof. Vineeth N Baasubramanian (IIIT Hyderabad), Computer Science and Engineering,
   Email: vineethnb@iith.ac.in

10. **MAPPING OF LAB WITH PROJECT/CONSULTANCY/R & D PROPOSALS**

1. Breast Cancer Prediction
2. Human Activity Recognition with Smart phones
3. Breast Cancer Prediction

11. INDUSTRIAL VISITS (IIP – INNOVATIVE INDUSTRIAL LEARNING PROGRAM)

Students visit schedule different software development MNC companies.

S.No	Type of industry	Nature of industry	Date of visit (Tentative Date)	No. of students participated	Year/branch	remarks
1						
2						

12. SHADOW ENGINEERING

1. The program which uplifts the knowledge of the students related to laboratories.
2. To improve the industry-college interactions.
3. To create industry like environment for all the students in order to make future Assignment.
4. This program leads to matrixing with the students.

13. CALIBIRATION, TESTING AND INSPECTION

Calibration:

Calibration: There is no need to calibrate any equipment for this concerned lab.
Installation: Installation Jupiter Notebook

14. MAINTAINANCE AND TROUBLESHOOTING

Maintenance:

Maintenance and trouble shooting of each equipment in a laboratory must follow the following guidelines:

Maintenance Schedules:

(1) Preventive Maintenance Schedules of lab will be decided by lab in charge along with concerned HOD. The details of schedule should be recorded in the following template of format.

S.No.	Software	Date of Maintenance	Type of Activity	Remarks
1	VM Software		Software installed	

(2) Maintenance Reports should be duly signed by in charges as well as HODs and duly approved by Principal periodically.

15. ASSESSMENT AND ACCREDITATION PROCEDURE AS PER NABL

Accreditation is the formal recognition, authorization and registration of a laboratory that has demonstrated its capability, competence and credibility to carry out the tasks. It provides the feedback to laboratories as to whether they are performing according to the technical competence as per guidelines of NABL (National Accreditation Board for Testing and Calibration Laboratories).

The laboratory should carry out the following important tasks towards getting ready for accreditation from NABL.

1. Preparation of methodology for each experiment
2. Preparation of Standard Operating procedure for each equipment
3. Preparation of Laboratory Manual as per the guidelines specified by Combined Lab Team(CLT) headed by Principal/HOD/Dean/in charge
4. Ensure Effective environmental conditions (temperature, humidity, storage and placement) in the laboratories by implementing proper housekeeping and cleaning of the equipment s from dust, dirt etc.
5. Ensure Calibration of instruments/equipment (Only NABL accredited authorized laboratories provide calibration).
6. All the details of Calibration should be included in the format specified exclusively for calibration procedure.
7. Ensure proper implementation of all the documents, formats to be included in the lab manual.
8. Impart training for all the technicians working in labs about the importance of documentation, log sheets, operating procedure of the lab.
9. Incorporate Internal Lab audits for effective functioning of the laboratories. Audits may be once in a month or 3 months or at the end of the semester. The audit schedule will be decided by the Chairman and Principal of the CLT team. Auditors should submit the detailed report of each lab duly signed by them to the Principal.

16. VIRTUAL LABS REFERENCE LINKS

No virtual labs are available.