ACHARYA NAGARJUNA UNIVERSITY

SCHEME OF INSTRUCTION AND EXAMINATION, w.e.f. 2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.Tech. ARTIFICIAL INTELLIGENCE & MACHINE LEARNING (AM)

II/IV B. Tech SEMESTER-I

S.No.	Course Details		Category	Scheme of Instruction			Scheme of Examination		
	Code	Subject Name		Hours in a Week			Marks		Credits
				L	T	P	Internal	External	
1	22CS/AM/ CYS/DS 211	Probability & Statistics	BSC	3	0	0	30	70	3
2	22AM/DS 212	Data Structures & Algorithms	PC	3	0	0	30	70	3
3	22AM/DS 213	UNIX Programming	PC	3	0	0	30	70	3
4	22AM/CS 214	Introduction to Artificial Intelligence	PC	3	0	0	30	70	3
5	22CS/AM/ CYS/DS 215	Professional Ethics and Human Values	MC	3	0	0	30	70	0
6	22 AM/ DS 251	Data Structures & Algorithms Lab	PC	0	0	3	30	70	1.5
7	22AM/DS 252	UNIX Lab	PC	2	0	0	30	70	1.5
8	22CS/AM 253	Introduction to AI Lab	PC	0	0	3	30	70	1.5
9	22CS/AM/ CYS/DS 254	Statistics Using R LAB	Skill	0	0	3	30	70	2
Total Credits									18.5

S. No	Category	Abbreviation	Required Credits Criteria
1	**BSC**	**Basic Science Course**	**3**
2	**PC**	**Program Core Course**	**13.5**
3	**SC**	**Skill Oriented Course**	**2**
4	**MC**	**Mandatory Course**	**0**

ACHARYA NAGARJUNA UNIVERSITY

SCHEME OF INSTRUCTION AND EXAMINATION, w.e.f. 2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.Tech. ARTIFICIAL INTELLIGENCE & MACHINE LEARNING (AM)

II/IV B. Tech SEMESTER-II

S.No.	Course Details		Category	Scheme of Instruction			Scheme of Examination		
	Code	Subject Name		Hours in a Week			Marks		Credits
				L	T	P	Internal	External	
1	22CS/AM/ CYS/DS 221	Discrete Mathematics	BSC	3	0	0	30	70	3
2	22CS/AM/ CYS/DS 222	Essence of Indian Traditional Knowledge	HSC	3	0	0	30	70	3
3	22AM/DS 223	Advanced Data Structures	PC	3	0	0	30	70	3
4	22AM/DS 224	Java Programming	PC	3	0	0	30	70	3
5	22AM/DS 225	Operating System	PC	3	0	0	30	70	3
6	22AM 226	Artificial Intelligence	PC	3	0	0	30	70	3
7	22AM/DS 261	Operating System LAB	PC	0	0	2	30	70	1.5
8	22AM/DS 262	Java Lab	PC	0	0	2	30	70	1.5
9	22AM 263	Artificial Intelligence Lab	PC	0	0	2	30	70	1.5
10	22CS/AM/ CYS/DS 264	Communicative English Lab II	SKILL	0	0	3	30	70	2
Total Credits									**24.5**
Internship 2 Months (Mandatory) during Summer vacation									
Honors/Minor Courses (The hours distribution can be 3-0-2 or 3-1-0 also)									**4**

S. No	Category	Abbreviation	Required Credits Criteria
1	BSC	Basic Science Course	3
2	PC	Program Core Course	16.5
3	SC	Skill Oriented Course	2
4	HSC	Humanities and Social Sciences	3

II/IV B. Tech SEMESTER-I

22CS/AM/CYS/DS 211	**Probability & Statistics**	**L**	**T**	**P**	**M**	**C**
		3	**0**	**0**	**100**	**3**

COURSE OBJECTIVES

To provide the students with sufficient knowledge in probability and statistics, this can be used in their respective fields.

COURSE OUTCOMES

By the end of the semester, the student will be able to:

1. Remembering the theory and have practical knowledge of Statistics, Measures of Central tendency, Variability, Skewness, Kurtosis, correlation, rank correlation, regression coefficients, principle of least squares.
2. Understanding the clear conception of the terms Probability, random variables (discrete and continuous), probability density functions, mathematical expectation.
3. Analysing evaluation of Probability distribution - Binomial, Poisson approximation to the binomial distribution and normal distribution-their properties.
4. Evaluation about the Estimation- sampling distribution, point estimation, Formulation of null hypothesis, Large Sample: Tests of significance and Confidence interval.
5. Evaluation of Student t-distribution, F-test, $\chi 2$ - test for goodness of fit, about test for independence of attributes.

Syllabus:

Unit 1: Descriptive statistics
Statistics Introduction, Population vs Sample, Measures of Central tendency, Measures of Variability (spread or variance) Skewness Kurtosis, correlation, correlation coefficient, rank correlation, regression coefficients, principle of least squares, method of least squares, regression lines.

UNIT 2: Probability
Probability, probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, random variables (discrete and continuous), probability density functions, properties, mathematical expectation, addition and multiplication theorems on mathematical expectation.

UNIT 3: Probability distributions
Probability distribution - Binomial, Poisson approximation to the binomial distribution and normal distribution-their properties. Central limit theorem applications.

Unit 4: Estimation and Testing of hypothesis, large sample tests
Estimation-parameters, statistics, sampling distribution, point estimation, Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test. Large Sample Tests: Test for single proportion, difference of proportions, test for single mean and difference of means.

Unit 5: Small sample tests

Student t-distribution (test for single mean, two means and paired t-test), testing of equality of variances (F-test), $\chi 2$ - test for goodness of fit, χ^2 - test for independence of attributes. Confidence interval for parameters in one sample and two sample problems, difference between testing of hypothesis and interval estimation.

Reference Books:

1. Miller and Freunds, Probability and Statistics for Engineers,7/e, Pearson, 2008.
2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.
3. S. Ross, a First Course in Probability, Pearson Education India, 2002.
4. W. Feller, an Introduction to Probability Theory and its Applications, 1/e, Wiley, 1968.

Mapping of course outcomes with program outcomes:

CO's No.	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**	-	**3**	**2**	-	-	-	-	-	-	**1**	-	-
CO2	**3**	**2**	-	**3**	**2**	-	-	-	-	-	-	**1**	-	-
CO3	**3**	**2**	-	**3**	**2**	-	-	-	-	-	-	**1**	-	-
CO4	**3**	**2**	-	**3**	**2**	-	-	-	-	-	-	**1**	-	-
CO5	**3**	**1**	-	**3**	**2**	-	-	-	-	-	-	**1**	-	-

22CS/AM/DS 212	Data Structures & Algorithms	L	T	P	M	C
		3	0	0	100	3

Objectives of the course:

1. To impart the basic concepts of data structures and algorithms.
2. To understand concepts about searching and sorting techniques
3. To understand basic concepts about stacks, queues, lists, trees and graphs.
4. To enable them to write algorithms for solving problems with the help of fundamental data structures.

Course Outcomes :

At the end of this course, the students will be able to

1. Remembering basic data structure operations and analyse the time and space complexity of algorithms.
2. Understanding algorithms using the basic operations of stacks and queues and analyse their complexity.
3. Understanding basic operations of linked lists and analyse their algorithm complexity.
4. Evaluate the performance of selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort algorithms in term of Space and Time complexity and implementing the searching algorithms.
5. Constructing binary trees, binary search tress, AVL trees and B+ trees and Graphs.

UNIT I:

Introduction: Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space trade off.

Array Data Structure: Array ADT and its operations, Time complexity.

UNIT II:

Stacks and Queues: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation – corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each types of Queues: Algorithms and their analysis.

UNIT III:

Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.

UNIT IV:

Sorting: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods.

Searching: Linear Search and Binary Search Techniques and their complexity analysis.

Hashing: Hash function, Open addressing and separate chaining.

UNIT V:

Trees: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis, Tree traversals.

Graph: Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

Reference books:

1. Data Structures and Algorithms by Alfred V Aho,John E Hopcroft, Jeffrey D Uiiman, Pearson Education.
2. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.
3. Classic Data Structures by Debasis Samanta, PHI Publications.
3. Data Structures LIPSCHUTZ , Schaum publications.

MAPPING OF CO's &PO's:

Mapping of Course Outcomes with Program Outcomes														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	3	2	-	1	-	-	-	-	-	-	-	2	1
CO2	3	2	1	-	1	-	-	-	-	-	-	-	2	1
CO3	3	2	1	-	1	-	-	-	-	-	-	-	1	2
CO4	3	2	1	-	1	-	-	-	-	-	-	-	2	1
CO5	2	2	2	2	1	-	-	-	-	-	-	-	2	1

22AM/DS 213	Unix Programming	L	T	P	M	C
		3	0	0	100	3

Course Objectives:

1. Written technical communication and effective use of concepts and terminology.
2. Facility with UNIX command syntax and semantics.
3. Ability to read and understand specifications, scripts and programs.
4. Individual capability in problem solving using the tools presented within the class.

Course Outcomes:

1. Understand the basic Unix architecture, commands and utilities of the UNIX operating system and to work confidently in Unix/Linux environment and open systems
2. Creating simple and complex shell scripts to automate various tasks using shell programming.
3. Analyse file management system calls.
4. Understand various concepts of process and process related commands in UNIX.
5. Understand UNIX system administration and Inter Process Communication.

UNIT I

Introduction to unix: Unix architecture , Features of Unix, Vi editor.
Directory Related utilities- pwd, mkdir, ls, cd ,rmdir.
File Handling and Text Processing utilities-
cp, mv, rm, ln, unlink, lp, cat, more, pg, head, tail, sort ,nl, grep, egrep, fgrep, cut, paste, join, tee, w ,chgrp, chmod, chown,find, cmp, diff, uniq, tr.
Disk utilities, Backup and other utilities-
du, df, mount, unmount, umask, ulimit,tar, cpio, dump , who, mail, compress, uncompress, gzip, gunzip, crypt, sed, tty,
Networking utilities – finger, telnet, rlogin, ftp, write, talk, wall.

UNIT-II

Bourne Shell programming:
Shell, functions of the shell , Meta characters, Input redirection, Output redirection, pipes, shell as programming language, shell variables, predefined local variables, predefined environment variables, Arithmetic, conditional expressions, control structures, positional parameters, passing command line arguments, Built – in Shell commands and shell programs.

UNIT-III

File management system calls:
Regular file management system calls – open( ),read( ), write( ) , lseek( ), Close(),unlink( ), stat( ), getdents( ). Miscellaneous file management system calls – chown( ) and fchown( ), chmod( ) ,fchmod( ), dup( ) ,dup2( ), fcntl( ), ioctl( ), link( ), mknod( ), sync( ),truncate( ) and ftruncate( ).

UNIT IV:

Process Management:
Creating a new process – fork( ),orphan processes, terminating a process – exit( ), zombie processes, waiting for child – wait( ),Differentiating a process – exec( ), changing directories – chdir( ), changing priorities- nice( ).

UNIT V

Signals: The defined signals, A list of signals, terminal signals, Requesting on Alarm signal – alarm( ), handling signals – signal( ), protecting critical code and chaining interrupt handlers, sending signals – kill( ), Death of children, suspending and Resuming processes, process Group's and control terminals.
Inter process communication: Pipes, Sockets, shared memory, semaphores.

Reference Book:

1. "**Unix for programmers and users**" 3rd edition by Graham Glass, King Ables, Pearson Education.
2. "**Advanced programming in the unix environment**" w- Richard Stevens 2nd Edition Pearson education
3. "**Unix programming environment"**, Kernighan and pike, Pearson education.
4. "**Your unix the ultimate guide**" Sumitabha Das, TMH 2nd edition.
5. "**Advanced unix programming**" by Marc J. Rochkind, 2nd edition Pearso

MAPPING OF CO's &PO's:

Mapping of Course Outcomes with Program Outcomes													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	1	-	1	-	-	-	-	-	-	-	-	1	2
CO2	2	2	1	2	-	-	-	-	-	-	-	-	1	1
CO3	3	1	-	1	-	-	-	-	-	-	-	-	1	1
CO4	2	1	1	1	-	-	-	-	-	-	-	-	1	2
CO5	2	1	1	1	-	-	-	-	-	-	-	-	1	1

22CS/AM 214	Introduction to Artificial Intelligence	L	T	P	M	C
		3	0	0	100	3

COURSE OBJECTIVE:

1. The main purpose of this course is to provide the most fundamental knowledge to the students so that they can understand what the AI is.
2. Due to limited time, we will try to eliminate theoretic proofs and formal notations as far as possible, so that the students can get the full picture of AI easily.
3. Students who become interested in AI may go on to the graduate school for further study

COURSE OUTCOMES:

After successful completion of this course, the students will be able to:
CO 1: Outline the fundamentals of artificial intelligence and characteristics of problems.
CO 2: Analyze different search techniques and predicate logic in artificial Intelligence.
CO 3: Interpret knowledge representation and symbolic reasoning using different rules.

UNIT - I

Introduction: Definition of AI, What is AI Technique? Foundations of AI, Applications of AI. Defining the problem as a State space search, production systems, Problem characteristics, Production System characteristics

Intelligent agents: Agents and Environments, Structure of agents.

UNIT - II

Problem Solving Techniques: Solving Problems by Searching: Problem Solving Agents, Searching for Solutions.

Heuristic Search Techniques- Generate & Test, Hill Climbing, Best First search, Problem reduction, Constraint satisfaction, Means Ends Analysis.

UNIT – III

Predicate Logic– Representing Simple Facts in Logic, Representing Instances and Isa relationships, Computable functions and Predicates, Resolution, Natural deduction.

UNIT – IV

Representing Knowledge Using Rules- Procedural vs Declarative knowledge, Logic Programming, Forward vs Backward Reasoning, Matching and Control Knowledge.

Logical Agents: Knowledge Based Agents, Logic, Propositional logic, First order logic, Syntax and Semantics in First order Logic

UNIT-V

Inference in first order logic: propositional vs. First order inference, Unification and Lifting, Forward chaining, Backward chaining, Resolution

Symbolic reasoning under uncertainty– Introduction to non-monotonic Reasoning, Logics for non-monotonic reasoning, Implementation Issues, Implementation: Depth First Search, Breadth First Search

Text Books

1. Artificial Intelligence by Elaine Rich & Kevin Knight, Second edition, Tata Mcgraw hill.
2. Introduction to Artificial Intelligence and Expert Systems by Dan W. Paterson, PHI
3. Stuart Russell and Peter Norvig, ―Artificial Intelligence: A Modern Approach‖ , 3rd Edition, Prentice Hall.

REFERENCE BOOKS:

1. A Classical Approach to Artificial Intelligence, M.C. Trivedi, Khanna Book Publishing, 2019.
2. Artificial Intelligence – A modern Approach by Stuart Russell and Peter Norvig, Second edition, Pearson.
3. Artificial Intelligence by Saroj Kaushik, Cengage Learning.

Mapping of CO's & PO's:

COS	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3				3		1			1	1	1	2	1
CO2		3	1			3	1			1	1	1	2	1
CO3		3	2		3		2			1	1	1	2	1

22CS/AM/CYS/DS 215 Professional Ethics and Human Values

L	T	P	M	C
3	0	0	100	0

Course Objectives:

- Understand the concept of ethics and its importance in professional life.
- Identify and analyze ethical issues and dilemmas in various professional domains.
- Develop ethical decision-making skills and strategies.
- Understand the role of values in shaping professional behavior.
- Foster ethical leadership and responsibility in professional settings.
- Reflect on personal ethical development and growth.

Course Outcomes:

1. Demonstrate knowledge of ethical theories and principles.
2. Analyze and evaluate ethical dilemmas in real-world scenarios.
3. Apply ethical decision-making models to resolve complex problems.
4. Examine the influence of personal and cultural values on professional conduct.
5. Demonstrate leadership skills with integrity and ethical behavior.

UNIT – I

Human Values: Morals, Values and Ethics – Integrity – Work Ethic – Service Learning – Civic Virtue – Respect for Others – Living Peacefully – caring – Sharing – Honesty – Courage – Valuing Time – Co-operation – Commitment – Empathy – Self-Confidence – Character – Spirituality.

UNIT – II

Engineering Ethics: Senses of 'Engineering Ethics' – Variety of moral issues – Types of inquiry – Moral dilemmas – Moral Autonomy – Kohlberg's theory – Gilligan's theory – Consensus and Controversy – Professions and Professionalism – Professional Ideals and Virtues – Theories about right action – Self-interest – Customs and Religion – Uses of Ethical Theories.

UNIT – III

Engineering as Social Experimentation: Engineering as Experimentation – Engineers as responsible Experimenters – Codes of Ethics – A Balanced Outlook on Law.

UNIT - IV

Safety, Responsibility and Rights: Safety and Risk – Assessment of Safety and Risk – Risk Benefit Analysis and reducing risk.

Collegiality and Loyalty – Respect for Authority – Collective Bargaining – Confidentiality – Conflicts of Interest – Occupational Crime – Professional Rights – Employee Rights – Intellectual Property Rights (IPR) - Discrimination

UNIT – V

Global Issues: Multinational Corporations – Environmental Ethics – Computer Ethics – Weapons Development – Engineers as Managers – Consulting Engineers – Engineers as Expert Witnesses and Advisors – Moral Leadership – Sample Code of Ethics like ASME, ASCE, IEEE, Institution of Engineers (India), Indian Institute of Materials Management, Institution of electronics and telecommunication engineers (ISTE), India, etc.

Reference Books:

1. R.S. Naagarazan “A Textbook on Professional ethics and Human Values”, New Age International Publihers, 2006.
2. Govindarajan. M, Natarajan. S, Senthilkumar. V.S, “Engineering Ethics”, Prentice Hall of India, 2004.
3. Charles D Fleddermann, "Engineering Ethics", Prentice Hall, New Jersey, 2004 (Indian Reprint).
4. Charles E Harris, Michael S Pritchard and Michael J Rabins, "Engineering Ethics Concepts and Cases", Thompson Learning, United States, 2000 (Indian Reprint now available).
5. John R Boatright, “Ethics and the Conduct of Business”, Pearson Education, New Delhi, 2003.
6. Edmund G Seebauer and Robert L Barry, “Fundamentals of Ethics for Scientists and Engineers”, Oxford University Press, Oxford, 2001.
7.

Mapping of CO’s & PO’s:

COS	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2					1			1	1	1	2	1
CO2		3	2				1			1	1	1	2	1
CO3	3		2				1			1	1	1	2	1
CO4				3			1			1	1	1	2	1
CO5					3		1			1	1	1	2	1

22CS/AM/DS 251	Data Structures & Algorithms Lab	**L**	**T**	**P**	**M**	**C**
		0	**0**	**3**	**100**	**1.5**

COURSE OBJECTIVES:

1. To impart the basic concepts of data structures and algorithms
2. To understand concepts about searching and sorting techniques
3. To Understand basic concepts about stacks, queues, lists, trees and graphs
4. To understanding about writing algorithms and step by step approach in solving problems
5. With the help of fundamental data structures

COURSE OUTCOMES:

1. Understand basic data structures such as arrays, linked lists, stacks and queues.
2. Implement the stack, Queue and their applications
3. Implement various types of linked lists and their applications
4. Ability to have knowledge of tree and graphs concepts.
5. Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data

1. Write a C program to perform insertion, deletion, display operations on Array Data Structure.

2. Write a C program to perform insertion, deletion, display operations on stack Data Structure using arrays.

3. Write a C program to convert given infix expression into postfix expression and evaluate the postfix expression.

4. Write a C program to convert given infix expression into prefix expression and evaluate the prefix expression.

5. Write a C program to perform insertion, deletion, and display operations on queue Data Structure using arrays.

6. Write a C program to perform insertion, deletion, display operations on circular Queue Data Structure using arrays.

7. Write a C program to perform insertion, deletion, display operations on single linked list Data Structure.

8. Write a C program to perform insertion, deletion, display operations on circular single linked list Data Structure.

9. Write a C program to perform insertion, deletion, display operations on double linked list Data Structure.

10. Write a C program to perform insertion, deletion, display operations on circular double linked list Data Structure.

11. Implement a polynomial ADT and write a program to read two polynomials and
print them, adds the polynomials, prints the sum, multiply the polynomials and
print the product.

12. Implement the following sorting operations:-
    (a) Shell Sort, (b) Heap Sort (c) Merge Sort (d) Quick Sort e) Radix Sort

13. Implement the following sorting operations:-
    (a) Sequential search (b) Binary Search

14. Write a C program for open addressing.

15. Implement Binary Tree ADT and write a program that reads postfix Arithmetic
expression form, builds the expression tree and performs tree Traversal on it.

16. Implement Binary search Tree ADT and write a program that interactively allows
    (a) Insertion (b) Deletion (c) Find_min (d) Find max (e) Find operations

17. Write a C program for Graph Traversals: a) DFS b) BFS

MAPPING:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**	**2**	**1**	**2**								**1**	**2**
CO2	**3**	**2**	**2**	**1**	**2**								**2**	**1**
CO3	**3**	**2**	**2**	**1**	**2**								**1**	**1**
C04	**3**	**2**	**2**	**1**	**2**								**1**	**1**
CO5	**3**	**2**	**2**	**1**	**2**								**1**	**1**

22AM/DS 252	UNIX Lab	L	T	P	M	C
		0	0	3	100	1.5

COURSE OBJECTIVES:

This course introduces basic understanding of UNIX OS, UNIX commands and File system and to familiarize students with the Linux environment. To make student learn fundamentals of shell scripting and shell programming. Emphases are on making student familiar with UNIX environment and issues related to it

COURSE OUTCOMES:

Upon completion of this course, the student will be able to:

1. Understand the basic concepts of UNIX Architecture and basic Commands.
2. Understand different types of Files, File system and basic file system commands.
3. Understand the commands related to Shell basics, vi editor and regular expression commands.
4. Implement vi editor commands and shell programs
5. Understand the concepts of advance file concepts, commands related to Shell script and filter commands.

Programs:

1. Working with different Unix commands, Pipes, I/O redirection.
2. Write Shell Programs for the following
   a. Display all the words which are entered as command line arguments.
   b. Changes Permissions of files in PWD as rwx for users.
   c. To print the list of all sub directories in the current directory.
   d. Program which receives any year from the keyboard and determine whether the year is leap year or not. If no argument is supplied the current year should be assumed.
   e. Program which takes two file names as arguments, if their contents are same then delete the second file.
3. Write shell scripts for the following
   a. To print the given number in the reversed order.
   b. To print first 25 Fibonacci numbers.
   c. To print the Prime numbers between the specified range.
   d. To print the first 50 Prime numbers.
4. Write shell scripts for the following
   a. To delete all lines containing the word ‘Unix’ in the files supplied as arguments.
   b. Menu driven program which has the following options.
      i. contents of /etc/passed
      ii. list of users who have currently logged in.
      iii. present working directory.
      iv. exit.
   c. For sorting, searching and insertion, deletion of elements in the list
5. Program to transfer the data from one file to another file by using un-buffered I/O.
6. Program to create two processes to run a loop in which one process adds all even numbers and the other adds all the odd numbers (Hint: use fork ( )).
7. Program to create to process ‘i’ and sends data to process ‘j’, prints the same after receiving it. (Hint: use fork( )).
8. Program to demonstrates orphan process.
9. Program which demonstrates how to avoid Zombie using wait( ) .

MAPPING OF CO's &PO's:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	**3**	**2**	**2**	**1**	**2**								**1**	**2**
CO2	**3**	**2**	**2**	**1**	**2**								**2**	**1**
CO3	**3**	**2**	**2**	**1**	**2**								**1**	**1**
C04	**3**	**2**	**2**	**1**	**2**								**1**	**1**
CO5	**3**	**2**	**2**	**1**	**2**								**1**	**1**

22CS/ AM 253	Introduction to AI Lab	L	T	P	M	C
		0	0	3	100	1.5

1. (a). Write a python program to print the multiplication table for the given number?
   (b). Write a python program to check whether the given number is prime or not?
   (c) Write a python program to find factorial of the given number?
2. Write a python program to implement simple Chatbot?
3. (a) Write a python program to implement List operations (Nested List, Length, Concatenation, Membership, Iteration, Indexing and Slicing)?
   (b) Write a python program to implement List methods (Add, Append, Extend & Delete).
4. (a). Write a python program to Illustrate Different Set Operations?
   (b). Write a python program to generate Calendar for the given month and year?
   (c). Write a python program to implement Simple Calculator program?
5. (a). Write a python program to Add Two Matrices.
   (b). Write a python program to Transpose a Matrix.
6. Write a python program to implement Breadth First Search Traversal?
7. Write a python program to implement Water Jug Problem?

22CS/AM/CYS/DS 254	STATISTICS USING R Lab	L	T	P	M	C
		0	0	3	100	2

List of practical's using R-programming

1. Correlation Coefficient
2. Regression Coefficient
3.Curve fitting
4.Graphs of pmf, pdfs and cdfs of discrete and continuous distributions
5.Estimation
6. Z test
7. t test
8. F test
9. χ^2 test
10.Interval Estimation

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

B.Tech. ARTIFICIAL INTELLIGENCE & MACHINE LEARNING (AM)

II/IV B. Tech SEMESTER-II

		L	T	P	M	C
22CS/AM/CYS/DS 221	**Discrete Mathematics**	**3**	**0**	**0**	**100**	**3**

Course Objectives:

At the end of the course, the student will

1. Introduce the concepts of mathematical logic.
2. Understand the combinatorial problems using counting principles,
3. Create generating functions and solve recurrence relations.
4. Use Directed & Un-Directed Graphs concepts and its applications.

Course Outcomes:

At the end of the course, the student will be able to

1. Understand formal methods of proof and propositional & First order logic to validate the propositional statements.
2. Apply techniques for counting the occurrences of discrete events including permutations, combinations with or without repetitions.
3. Analyse generating function and recurrence relations.
4. Construct the real-world problems using directed and undirected graphs.

UNIT – I

Boolean Algebra, Properties, Boolean polynomials, fundamentals Normal forms, Disjunctive Normal form, Complete and compliment of a Boolean expression in D.N.F, Conjunctive Normal form, Complete C.N.F and compliment of a Boolean expression in C.N.F

UNIT – II

Foundations: Sets, Relations and Functions, Methods of Proof and Problem Solving Strategies, Fundamentals of Logic, Logical Inferences, Methods of Proof of an implication, First order Logic & Other methods of proof, Rules of Inference for Quantified propositions, Disjunction normal forms, Conjunction normal forms, Mathematical Induction.

UNIT – III

Recurrence relations, Solving recurrence relations by Substitution and generating functions. The methods of characteristic roots. Relations and digraphs, Special properties of binary relations, Equivalence relations. Operations on relation.

UNIT – IV

Partial ordering relations, Poset, Hasse diagram, Lattices, Enumerations, Paths and Closures, Directed Graphs and Adjacency Matrices, Application : Topological Sorting.

UNIT-V:

Graphs: Basic Concepts, Isomorphisms and Subgraphs, hand shaking theorem, Planar Graphs, Euler's Formula; Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four Color Problem.

TEXT BOOK:

1. Toe L.Mott, Abraham Kandel& Theodore P.Baker, Discrete Mathematics forComputer Scientists & Mathematicians, PHI 2nd edition,2008.

2. J.P. Trembly and R. Manohar- Discrete Mathematics for Computer Scientists& Mathematicians, PHI Ltd., New Delhi, 2nd Edition, 2008.

3. Narasinghdeo Graph Theory , Narosa Publishers

4. Satyanarayana Bhavanari and Syamprasad Kuncham. "Discrete Mathematics and Graph Theory" by PHI, 2014 second edition.

5. SatyanarayanaBhavanari,T.V.PradeepKumar,Sk. Mohiddinshaw" Mathematical Foundations for Computer Sciences" by BS Publications, first editions, 2016.

REFERENCE BOOKS:

1. T. Sengadir- Discrete Mathematics-Pearson Education

2. C.L. Liu and D.P. Mohapatra-Elements of Discrete Mathematics, TataMcGraw-Hill ,3rd Edition, 2008.

3. Seymour Lipschutz, Lipson-Discrete Mathematics-Scaums outlines-TMH.

4. Santha-Discrete Mathematics-Cengage Learning

5. Kenneth H Rosen-Discrete Mathematics & its Applications , TMH, 6ht Edition,2009.

Mapping of Course Outcomes with Program Outcomes														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3	3	3	2	-	-	-	-	-	2	-	3	1	1
CO2	3	3	2	2	-	-	-	-	-	2	-	2	1	1
CO3	3	3	3	2	-	-	-	-	-	2	-	2	1	1
CO4	3	3	3	2	-	-	-	-	-	2	-	3	1	1

22CS/AM/CYS/DS 222 Essence of Indian Traditional Knowledge

L	T	P	M	C
3	0	0	100	3

Course Objectives:

The course will introduce the students to:

1. To get a knowledge in Indian Culture
2. To know Indian languages , literature and the fine arts in India.
3. To explore the science and scientists of Medieval and Modern India..

Course Outcomes:

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

1. Understand philosophy of Indian culture.
2. Distinguish the Indian languages and literature.
3. Learn the philosophy of ancient, medieval and modern India.
4. Acquire the information about the fine arts in India
5. Know the contribution of scientists in different eras

UNIT I:

Introduction to Culture: Culture, civilization, culture and heritage, general characteristics of culture, importance of culture in human literature, Indian Culture, Ancient India, Medieval India, Modern India.

UNIT II:

Indian Languages, culture and Literature: The role of Sanskrit, Significance of scriptures to current society, Indian philosophies, other Sanskrit literature, literature of South India.

UNIT III:

Religion and Philosophy: Religion and Philosophy in ancient India, Religion and Philosophy in Medieval India, Religious reform movements in Modern India(selected movements only).

UNIT IV:

Fine Arts in India: (Arts, Technology & Engineering): Indian painting, Indian handicrafts, music, divisions of Indian classic music, modern Indian music, Dance and Drama, Indian Architecture (Ancient, Medieval and Modern), Science and Technology in India, development of science in ancient, medieval and modern India.

UNIT V:

Education system in India: Education in Ancient, Medieval and Modern India, aims of Education, subjects, languages, science and scientists of Ancient India, Medieval and Modern India.

Text Books:

1. Kapil Kapoor, “Text and Interpretation: The India Tradition”, ISBN: 81246033375, 2005
2. “Science and Samskrit”, Samskrita BhartiPublisher, ISBN 13:978-8187276333, 2007
3. NCERT, “Position paper on Arts, Music, Dance and Theatre”, ISBN: 81-7450 494- X, 200

MAPPING OF CO's &PO's:

Mapping of Course Outcomes with Program Outcomes														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	-	-	-	-	-	2	1	2	1	-	-	-	1	1
CO2	-	-	-	-	-	2	1	2	1	-	-	-	1	1
CO3	-	-	-	-	-	2	1	2	1	-	-	-	1	1
CO4	-	-	-	-	-	2	1	2	1	-	-	-	1	1
CO5	-	-	-	-	-	2	1	2	1	-	-	-	1	2

22AM/DS 223	Advanced Data Structures	L	T	P	M	C
		3	0	0	100	3

Course Objectives:

- To understand the usage of algorithms in computing.
- To learn and use hierarchical data structures and its operations
- To learn the usage of graphs and its applications.
- To select and design data structures and algorithms that is appropriate for problems.

COURSE OUTCOMES:

1. Understanding the role of algorithms in computing and finding time complexity for recursive algorithms.
2. Analysehierarchical data structures and algorithms to solve computing problems.
3. Create algorithms using graph structure to solve real-life problems.
4. Understanding partition ADT.
5. Implementing data structures for handling strings.

UNIT-1

ROLE OF ALGORITHMS IN COMPUTING: Review of Basic Concepts, Asymptotic Analysis of Recurrences:The Substitution Method,iterative method, Recursion-Tree Method, master method. Randomized Algorithms, Randomized Quicksort, Algorithm Analysis Techniques - Amortized Analysis.

UNIT II:

HIERARCHICAL DATA STRUCTURES: Red-Black trees: Properties of Red-Black Trees – Rotations – Insertion – Deletion , Splay Trees: Rotations – Insertion – Deletion ,B-trees,B+ trees, Heap trees, priority queues, Binomial Heaps, Fibonacci Heaps.

UNIT III:

GRAPHS: Elementary Graph Algorithms: Representations of Graphs – Breadth-First Search – Depth-First Search – Topological Sort – Strongly Connected Components, Connected graphs, Articulation points in a graph, minimum cost spanning tree.Network Flows-Max flow, min-cut theorem, Ford-Fulkerson, Edmonds-Karp algorithm, Bipartite Matching.

UNIT IV:

Partition ADT: Disjoint sets, operations on sets, weighted union or union by rank, path compression, Permutations and Combinations.

Skip Lists: Skip list representation, Search and Update Operations on skip lists.

UNIT V:

Data Structures for Strings: Tries and Compressed Tries, Dictionaries allowing Errors in Queries, Suffix Trees and Suffix Arrays.

String Matching – The naive string-matching algorithm, The Rabin-Karp algorithm, The Knuth-Morris-Pratt algorithm.

Text Books:

1. Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, ―Data Structures and Algorithms‖, Pearson Education, Reprint 2006.

2. Advanced Data Structures, PETER BRASS City College of New York, CAMBRIDGE UNIVERSITY PRESS.

3. Classic Data Structures, DebasisSamanta, PHI.

4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, ―Introduction to Algorithms‖, Third Edition, Prentice-Hall, 2011.

Reference Books:

1. Robert Sedgewick and Kevin Wayne, ―ALGORITHMS‖, Fourth Edition, Pearson Education.

2. S.Sridhar,‖Design and Analysis of Algorithms‖, First Edition, Oxford University Press. 2014

MAPPING OF CO's &PO's

Mapping of Course Outcomes with Program Outcomes														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PS01**	**PSO2**
CO1	3	1	2	1	2	-	-	-	-	-	-	-	1	1
CO2	3	-	2	1	1	-	-	-	-	-	-	-	1	1
CO3	3	2	2	1	1	-	-	-	-	-	-	-	1	1
CO4	2	2	2	1	1	-	-	-	-	-	-	-	1	1
CO5	1	-	2	1	1	-	-	-	-	-	-	-	1	2

22AM/DS 224	Java Programming	L	T	P	M	C
		3	0	0	100	3

COURSE OBJECTIVES:

1. Understand the fundamental concepts of Java programming language.
2. Design and implement object-oriented programs using Java.
3. Implement algorithms and data structures in Java.
4. Develop graphical user interfaces (GUIs) using Java Swing
5. Utilize exception handling and error handling techniques in Java programs.

COURSE OUTCOMES:

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

1. Apply knowledge of basic syntax, data types, and control structures of Java.
2. Design and implement object-oriented programs using classes, objects, and inheritance.
3. Implement common algorithms and data structures using Java programming.
4. Develop GUI applications using Java Swing
5. Implement exception handling mechanisms and error handling techniques in Java programs

Syllabus:

UNIT - I

OOP concepts – Data abstraction, encapsulation, inheritance, benefits of inheritance, polymorphism, classes and objects, Procedural and object-oriented programming paradigms.

Java programming – History of Java, comments, data types variables constants, scope and life time of variables, operators, operator hierarchy, expressions, type conversion and casting, enumerated types, simple java stand along programs, arrays, console input and output, formatting output, constructors, methods, parameter passing, static fields and methods, access control, this reference, overloading methods and constructors, recursion, garbage collection, building strings, exploring string class. .

UNIT - II

Inheritance – inheritance hierarchies, super and sub classes, Member access rules, super keyword, preventing inheritance: final classes and methods, the Object class and its methods.
Polymorphism – dynamic binding, method overriding, abstract classes and methods.
Interfaces – interfaces vs. Abstract classes, defining an interface implementing interfaces, accessing implementations through interface references, extending interface.
Inner classes – Uses of inner classes, local inner classes, anonymous inner classes, static inner classes, examples.
Packages – Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages.

UNIT - III

Exception handling - Dealing with errors, benefits of exception handling, the classification of exceptions – exception hierarchy, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, re-throwing exceptions, exception specification, built in exceptions, creating own exception sub classes.

Multithreading - Differences between multiple processes and multiple threads, thread states, creating threads, interrupting threads, thread priorities, synchronizing threads, inter-thread communication, producer consumer pattern.

UNIT – IV

Collection Framework in Java - Introduction to Java Collections, Overview of Java Collection Frame work, Generics, Commonly used Collection classes – Array List, Vector, Hash table, Stack, Enumeration, Iterator, String Tokenizer, Random, Scanner, calendar and Properties

Files – Streams – byte streams, character streams, text input/output, binary input/output, random access file operations, File management using File class.

UNIT – V

GUI Programming with Java – The AWT class hierarchy, introduction to Swing, Swing vs AWT, Hierarchy for Swing components, Containers – JFrame, JApplet, JDialog, JPanel, Overview of some swing components – Jbutton, JLabel, JTextField, JTestArea, Simple swing applications, Layout management – Layout manger types – border, grid and flow

Event handling – Events, Event Sources, Event classes, Event Listeners, Relationship between Event sources and Listeners, Delegation event model, Examples: handling a button click, handling mouse events, Adapter classes.

TEXT BOOK:

1. Java Fundamentals – A comprehensive introduction, Herbert Schildt and Dale Skrien, TMH

REFERENCE BOOKS:

1. Java for Programmers, P.J.Deitel and H.M.Deitel, Pearson education (OR) Java: How to program P.J.Deitel and H.M.Deitel, PHI.
2. Object Oriented Programming through Java, P. Radha Krishna Universities Press.
3. Thinking in Java, Bruce Eckel, Pearson Education.
4. Herbert Schildt (2010), The complete reference, 7th edition, Tata Mc graw Hill, New Delhi
5. Cay S. Horstmann and Gary Cornell, "Core Java: Volume I – Fundamentals", Eighth Edition, Sun Microsystems Press, 2008.
6. Head First Java, O'rielly publications
7. T. Budd (2009), An Introduction to Object Oriented Programming, 3rd edition, Pearson Education, India.
8. Nino, F. A. Hosch (2002), An Introduction to programming and OO design using Java, John Wiley & sons, New Jersey.
9. Y. Daniel Liang (2010), Introduction to Java programming, 7th edition, Pearson education, India.

MAPPING OF CO's &PO's

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1		1	1			1						1	1
CO2	1	3			1								1	1
CO3	1	1	3	1									1	1
C04	1	1	2	3									1	1
C05	1	1	2	1	2								1	1

22AM/DS 225 **Operating Systems**

L	T	P	M	C
3	0	0	100	3

Objectives of the course:

To learn the fundamentals of Operating Systems.

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

Course Outcomes:

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

1. Remembering the fundamental concepts of an operating system functionality, and processes.
2. Apply the concepts of multithreading and IPC mechanisms and also analyse the performance of CPU scheduling algorithms, page replacement algorithms, and disk scheduling algorithms.
3. Understanding the methods to solve critical section problem and deadlock handling in a system.
4. Analyse the effectiveness and the hardware support required for contiguous, non-contiguous, and virtual memory management schemes.
5. Analyse various disk scheduling methods.

UNIT I:

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

UNIT II:

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

UNIT III:

Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer\ Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc.

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

UNIT IV:

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

UNIT V:

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.

Reference books:

1. Operating System Concepts Essentials, 9th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.
2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.
3. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
4. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison-Wesley
5. Design of the Unix Operating Systems, 8th Edition by Maurice Bach, Prentice-Hall of India
6. Understanding the Linux Kernel, 3rd Edition, Daniel P. Bovet, Marco Cesati,O'Reilly and Associates.

MAPPING OF CO's &PO's:

Mapping of Course Outcomes with Program Outcomes														
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	2	-	2	-		-	-	-	-	-	-	1	1	2
CO2	3	-	2	1	1	-	-	-	-	-	-	1	1	2
CO3	2	-	2	1	1	-	-	-	-	-	-	1	1	1
CO4	2	2	3	1	1	-	-	-	-	-	-	1	1	2
CO5	2	2	2	-	1	-	-	-	-	-	-	1	1	1

22AM 226	ARTIFICIAL INTELLIGENCE	L	T	P	M	C
		3	0	0	100	3

COURSE OUTCOMES

1. Apply Planning and Learning for solving AI problems.
2. Analyze a given problem and apply AI Techniques.
3. Interpret Statistical reasoning and filler structures.
4. Apply various problem planning systems and different learning methodologies

UNIT – I

Statistical Reasoning- Probability and Bayes' Theorem, Certainty Factors and Rule-based Systems, Bayesian Networks, Dempster-Shafer Theorem.

Weak and Strong Slot Filler Structures- Semantic nets, Frames, Conceptual dependency, Scripts.

UNIT – II

Planning- The Planning problem, planning with state space search, planning graphs, The Block world example, Components of a Planning System, Goal Stack Planning, Non Linear Planning using Constraint Posting, Hierarchical Planning, conditional planning, Continuous and Multi Agent planning.

UNIT - III

Learning- Role of learning, Learning by taking advice, Learning in problem solving, Learning from examples: induction, Explanation-based learning, Discovery, Analogy, Knowledge in Learning, Learning probabilistic Models, Reinforcement Learning, Forms of Learning, Supervised Learning, Learning Decision Trees, Evaluating and choosing best hypothesis, The theory of learning, Regression and Classification with Linear models

Unit-IV

Artificial Neural Networks: Neural network structures, Single-layer feedforward neural networks (perceptron's), Multilayer feed-forward neural networks, learning in multilayer networks, learning neural network structures.

UNIT-V

Natural Language Processing- Syntactic processing, Semantic analysis, Discourse and pragmatic processing, Statistical Natural language Processing, Spell Checking.

Parallel and Distributed AI: Parallelism in Reasoning Systems, Distributed Reasoning Systems

Text Books

1. Stuart Russell and Peter Norvig, ―Artificial Intelligence: A Modern Approach‖ , 3rd Edition, Prentice Hall.
2. Artificial Intelligence by Elaine Rich & Kevin Knight, Second edition, Tata Mcgraw hill.
3. Introduction to Artificial Intelligence and Expert Systems by Dan W. Paterson, PHI

REFERENCE BOOKS:

1. Artificial Intelligence – A modern Approach by Stuart Russell and Peter Norvig, Second edition, Pearson.
2. Artificial Intelligence by Saroj Kaushik, Cengage Learning.
3. A Classical Approach to Artificial Intelligence, M.C. Trivedi, Khanna Book Publishing, 2019.

Mapping of CO's & PO's:

COS	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2				1			1	1	1	2	1
CO2	3	3	1				1			1	1	1	2	1
CO3	3	3	2				1			1	1	1	2	1
CO4	3	2	1				1			1	1	1	2	1
CO5	3	3	2				1			1	1	1	2	1

22AM/DS 261 **Operating Systems Lab**

L	T	P	M	C
0	0	3	100	1.5

COURSE OBJECTIVES: To understand the functionalities of various layers of OSI model

· To explain the difference between hardware, software; operating systems, programs and files.
· Identify the purpose of different software applications.

COURSE OUTCOMES: At the end of the course the students are able to:

· Ability to implement inter process communication between two processes.

· Ability to design and solve synchronization problems.

· Ability to simulate and implement operating system concepts such as scheduling, Deadlock management, file management, and memory management.

List of Programs:

1. Simulate the following CPU scheduling algorithms. a) FCFS b) SJF c) Round Robin d) Priority.

2. Write a C program to simulate producer-consumer problem using Semaphores

3. Write a C program to simulate the concept of Dining-philosophers problem.

4. Simulate MVT and MFT.

5. Write a C program to simulate the following contiguous memory allocation Techniques a) Worst fit b) Best fit c) First fit.

6. Simulate all page replacement algorithms a)FIFO b) LRU c) OPTIMAL

7. Simulate all File Organization Techniques a) Single level directory b) Two level directory 8. Simulate all file allocation strategies a) Sequential b) Indexed c) Linked.

9. Simulate Bankers Algorithm for Dead Lock Avoidance.

10. Simulate Bankers Algorithm for Dead Lock Prevention.

11. Write a C program to simulate disk scheduling algorithms. a) FCFS b) SCAN c) C-SCAN

MAPPING OF CO's &PO's:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1												2		
CO2	1	1										2		2
CO3	1	1										2		1

22AM/DS 262	JAVA Lab	L	T	P	M	C
		0	0	3	100	1.5

COURSE OBJECTIVES:

1. To teach fundamentals of object oriented programming in Java. Understand various concepts of JAVA.
2. To familiarize Java environment to create, debug and run simple Java programs.
3. To demonstrate java compiler and eclipse platform and learn how to use Net Beans IDE to create Java Application.

COURSE OUTCOMES:

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

1. Implement Object oriented features using Java
2. Apply the concept of polymorphism and inheritance.
3. Implement exception handling
4. Develop network and window application using awt and swings.

Programs:

1. Write a Java program that prompts the user for an integer and then prints out all prime numbers up to that integer. (use Scanner class to read input)
2. Write a Java program that uses both recursive and non recursive functions to print the nth value in the Fibonacci sequence.
3. Write a Java program to multiply two given matrices.
4. Write a Java program that checks whether a given string is a palindrome or not.
5. Write a Java program to create a Student class and find the grade of the student.
6. Write a java program to create an abstract class named Shape contains number Of Sides ( ) method and Trapezoid, Triangle and Hexagon classes extends the class Shape.
7. Write a Java program to read copy content of one file to other by handling all file related exceptions.
8. Write a Java program that reads a file name from the user, and then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
9. Write a Java program that reads a file and displays the file on the screen.
10. 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.
11. Write a Java program for handling mouse events.
12. Develop simple calculator using Swings.
13. Write Java program for Randomized Quick sort method to sort a given list of integers in ascending order.
14. Write a Java program that uses functions to perform the following:
    a. Creating a SplayTree of integers ii) insertion iii) Traversing splay tree in preorder, inorder and postorder.
15. Write a java program to implement prims and krushkals algorithm.
16. Write a java program for BFS and DFS traversals.
17. Write a java program for Edmonds-Karp algorithm.
18. Write a java program to perform various operations on Disjoint sets

MAPPING OF CO's &PO's:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2	3	3	1	1	-	-	-	-	-	-	1	1	1
CO2	2	3	3	1	1	-	-	-	-	-	-	1	1	1
CO3	2	3	3	1	1	-	-	-	-	-	-	1	1	1
C04	-	-	-	-	-	-	-	2	2	2	-	-	1	1

22AM 263	AI Lab	L	T	P	M	C
		0	0	3	100	1.5

1. Write a python program to remove punctuations from the given string?
2. Write a python program to sort the sentence in alphabetical order?
3. Write a program to implement Hangman game using python.
4. Write a program to implement Tic-Tac-Toe game using python.
5. Write a python program to remove stop words for a given passage from a text file using NLTK?
6. Write a python program to implement stemming for a given sentence using NLTK?
7. Write a python program to POS (Parts of Speech) tagging for the give sentence using NLTK?
8. Write a python program to implement Lemmatization using NLTK?
9. Write a python program to for Text Classification for the give sentence using NLTK?

22CS/AM/CYS/DS 264	Communicative English Lab II	L	T	P	M	C
		0	0	3	100	1.5

Course Objectives:

The main course objective of ***Advanced English Communication Skills Lab*** is to develop the student's Non-Verbal Communication, Cognitive and Poignant Skills, Interview Skills, Employability and Interpersonal skills, which relate to situations in the work place. The skills imparted to the learners are body language, leadership, time management, team management, assertive skills, group discussions, interview techniques and positive work ethics ...etc.

The methodology includes Interactive sessions, Role Play, Team Work/Group Work/Pair Work and Peer Evaluation. The emphasis is on learning by doing to improve the learners' life skills.

Course Outcomes:

CO1	To realize the importance of communication skills in job arena To enhance the students ability to communicate
CO2	Able to learn vocabulary for GRE, TOEFL, IELTS,IES etc
CO3	Capable to participate in all recruitment procedures
CO4	Able to communicate effectively over a phone and proficient to demonstrate telephoning skills
CO5	Able to describe procedures and improves analytical thinking
CO6	Able to know the importance of personality development

Syllabus:

Module-l Communication Skills

I. Verbal

a) Types of Communication
b) Barriers to Communication
c) Strategies for effective communication

II. Nonverbal Skills -

a) Body Language – Voluntary and Involuntary
b) Kinesics
c) Facial Expressions
d) Proxemics
e) Oculesics
f) Haptics and Chronemics

Module-2: Advanced Vocabulary

a) Word list (GRE & TOEFL related)
b) One Word Substitutes
c) Idioms

Module-3: Employability Skills (Ref: 6)

a) Interview Skills
b) Group Discussion
c) Resume Writing

Module-4: Telephonic Skills

a) Formal &Informal interaction

b) Receiving Messages & Complaints

c) Tone modulation

Module-5: Descriptions

a) Process Description
b) Pictures
c) Narration

Module-6: Behavioural Skills

a) Emotional Intelligence
b) Positive Attitude
c) Team Work
d) Organization Skills

MAPPING:

	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	-	-	-	2	2	-	-	2	3	-	3	3	1	1
CO2	-	-	-	2	2			2	3	-	3	3	1	1
CO3	-	-	-	2	2	-	-	2	3	-	3	3	1	1
C04	-	-	-	2	2			2	3	-	3	3	1	1
CO5	-	-	-	2	2	-	-	2	3	-	3	3	1	1