JHARKHAND Rai University
RANCHI

LAB MANUAL

B. TECH (CSE)

(COMPUTER SCIENCE & ENGINEERING)

(2024-2028) onwards

JHARKHAND Rai University RANCHI

Vision

To develop the Department of Computer Science & Information Technology as a Center for Excellence to produce leading Professionals who can serve the society with innovative skills, Computer Experts, Researchers to meet the needs of the software industry in national /global scenario responding to the challenges of ever changing world.

Mission

We endeavor to provide the best possible learning environment to enhance innovations, research capabilities, problem solving skills, leadership qualities, team spirit and ethical responsibilities.

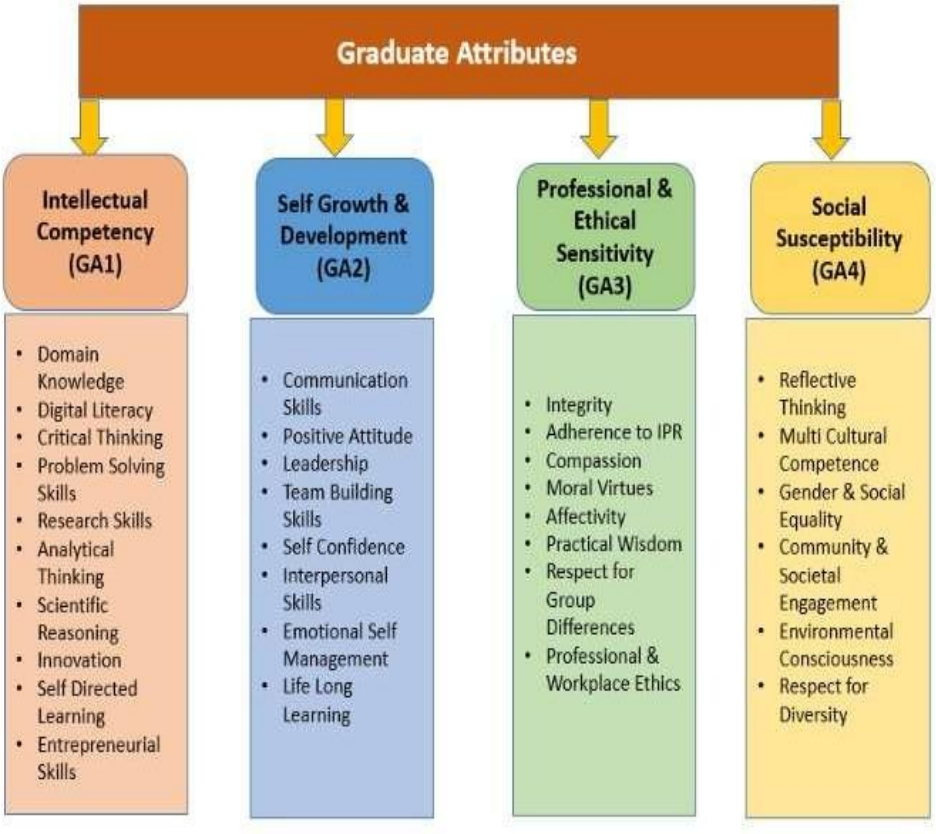
To nurture the talent of the students to be successful, ethical and effective problem solvers who will contribute positively to the economic growth of the nation and prepare to respond to the challenges.

Graduate Attributes

Jharkhand Rai University is a mecca of transformative education which strongly believes in the holistic development of students. The university provides the cutting-edge of holistic learning to develop promising youngsters into leaders of tomorrow with globally relevant, future-ready and actionable intelligence. The objective of the Department is to make each student proficient in synthesizing/analysing information and be ethical, socially responsible, and just when making decisions. JRU ensures inclusive and equitable quality education and promote lifelong learning opportunities for all.

Every graduate of the Department will be developed to possess the following attributes:

1. Intellectual Competency
2. Self-Growth & Development
3. Professional & Ethical Sensitivity
4. Social Susceptibility



Program: B.Tech
Semester: First
Course: Physics I Lab
Course Code: 3BSC101P

L	T	P	C
0	0	2	1

List of Laboratory Experiments/Demonstrations:

1. Find the acceleration due to gravity using Kater's pendulum.
2. Find the resistance of a given wire using Meter Bridge.
3. To establish the current voltage relationship for a metallic conductor and find its resistance.
4. To determine the unknown resistance of given wire using Potentiometer.
5. Find the acceleration due to gravity using Simple pendulum.
6. To determine Young' modulus of the material of a given wire using Searle's Apparatus
7. To determine the unknown resistance of given wire using Potentiometer.
8. To determine the wavelength of sodium light by Newton's ring method.
9. To measure the wavelength of Mercury spectrum using Spectrometer grating.
10. To study the variation of magnetic field with distances using Helmholtz Galvanometer.

Program: B.Tech
Semester: First
Course: Basic Electrical Engineering Lab
Course Code: 3ESC101P

L	T	P	C
0	0	2	1

List of Laboratory Experiments/Demonstrations:

1. Basic safety precautions. Introduction and use of measuring instruments – voltmeter, ammeter, multi-meter, oscilloscope. Real-life resistors, capacitors and inductors.

2. Measuring the steady-state and transient time-response of R-L, R-C, and R-L-C circuits to a step change in voltage (transient may be observed on a storage oscilloscope). Sinusoidal steady state response of R-L, and R-C circuits – impedance calculation and verification. Observation of phase differences between current and voltage. Resonance in R-L-C circuits.

3. Transformers: Observation of the no-load current waveform on an oscilloscope (non-sinusoidal wave-shape due to B-H curve nonlinearity should be shown along with a discussion about harmonics). Loading of a transformer: measurement of primary and secondary voltages and currents, and power.

4. Three-phase transformers: Star and Delta connections. Voltage and Current relationships (line-line voltage, phase-to-neutral voltage, line and phase currents). Phase-shifts between the primary and secondary side. Cumulative three-phase power in balanced three-phase circuits.

5. Demonstration of cut-out sections of machines: dc machine (commutator-brush arrangement), induction machine (squirrel cage rotor), synchronous machine (field winging - slip ring arrangement) and single-phase induction machine.

6. Torque Speed Characteristic of separately excited dc motor.

7. Synchronous speed of two and four-pole, three-phase induction motors. Direction reversal by change of phase-sequence of connections. Torque-Slip Characteristic of an induction motor. Generator operation of an induction machine driven at super-synchronous speed.

8. Synchronous Machine operating as a generator: stand-alone operation with a load. Control of voltage through field excitation.

9. Demonstration of (a) dc-dc converters (b) dc-ac converters – PWM waveform (c) the use of dc-ac converter for speed control of an induction motor and (d) Components of LT switchgear.

Program: B.Tech
Semester: First
Course: Engineering Graphics & Design Lab
Course Code: 3ESC102P

L	T	P	C
0	0	2	1

Laboratory Experiments/Demonstrations is to be conducted as per module content.

Program: B.Tech
Semester: Second
Course: Chemistry I Lab
Course Code: 3BSC103P

L	T	P	C
0	0	2	1

List of Laboratory Experiments/Demonstrations:

1. Determination of surface tension and viscosity
2. Thin layer chromatography
3. Ion exchange column for removal of hardness of water
4. Determination of chloride content of water
5. Colligative properties using freezing point depression
6. Determination of the rate constant of a reaction
7. Determination of cell constant and conductance of solutions
8. Potentiometry - determination of redox potentials and emfs.
9. Synthesis of a polymer/drug
10. Saponification/acid value of an oil
11. Chemical analysis of a salt
12. Lattice structures and packing of spheres
13. Models of potential energy surfaces
14. Chemical oscillations- Iodine clock reaction
15. Estimation of Hcl solution supplied titrating it against N/10 Hcl solution.
16. Adsorption of acetic acid by charcoal
17. Use of the capillary viscosimeters to the demonstrate of the isoelectric point as the pH of minimum viscosity for gelatin sols and/or coagulatio n of the white part of egg

Program: B.Tech
Semester: Second
Course: Programming for Problem Solving Lab
Course Code: 3ESC103P

L	T	P	C
0	0	2	1

Tutorial and Lab:

Tutorial 1: Problem solving using computers: Lab1: Familiarization with programming environment

Tutorial 2: Variable types and type conversions: Lab 2: Simple computational problems using arithmetic expressions

Tutorial 3: Branching and logical expressions: Lab 3: Problems involving if-then-else structures

Tutorial 4: Loops, while and for loops: Lab 4: Iterative problems e.g., sum of series

Tutorial 5: 1D Arrays: searching, sorting: Lab 5: 1D Array manipulation

Tutorial 6: 2D arrays and Strings, memory structure: Lab 6: Matrix problems, String operations

Tutorial 7: Functions, call by value: Lab 7: Simple functions

Tutorial 8 &9: Numerical methods (Root finding, numerical differentiation, numerical integration): Lab 8 and 9: Numerical methods problems

Tutorial 10: Recursion, structure of recursive calls: Lab 10: Recursive functions

Tutorial 11: Pointers, structures and dynamic memory allocation Lab 11: Pointers and structures

Tutorial 12: File handling: Lab 12: File operations

Program: B.Tech CSE
Semester: Second
Course: Computer Assembly and Repair
Course Code: 3ESC105

L	T	P	C
1	0	2	2

---

1. Demonstration of Hardware peripherals: CPU, RAM, SMPS, Motherboard, NIC card, Processor, Processor cooling fan, PCI card, HDD.
2. Demonstration of various ports: CPU , VGA port, PS/2 (keyboard, mouse) ,USB, LAN, Speaker, Audio.
3. Identify the Computer Name and Hardware Specification (RAM capacity, Processor type, HDD, 32 bit/ 64 bit)
4. Identify and Troubleshoot the problems of RAM (beep sound with blue screen), SMPS and motherboard (CPU is not switched ON)
5. Configure BIOS settings- disable and enable USB and LAN.
6. Identify, how to recover the hidden files from corrupted pendrive using command.
7. Recover the contents from crashed Hard Disk using Disk Drill software.
8. Install Operating System – Windows family (Windows 7/ Windows 10) and also make partitions.
9. Install Operating System - Unix family (Linux/UBUNTU)
10. Install Application software – python 3.8, MS- Office 2010/2013, MySQL, TOAD, Openoffice, etc.,
11. Install any one of the antivirus software (Avast, Kaspersky, etc.,) and observe the variations before and after installation.
12. Add new Hardware device (keyboard, mouse, Speaker, Microphone)
13. Connect the LCD Projector with Laptop / CPU.
14. Adding additional RAM to the system.(expanding RAM size).
15. Graphic Card insertion.
16. Assemble and Disassemble Desktop System.

References:

1. Dan Gookin ,Troubleshooting & Maintaining Your PC ALL-IN-ONE, 3rd Edition,2017, John Wiley & Sons.
2. Mike Meyers, Scott Jernigan, Dan Lachance, ”CompTIA Fundamentals + Exam Guide (All-in-One), 2nd Edition, 2019, Mc Graw Hill Education.

Web References:

1. https://www.youtube.com/watch?v=ItxwyMR0SnY&list=PLeH4ngtDM7eE-1_mdWuXWyZrI_FMHnyJ0&index=5
2. https://www.cleverfiles.com/howto/crashed-hard-drive-recovery.html

Program: B.Tech
Semester: Third
Course: Analog Electronic Circuits Lab
Course Code: 3ECS201P

L	T	P	C
0	0	4	2

Course Objective:

1. The goal of this course is to introduce and verify basic principles, operation and applications of the various Analog Electronic circuits of Diode, BJT and MOSFET for various functions.
2. To make students understand and analyze the design and working of Operational amplifiers and their configurations.

List of Experiment (Analog Electronics Circuit)

1. Verify & simulate the Ohm's Law for Resistance in series by using LabVIEW.
2. Design and simulate P-N junction diode circuit to verify its I-V characteristics byusing simulation software LabVIEW.
3. Design & Verify the Bipolar Junction Transistor for both Common Emitter &common Base.
4. Design and simulate Zener diode circuit to plot Volt-Ampere characteristics by using LabVIEW.
5. Study of basic properties of operational Amplifier:
   A. Inverting
   B. Non- Inverting Amplifier
6. Design & Simulate the Differentiator & Integrator using operational Amplifier byusing LabVIEW.
7. Study the input and output Characteristics of MOSFET.
8. Design & plot the graph of RC Differentiator and Integrator by using LabVIEW.

Program: B.Tech
Semester: Third
Course: Data Structure & Algorithms Lab
Course Code: 3PCCCS201P

L	T	P	C
0	0	4	2

Program:

1. To read and display n numbers using an array.
2. To find transpose a 3 X 3 matrix.
3. To insert a number at a given location in an array.
4. To delete a number from a given location in an array.
5. To create a linked list
6. To create a linked list and perform insertions:
   a) at beginning b) at end c) before a given node
7. To create a linked list and perform deletions:
   a) from beginning b) from end c) at a given node
8. To create a circular linked list and perform insertion at the beginning of list.
9. To create a circular linked list and perform insertion at the end of list.
10. To perform Push, Pop and Peep operations on a stack.
11. To implement a linear queue.
12. To implement a priority queue.
13. To search an element in an array using linear search technique.
14. To search an element in an array using binary search technique.
15. To sort an array using insertion sort algorithm.
16. To implement quick sort algorithm.
17. To sort an array using bubble sort algorithm.

Program: B.Tech
Semester: Third
Course: Computer Organization & Architecture Lab
Course Code: 3PCCCS202P

L	T	P	C
0	0	4	2

List of Experiments:

1. To design the circuit of half adder.
2. To design the circuit of full adder.
3. To design the circuit of half subtractor.
4. To design the circuit of full subtractor.
5. To design an 8×1 Multiplexer.
6. To design a 4 bit combinational shifter.
7. To design a BCD adder.
8. To design a 4-bit adder subtractor.
9. To design 2:4 Decoder
10. Write the working of 8085 simulator GNUsim8085 and basic architecture of 8085along with small introduction.
11. Study the complete instruction set of 8085 and write the instructions in the instructionset of 8085 along with examples.
12. Write an assembly language code in GNUsim8085 to implement data transfer instruction.
13. Write an assembly language code in GNUsim8085 to store numbers in reverse orderin memory location.
14. Write an assembly language code in GNUsim8085 to implement arithmetic instruction.
15. Write an assembly language code in GNUsim8085 to find the factorial of a number.

Program: B.Tech
Semester: Fourth
Course: Digital Electronics Lab
Course Code: 3ESC202P

L	T	P	C
0	0	4	2

List of Experiment:

1. To illustrate & verify the working of AND, OR & NOT GATE.
2. To illustrate & verify the working of Exclusive OR & Exclusive NOR GATE
3. To illustrate & verify the working of NAND & NOR GATE
4. To Demonstrate the De-Morgan's Theorm.
5. To illustrate the working of Full adder & Half adder using various logic GATES.
6. To illustrate the working of Full subtractor & Half subtractor using various logic GATES.
7. To study IC 7404, IC 7432& IC 740 and verify the AND, OR & NOT GATE presence init.

Program: B.Tech
Semester: Fourth
Course: Object Oriented Programming with Java Lab
Course Code: 3PCCCS203P

L	T	P	C
0	0	4	2

List of Experiment:

1. Program to find square root of given number
2. Program to enter principal, rate & time and find simple interest
3. Program to find whether a year is leap year or not
4. Program to enter a number from keyboard and find out Fibonacci series
5. Program to enter a number from keyboard and find out factorial of the number
6. Program to enter a number from keyboard and check whether the number is palindrome ornot
7. Program to enter a number from keyboard and print the prime numbers present within it
8. Program to enter a number from keyboard and determine whether it is Armstrong or not.
9. Program to demonstrate switch statement
10. To swap two numbers without using third variable
11. To find the greatest among 3 numbers
12. Program to sort an array in an ascending order
13. Program to find out the sum and average of the elements present in an array
14. Program to add the elements of two different two dimensional array.
15. Program to find out the biggest and smallest number from a matrix.
16. Program to implement the concept of final class
17. Program to o implement the concept of interface
18. Program to reverse a specified string.
19. Write a program in java to show the user defined package.
20. Program to create an applet
21. Program to implement the concept of thread

Program: B.Tech
Semester: Fourth
Course: Design and Analysis of Algorithms Lab
Course Code: 3PCCCS205P

L	T	P	C
0	0	4	2

List of Experiment:

1. To implement Binary Search.
2. To implement Longest Common Subsequence (LCS).
3. To implement Matrix Chain Multiplication (MCM).
4. To implement Travelling Salesman Problem (TSP)
5. To implement MST using Kruskal's algorithm.
6. To implement MST using Prim's algorithm.
7. To implement DFS on a graph.
8. To implement BFS on a graph
9. To implement Dijkstra algorithm.
10. To implement 0/1 knapsack problem.
11. To implement Quick sort.
12. To implement Merge sort.
13. To implement Huffman Coding technique.
14. To implement All Pairs Shortest Path Problem(i.e Floyd-Warshall Algorithm)

Program: B.Tech
Semester: Fifth
Course: Database Management Systems Lab
Course Code: 3PCCCS301P

L	T	P	C
0	0	4	2

Course Objective:
The objectives of this course are:

- To Learn and practice data modeling using the entity-relationship and developing database
- Understand the use of Structured Query Language (SQL) and learn SQL syntax
- Understanding the basic principles of modeling of database using UML and apply normalization techniques to normalize the database system.
- Learn Multidimensional schemas suitable for data warehousing. And learn the Difference between OLTP (Online Transaction Processing) and OLAP (Online Analytical Processing).
- To demonstrate the principles behind the logical database design and Data Warehouse Modeling.

Course Outcome:
On completion of the course students will be able to:

- Describe the fundamental elements of relational database management systems.
- Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
- Design ER-models to represent simple database application scenarios.
- Convert the ER-model to relational tables, populate relational database and formulate SQL.
- Improve the database design by normalization.

List of Programs as Assignments:

Lab Assignment No: 1

Objective: Implementation of DDL commands of SQL with suitable examples

- Create table
- Alter table
- Drop Table

Lab Assignment No: 2

Objective: Implementation of DML commands of SQL with suitable examples

- Insert
- Update
- Delete

Lab Assignment No: 3

Objective: Implementation of different types of function with suitable examples

- Number function
- Aggregate Function
- Character Function
- Conversion Function
- Date Function

Lab Assignment No: 4

Objective: Study & Implementation of PL/SQL.

Lab Assignment No: 5

Objective Implementation of different types of operators in SQL

- Arithmetic Operators
- Logical Operators
- Comparison Operator
- Special Operator
- Set Operation

Lab Assignment No: 6

Objective: Implementation of different types of Joins

- Inner Join
- Outer Join
- Natural Join etc..

Lab Assignment No: 7

Objective: Study & Implementation of SQL Triggers.

Lab Assignment No: 8

Objective:

- Creating Database /Table Space
- Managing Users: Create User, Delete User
- Managing roles:-Grant, Revoke.

Lab Assignment No: 9

Objective: Study and Implementation of

- Group By & having clause
- Order by clause
- Indexing

Lab Assignment No: 10

Objective: Study & Implementation of

- Sub queries
- Views

Lab Assignment No: 11

Objective: Study & Implementation of different types of constraints.

Books recommended:

TEXT BOOKS

1. A.Silberschatz et.al - Database System Concepts, 5thEdn, Tata Mc-Graw Hill, New Delhi – 2000.

REFERENCE BOOKS

1. Date C.J. - An Introduction to Database System, Pearson Education, New Delhi, 2005.
2. R.Elmasri, Fundamentals of Database Systems, Pearson Education, New Delhi, 2005.

Program: B.Tech
Semester: Fifth
Course: IT Workshop (Sci Lab/MATLAB)
Course Code: 3PCCCS304P

L	T	P	C
0	0	2	1

Course Objective:
Students will be able to

- Understand and use the basic Matlab functions and understand its environment andvariables.
- Know about handling operations and advanced features like menus and toolbars.
- Implement programs with the use of arrays, strings and graphical data representations.
- Understand Python, Data Types, Operators, Arrays.
- Implement Functions and loops, object oriented programming using Python.

Course Outcome:

- After the successful completion of the course, the students will be able to:Apply features of Matlab and algorithms to solve problems
- Develop application programs with the help of various tool boxes available in Matlab.
- Apply data analysis through graphical data representations
- Implement programs with the use of arrays, strings in Matlab
- Implement Functions and loops, using Python

List of Experiments

1. Practicing SCILAB environment with simple exercises to familiarize Command Window, History, Workspace, Current Directory, Figure window, Edit window, Shortcuts, Help files.
2. Data types, Constants and Variables, Character constants, operators, Assignment statements using python
3. Control Structures: For loops, While, If control structures, Switch, Break, Continue statements using python
4. Input-Output functions, Reading and Storing Data using python
5. Vectors and Matrices, commands to operate on vectors and matrices, matrixManipulations.
6. Arithmetic operations on Matrices, Relational operations on Matrices, LogicalOperations on Matrices.
7. Polynomial Evaluation, Roots of Polynomial, Arithmetic operations on Polynomials.
8. Graphics: 2D plots, Printing labels, Grid & Axes box, Text in plot, Bar and Pie chart.

Text Book

1. Bansal R.K, Goel A.K., Sharma M.K., "MATLAB and its Applications in Engineering", Pearson Education, 2012.

References

1. Amos Gilat,"MATLAB-An Introduction with Applications", Wiley India, 2009.
2. Stephen.J.Chapman,"Programming in MATLAB for Engineers", Cengage Learning,201

Program: B.Tech
Semester: Fifth
Course: Operating Systems Lab
Course Code: 3PCCCS305P

L	T	P	C
0	0	4	2

Course Objective:

- To learn the mechanisms of OS to handle processes and threads and their communication
- To learn the mechanisms involved in memory management in contemporary OS To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
- To know the components and management aspects of concurrency management To learn to implement simple OS mechanisms

Course Outcome:
After the successful completion of the course, the students will be able to:

- Create processes and threads.
- Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.
- For a given specification of memory organization develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.
- Design and implement file management system.

List of Programs as Assignments:

1. Lab Assignment No: 1
Objective: To Understand and Implement Directory StructureQ1. WAP to create a File directory system.

2. Lab Assignment No: 2
Objective: To Understand and Implement Scheduling processesQ1. WAP to schedule various processes

3. Lab Assignment No: 3
Objective: To Understand and Implement FCFSQ1. WAP to implement FCFS CPU Scheduling

4. Lab Assignment No: 4
Objective: To Understand and Implement SJFQ1. WAP to implement SJF CPU scheduling.

5. Lab Assignment No: 5
Objective: To Understand and Implement SRTFQ1.WAP to implement SRTF CPU scheduling.

6. Lab Assignment No: 6
Objective: To Understand and Implement Scheduling algorithmsQ1. WAP to implement Round Robin Scheduling

7. Lab Assignment No: 7
Objective: To Understand and Implement Scheduling algorithmsQ1 WAP to implement SRTF scheduling .

8. Lab Assignment No: 8
Objective: To Understand and Implement context switching
Q1. WAP to implement Round Robin Scheduling with context switching.

9. Lab Assignment No: 9

Objective: To Understand and Implement context switching.Q1.WAP to implement SRTF with context switching.

10. Lab Assignment No: 10

Objective: To Understand and Implement Page Replacement TechniquesQ1. WAP to implement FCFS page replacement algorithm.

Q2. WAP to implement Optimal page replacement algorithm.

Program: B.Tech
Semester: Fifth
Course: Introduction to Python Programming Lab
Course Code: 3TECCS301P

L	T	P	C
0	0	2	1

Course Objective:
Students will be able to

- To learn and understand Python programming basics and paradigm.
- To learn and understand python looping, control statements and string manipulations.
- Students should be made familiar with the concepts of GUI controls and designingGUI applications.
- To learn and know the concepts of file handling, exception handling.

Course Content:

List of experiments:

1. Write a program to demonstrate different number datatypes in python.
2. Write a program to perform different arithmetic operations on numbers in python.
3. Write a program to create, concatenate and print a string and accessing substring from agiven string.
4. Write a python script to print the current date
5. Write a python program to create, append and remove lists in python.
6. Write a program to demonstrate working with tuples in python.
7. Write a program to demonstrate working with dictionaries in python.
8. Write a python program to find largest of three numbers
9. Write a python program to convert temperature to and from Celsius to Fahrenheit.
10. Write a python program to print prim numbers less than 20.
11. Write a python program to find factorial of a number using recursion.
12. Write a python program to define a module and import a specific function in that moduleto another program.
13. Write a Python class to implement pow (x, n).
14. Write a Python class to reverse a string word by word.

Program: B.Tech
Semester: Sixth
Course: Compiler Design Lab
Course Code: 3PCCCS306P

L	T	P	C
0	0	4	2

Course Objective:
The objectives of this course are:

- To understand the basic component of Natural Language Processing.
- To explore the application areas of Natural Language Processing.
- To understand the idea of Language Modeling.
- To explore the basic concepts of Parts-of-speech Tagging.
- To understand the concepts of language modeling.

Course Outcome:
On completion of the course students will be able to:

- Apply different compiler writing tools to implement the different Phases.
- Analyze the data flow and control flow.
- Construct the intermediate representation.
- Design and develop various modules of a compiler.
- Develop modules of compiler using Lex and Yacc tools.

List of Programs as Assignments:

1. Lab Assignment No: 1

Objective: To Understand the concept of tokens.

Q1. C program to count white spaces, numbers, words in a file./

2. Lab Assignment No: 2

Objective: To Understand the process of identification of tokens.

Q1. C program to design Finite automata to identify different tokens (identifiers, constants, Operators, etc.).

3. Lab Assignment No: 3

Objective: To have a brief Understanding to lex programming. Q1.

Count number of a‟s in given string.

Q2. Identify different patterns like aa, ab, not containing a, etc. in given string .

4. Lab Assignment No: 4

Objective: To Understand lex programming tool.

Q1. Lex program to Identify all tokens of C programs.

5. Lab Assignment No: 5

Objective: To Understand and Implement structure of any programming language. Q1.Design and Code individual programming code with all possible tokens in programming language.

6. Lab Assignment No: 6

Objective: To Understand lex programming tool in depth. Q1.

Starting and ending with „a‟.

Q2. # a‟s divisible by 2 or b‟s divisble by 3. Q3.

4th Symbol „a‟ from RHS.

Q4. Output code after removing white spaces and comment.

7. Lab Assignment No: 7

Objective: To Understand and Implement Parser using yacc. Q1.

Build parsers using yacc for L(G)={a

nb

n

| n >= 1} over {a,b}

8. Lab Assignment No: 8

Objective: To Understand and Implement parser for different grammars.

Q1.Build Parser using yacc for L(G) where rule set of G is { S ->aSb, S->bSa, S->c} over {a,b,c}.

9. Lab Assignment No: 9

Objective: To Understand and Implement parser coding.

Q1. Build parser using yacc to convert the infix expression to postfix expression.

10. Lab Assignment No: 10

Objective: To Understand and Implement parser coding.

Q1. Build a calculator in yacc which takes expression in postfix notation.

Q2. Build parsers using yacc to convert the prefix expression into the postfix expression.

11. Lab Assignment No: 11

Objective: To Understand and Implement parser for validation and operations. Q1. Build parsers using yacc to validate the C statements. E.g int a,b,c;(valid) Q2. Build calculator in yacc.

Books recommended:

Text books

lex&yacc (2nd ed.) :O'Reilly & Associates, Inc. Sebastopol, CA, USA ©1992 .

Reference books

Lex &Yacc:O'Reilly & Associates, Inc. Sebastopol, CA, USA ©1992.

Program: B.Tech
Semester: Sixth
Course: Computer Networks Lab
Course Code: 3PCCCS307P

L	T	P	C
0	0	4	2

Course Objective:
The objectives of this course are:

- To familiarize the student in introducing and exploring various Network topologies and networking protocols
- To understand the use of client/server architecture in application
- To enable the student on how to approach for networking problems using networking simulation tools.
- To Design reliable servers using both TCP and UDP sockets
- Familiar with network tools and network programming.

Course Outcome:
On completion of the course students will be able to:

- Express programming & simulation for networking problems.
- Understand of various aspects of networking devices
- Design and implement simulation of a simple LAN and a WAN that meet a specific set of criteria
- Identify the elements of a communication network
- Simulate various OSI layer protocols using C/C++/ Java

List of Programs as Assignments:

1. Lab Assignment No: 1

Q1. To familiarize with the Lab Network Topology, Locating different interfaces, routers and switches. Studying different pools of IP addresses.

Q2. Implement the data link layer framing methods such as character, character stuffing, and bit stuffing.

Q3. To learn and observe the usage of different networking commands e.g. PING, TRACEROUTE. Learning remote login using telnet session. Measuring typical average delays between different locations of the network.

2. Lab Assignment No: 2

Q1. What is the IP of the machine you are using? Compare it with the IP of your neighbors. Are the IPs of your neighbors same? Why or Why not?

Q2. Ping" is a tool used to determine if a server is responding and to estimate the round trip time of a message sent to that server. Use the ping command for the following URLs and record the success or failure statistics along with the average round trip time.

a) google.com

b) facebook.com

c) jru.edu.in

Q3. Trace the route that is taken when you try to access:

a) google.com

b) facebook.com

c) jru.edu.in

Q4. Network Commands on Linux / Unix

3. Lab Assignment No: 3

Q1. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC 32.

Q2. Implementation of Sub-netting and Super-netting.

Q3. To study different types of transmission media, various topologies, and configure modem of computer HUB and Switches.

4. Lab Assignment No: 4

Q1. Write a C/C++ program to determine if the IP address is in Class A, B, C, D, or E. Q2.

Write a C/C++ program to determine if the IP address is in Class A, B, or C.

Q3. Write a C/C++ program to translate dotted decimal IP address into 32 bit address.

Q4. To implement a routing protocol and check its connectivity in a variable length subnet masked network

Q5. Write a C/C++ program to perform bit stuffing and de-stuffing.

5. Lab Assignment No: 5

Q1. Implement Dijkstra,,s algorithm to compute the Shortest path through a graph. Q2. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm Q3. Take an example subnet of hosts. Obtain broadcast tree for it.

6. Lab Assignment No: 6

Q1. Build implementations of the Internet protocols

Q2. Implementation of Stop and Wait Protocol and Sliding Window Protocol. Q3. Write a code simulating ARP /RARP protocols.

7. Lab Assignment No: 7

Q1. Create a socket for HTTP for web page upload and download Q2.
Write a code simulating PING and TRACEROUTE commands.

Books recommended:

Suggested books:

1. William Stallings, Data and Computer Communication, Prentice Hall of India.
2. Behrouz A. Forouzan, Data Communication and Networking, McGraw-Hill.
3. Andrew S. Tanenbaum, Computer Networks, Prentice Hall.

Suggested reference books

1. W. Richard Stevens, TCP/IP Illustrated, Volume 1, Addison-Wesley.
2. Douglas Comer, Internetworking with TCP/IP, Volume 1, Prentice Hall of India.

Program: B.Tech
Semester: Sixth
Course: Fundamentals of Software Engineering Lab
Course Code: 3PCCCS308P

L	T	P	C
0	0	2	1

Course Objective:
Students will be able to

- Familiarize the students with the fundamental concepts of Software Engineering
- Impart state-of-the-art knowledge on SRSand UML
- Explore case studies to demonstrate practical applications of different concepts
- Provide a platform where they can solve real life problems

Course Outcome:
After the successful completion of the course, the students will be able to:

- Prepare efficient models for development of software for various projects
- Collect the requirements the client wants for the software being produced
- Design the UML diagrams necessary for the software being developed
- Create and specify feasible software designs based on the requirements/specifications
- Assess the extent and costs of a project with the help of several different assessment methods

JHARKHAND Rai University RANCHI

SYLLABUS

List of Programs as Assignments:

1. ***Lab Assignment No: 1***

Objective: To Understand and Implement Identification of Requirements from ProblemStatements

Q1. To consider the problem statement for a project to be developed and list out the ambiguities, inconsistencies and incompleteness of the problem statement.

Q2. To identify different functionaries to be obtained from a system and characteristics that asystem should have, but not possessed by the system itself

2. ***Lab Assignment No: 2***

Objective: To Understand and Implement Estimation ofProject Metrics

Q1. To estimate the minimum size of the team one would require to develop a project throughapplication of intermediate COCOMO.

Q2. To use Halstead's metrics to estimate the effort required to recreate a program in JAVAfrom C.

3. ***Lab Assignment No: 3***

Objective: To Understand and Implement Modeling UML Use Case Diagrams and Capturing UseCase Scenarios

Q1. To draw a use case diagram for the given case study.

Q2. To identify the primary and secondary actors for the system and generalization of use casesand «include» stereotypes to prevent redundancy in the coding phase.

4. ***Lab Assignment No: 4***

Objective: To Understand and Implement E-R Modeling from the Problem Statements

Q1. To identify the possible entity sets, their attributes, and relationships for the given casestudy.

Q2. To draw an ER diagram for the given case study.

5. ***Lab Assignment No: 5***

Objective: To Understand and Implement Identification of Domain Classes from the ProblemStatements

Q1. To identify potential classes and their attributes for the given case study.

Q2. To utilize expert knowledge on the subject matter to identify other relevant classes.

6. ***Lab Assignment No: 6***

Objective: To Understand and Implement Identification of Components from the ProblemStatements

Q1. To identify potential components for the given case study.Q2. To draw component diagram for the given case study

7. Lab Assignment No: 7

Objective: To Understand and Implement State Chart and Activity Modeling

Q1. To draw a statechart diagram to graphically represent the given case study.

Q2. To draw an activity diagram to graphically represent the workflow of the given case study.

8. Lab Assignment No: 8

Objective: To Understand and Implement Modeling UML Class Diagrams and Sequence diagramsQ1. To draw class diagram for the given case study.

Q2. To draw sequence diagram for the given case study.

9. Lab Assignment No: 9

Objective: To Understand and Implement Modeling Data Flow Diagrams

Q1. To draw data flow diagram (Level 0, 1 and 2) for the given case study.

10. Lab Assignment No: 10

Objective: To Understand and Implement Estimation of Test Coverage Metrics and Structural Complexity

Q1. To identify the basic blocks for a given programQ2. To draw a CFG using the basic blocks

Q3. To determine McCabe's complexity from a CFG.

11. Lab Assignment No: 11

Objective: To Understand and Implement Designing Test SuitesQ1. To design a test suite for the given case study.

Q2. To verify implementation of functional requirements by writing test cases.Q3. To analyze results of testing to ascertain the current state of the project.

12. Lab Assignment No: 12

Objective: To Understand and Implement Forward and Reverse EngineeringQ1. To obtain programs from UML diagrams.

Q2. To obtain UML diagrams from programs.

Program: B.Tech
Semester: Sixth
Course: Distributed Database Management System Lab
Course Code: 3TECCS311P

L	T	P	C
0	0	2	1

Course Objectives:

Students will be able to

- To design and implement a database schema for a given problem domain
- To create and manipulate tables using SQL queries
- To prepare a Database for a given problem
- To develop applications using PL/SQL

Course Outcomes:

After the successful completion of the course, the students will be able to:

- Understand, analyze and apply common SQL statements including DDL, DML and DCL statements to perform different operations.
- Design different views of tables for different users and to apply embedded and nested queries.
- Design and implement a database for a given problem according to well-known design principles that balance data retrieval performance with data consistency.
- Apply normalization techniques to avoid redundancy

List of Experiments:

1. A) Introduction of Database management systems, Oracle concepts and Create a table.
B) How to insert data in a table using insert and display the records in a table.

2. A) Update or Delete records of a table and modifying structure of a table using Alter and Drop command.
B) Study of character functions for manipulation of data items.

3. To perform join operation between various tables.
4. Applying constraint using two tables.
5. How to retrieve data from different tables using sub queries and correlated queries.
6. Create two databases either on single DBMS and Design Database to fragment and share the fragments from both database and write single query for creating view.
7. Understanding of Database Objects: synonym, sequence, index and view.
8. To study the concepts of Normalization.
9. Case study on noSQL.
10. Case study on hadoop.

Program: B.Tech
Semester: Sixth
Course: Web Technology Lab
Course Code: 3TECCS313P

L	T	P	C
0	0	2	1

Course Objective:
The objectives of this course are:

- To make student familiar with client server architecture
- To make student able to develop a web application using java technologies.
- To gain the skills and project-based experience needed for entry into web application and development careers.

Course Outcome:
On completion of the course students will be able to:

- Students are able to develop a dynamic webpage by the use of java script and DHTML.
- Students will be able to write a well formed / valid XML document.
- Students will be able to connect a java program to a DBMS and perform insert, update and delete operations on DBMS table.
- Students will be able to write a server side java application called Servlet to catch form data sent from client, process it and store it on database.
- Students will be able to write a server side java application called JSP to catch form data sent from client and store it on database.

Lab Exercises

1. Write a HTML program for the demonstration of Lists.
    a. Unordered List
    b. Ordered List
    c. Definition List
    d. Nested List
2. Write a HTML program for demonstrating Hyperlinks.
    a. Navigation from one page to another.
    b. Navigation within the page.
3. Write a HTML program for time-table using tables.
4. Write a HTML program to develop a static Home Page using frames.
5. Write a HTML program to develop a static Registration Form.
6. Write a HTML program to develop a static Login Page.
7. Write a HTML program to develop a static Web Page for Catalog.
8. Write a HTML program to develop a static Web Page for Shopping Cart.
9. Write HTML for demonstration of cascading stylesheets.
    a. Embedded stylesheets.
    b. External stylesheets.
    c. Inline styles.
10. Write a javascript program to validate USER LOGIN page.
11. Write a javascript program for validating REGISTRATION FORM
12. Write a program for implementing XML document for CUSTOMER DETAILS.
13. Write an internal Document Type Definition to validate XML for CUSTOMER DETAILS?
14. Write an external Document Type Definition to validate XML for CUSTOMER DETAILS?
15. Write an XML for person information and access the data using XSL.
16. Write an XML for student information and access second students data using DOM.
17. Write a program to display contents of XML file in a table using Extensible Style Sheets.

WEB TECHNOLOGIES LAB MANUAL

18. Write a simple servlet that displays a message.
19. Write a servlet that reads parameters from employee login page.
20. Write a servlet for creating a cookie and retrieving it.
21. Write a servlet for session tracking.
22. Write a JSP that reads parameters from user login page.
23. Write a JSP that reads a value, creates a cookie and retrieves it.
24. Write a JSP for session tracking.

25. Write a servlet that connects to the database and retrieves the data and displays it.

Program: B.Tech

Semester: Seven

Course: Artificial Intelligence Lab

Course code: 3TECCS402P

L	T	P	C
0	0	2	1

Course Objectives:

The objectives of this course are:

- To impart knowledge about the practical aspects in Artificial Intelligence related problems
- To program different AI methods using a programming language
- To know how the logical operations and reason based AI problems are used using programming

Course Outcomes:

Upon successful completion of this course the students would be able:

CO1: Solve various kinds of problems using AI techniques.
CO2: Solve basic AI based problems using any programming language.
CO3: Understand to implement the various kinds of AI based algorithms.
CO4: Apply AI techniques to real-world problems to develop intelligent systems.
CO5: To understand problems related to AI

Artificial Intelligence Lab:

1. Write a program to implement the Hill Climbing problem
2. Write a program to implement the Towers of Hanoi problem
3. Write a program to implement the Missionaries and Cannibals problem
4. Write a program to implement the 8 queen's problem
5. Write a program to implement the A* Algorithm
6. Write a program to implement the Breadth first algorithm
7. Write a program to implement the Depth first algorithm
8. Write a program to implement the predicate logic
9. To study various datasets used to train the AI models.
10. To implement linear classification.
11. To implement non-linear classification.
12. To implement feed-forward neural network.
13. To implement back-propagation neural network.
14. To study and implement CNN.
15. To study and implement RNN and LSTM based sequence learning models.
16. To study and implement transformers.

Program: B. Tech
Semester: Seven
Course: AI for Games Lab
Course code: 3TECCS406P

L	T	P	C
0	0	2	1

This lab will introduce students to the basic principles and techniques of artificial intelligence (AI) in the context of game development. Students will learn about different AI strategies and how to implement them to create intelligent behaviors in games.

Learning Objectives

- Understand the basics of AI in game development.
- Implement path finding algorithms using A*.
- Create NPC behaviors using finite state machines (FSM).
- Develop decision-making processes using behavior trees.

Prerequisites

- Basic knowledge of programming (preferably in C# or Python).
- Familiarity with a game development platform (Unity or Unreal Engine).

Detailed of Lab Practices

1. Write a program to implement Path finding with A* Algorithm
2. Write a program to implement Finite State Machines (FSM) for NPC Behavior
3. Write a program to implement Decision-Making with Behavior Trees
4. Write a program to implement First Person Shooter Game
5. Write a program to implement Real Time Strategy (RTS) Game
6. Write a program to implement Role Playing Games (RPGs)
7. Write a program to implement Plat former Game
8. Write a program to implement Tic Tac Toe Game
9. Write a program to implement God Games
10. Write a program to implement Sports Games
11. Write a program to implement Hidden Door Games
12. Write a program of The Case of 'Angry Pumpkins' Games
13. Write a program to implement **Storytelling** Games
14. Write a program to implement Decision-Making with Behavior Trees