JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

B.TECH-CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
COURSE STRUCTURE

(Applicable for the batches admitted from 2020-21)

I YEAR **I SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J1001	Mathematics-I (Linear Algebra & Calculus)	3	1	0	4
2	J1302	Engineering Graphics	1	0	4	3
3	J1202	Basic Electrical & Electronics Engineering	2	1	0	3
4	J1501	Programming For Problem Solving	3	1	0	4
5	J1203	Basic Electrical & Electronics Engineering Lab	0	0	3	1.5
6	J1502	Programming For Problem Solving Lab	0	0	3	1.5
7	J1507	IT and Engineering Workshop	1	0	2	2
		Total Credits	***10***	***3***	***12***	***19***

I YEAR **II SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J2002	Mathematics-II (ODE's and Multivariable Calculus)	3	1	0	4
2	J2007	Engineering Physics	3	1	0	4
3	J2008	Engineering Chemistry	3	1	0	4
4	J2011	English	2	0	0	2
5	J2508	Data Structures	3	0	0	3
6	J2509	Data Structures Lab	0	0	3	1.5
7	J2009	Engineering Physics & Engineering Chemistry Lab	0	0	3	1.5
8	J2012	English Language and Communication Skills Lab	0	0	2	1
		Total Credits	***14***	***3***	***8***	***21***

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

B.TECH-CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
COURSE STRUCTURE

(Applicable for the batches admitted from 2020-21)

II YEAR I SEM **III SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J3006	Applied Probability and Statistics	3	0	0	3
2	J3591	Introduction to Artificial Intelligence	3	0	0	3
3	J3419	Computer Organization	3	0	0	3
4	J3592	OOPs through Java	3	0	0	3
5	J3593	Python Programming	3	0	0	3
6	J3594	OOPs through Java Lab	0	0	2	1
7	J3595	Python Programming Lab	0	0	2	1
8	HSMC (H-102)	UHV-II ***(Mandatory Course)***	2	1	0	3
		Total Credits	***17***	***1***	***4***	***20***

II YR II SEM **IV SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J4004	Discrete Mathematics	3	0	0	3
2	J4596	Introduction to Machine Learning	3	0	0	3
3	J4518	Database Management Systems	3	0	0	3
4	J4513	Operating Systems	3	0	0	3
5	J4511	Design and Analysis of Algorithms	3	0	0	3
6	J4525	Database Management Systems Lab	0	0	4	2
7	J4516	Operating Systems Lab	0	0	3	1.5
8	J4597	ML Lab	0	0	3	1.5
		Total Credits	***15***	***0***	***10***	***20***
9	JMC02	Gender Sensitization ***(Mandatory Course)***	2	0	0	0

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING
B.TECH-CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

COURSE STRUCTURE
(Applicable for the batches admitted from 2020-21)

III YR I SEM **V SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J5528	Computer Networks	3	0	0	3
3	J5598	Automata and Compiler Design	2	1	0	3
3	J5599	Introduction to Data Science	3	0	0	3
4	J5538	Data Mining	3	0	0	3
5		**Professional Elective –I**	3	0	0	3
6	J55100	Data Science Lab	0	0	4	2
7	J55101	Computer Networks Lab	0	0	4	2
8	J5549	Data Mining Lab	0	0	4	2
		Total Credits	***14***	***1***	***12***	***21***
9	JMC03	Constitution of India***(Mandatory Course)***	**3**	**0**	**0**	0

III YR II SEM **VI SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J6520	Web Programming	2	1	0	3
2	J6535	Software Engineering	2	1	0	3
3		**Professional Elective –II**	2	1	0	3
4		**Professional Elective –III**	3	0	0	3
5		**Open Elective-I**	3	0	0	3
6	J6526	Web Programming Lab	0	0	4	2
7	J6537	OOAD Lab	0	0	2	1
8	J6580	Internship	0	0	2	1
		Total Credits	***12***	***3***	***8***	***19***

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

B.TECH-CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

COURSE STRUCTURE
(Applicable for the batches admitted from 2020-21)

IV YR I SEM **VII SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1	J65102	Deep Learning	3	0	0	3
2		**Open Elective-II**	3	0	0	3
3		**Professional Elective-IV**	2	1	0	3
4		**Professional Elective-V**	2	1	0	3
5	J75103	Deep Learning Lab	0	0	4	2
6	J75104	Data Analytics Lab	0	0	4	2
7	J7581	Mini Project	0	0	8	4
8	J7582	Technical Seminar	0	0	2	1
		Total Credits	***10***	***2***	***18***	***21***

IV YR II SEM **VIII SEMESTER**

S.No	Subject Code	Subject Name	L	T	P	Credits
1		**Open Elective-III**	3	0	0	3
		Open Elective-IV	3	0	0	3
3		**Professional Elective-VI**	2	1	0	3
4	J8583	Comprehensive Viva-Voce	0	0	4	2
5	J8584	Major Project	0	0	16	8
		Total Credits	***8***	***1***	***20***	***19***
	J8585	**NSS***	**0**	**0**	**0**	**2**

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES

(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

B.TECH-CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(Applicable for the batches admitted from 2020-21)

LIST OF PROFESSIONAL ELECTIVES

Professional Elective-I	**Professional Elective-II**
(J5533) Distributed Computing	(J6548) Software Testing Methodologies
(J5547) Cloud Computing	(J6559)Semantic Web and Social Networks
(J5553) Soft Computing	(J6570) Cyber security
(J5551) Internet of Things	(J6552)Human Computer Interaction
Professional Elective-III	**Professional Elective-IV**
(J6571) Genetic algorithm and its Applications	(J7456) Digital Image Processing
(J6572) Artificial Neural networks	(J7556) Natural Language Processing
(J6573) Fuzzy Logic and its Applications	(J7575) Statistical Machine Learning
(J6574)Computer Vision	(J7576) Nature Inspired Computing Techniques
Professional Elective-V	**Professional Elective-VI**
(J7577) Applied Machine Learning	(J8587) Advanced Algorithms
(J7578) Computational Neuroscience	(J8588) Speech Systems
(J7579) Intelligent Machining	(J8589) Virtual Reality
(J7586) Robotics: Computational Motion Planning	(J8590) Block Chain Technology

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

B.TECH-CSE (ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)
(Applicable for the batches admitted from 2020-21)

LIST OF OPEN ELECTIVES OFFERED AT COLLEGE LEVEL

Sl.No.	Subject code	Title
1.	**J_150**	Remote Sensing & GIS
2.	**J_151**	Traffic Engineering and Transportation Planning
3.	**J_152**	Disaster Preparedness & Planning
4.	**J_153**	Environmental Impact Assessment
5.	**J_219**	Control Systems
6.	**J_223**	Renewable Energy Sources
7.	**J_224**	Energy Storage Systems
8.	**J_238**	Industrial Electricals Systems
9.	**J_249**	Electrical Engineering Materials
10.	**J_250**	Neural Networks & Fuzzy Logic
11.	**J_351**	Basic Mechanical Engineering
12.	**J_352**	Applied Mechanics
13.	**J_353**	Material Science
14.	**J_354**	Basics of Thermodynamics
15.	**J_355**	Strength of Materials
16.	**J_356**	Modeling and Simulation of manufacturing systems
17.	**J_357**	Mechatronics
18.	**J_358**	Finite Element Analysis
19.	**J_359**	Nano Technology
20.	**J_402**	Signals and Systems
21.	**J_409**	Digital System Design
22.	**J_410**	Electromagnetic Waves and Transmission Lines
23.	**J_414**	IC Applications
24.	**J_415**	Digital Signal Processing
25.	**J_418**	Bio Medical Electronics
26.	**J_419**	Computer Organization
27.	**J_422**	Linear Control Systems
28.	**J_424**	Microprocessors and Microcontrollers
29.	**J_434**	Image and Video Processing
30.	**J_437**	Embedded Systems
31.	**J_447**	Wireless Sensor Networks
32.	**J_454**	Microprocessors and Interfacing
33.	**J_456**	Digital Image Processing
34.	**J_518**	Database Management Systems
35.	**J_528**	Computer Networks

36.	**J_529**	Machine Learning
37.	**J_538**	Data Mining
38.	**J_539**	Cryptography & Network Security
39.	**J_547**	Cloud Computing
40.	**J_551**	Internet of Things(IoT)
41.	**J_553**	Soft Computing
42.	**J_555**	Data Science & Big Data Analytics
43.	**J_556**	Natural Language Processing
44.	**J_559**	Semantic Web& Social Networks
45.	**J_560**	E-Commerce
46.	**J_563**	IT Infrastructure Management
47.	**J_564**	Mobile Application Development
48.	**J_565**	System Modeling and Simulation
49.	**J_566**	Free and Open Source Softwares
50.	**J_567**	Android Development
51.	**J_568**	Data Analysis using Open Source Tool
52.	**J_569**	IOS Development
53.	**J_E01**	Management Science
54.	**J_E02**	Managerial Economics and Financial Analysis
55.	**J_E03**	Total Quality Management
56.	**J_E04**	Global Marketing
57.	**J_E05**	Green Marketing
58.	**J_E06**	Intellectual Property Rights
59.	**J_E07**	Supply Chain Management
60.	**J_E08**	Statistical Quality Control
61.	**J_E09**	Financial Statement Analysis and Reporting
62.	**J_E10**	Micro Small Medium Enterprises Management
63.	**J_E11**	Entrepreneurship Development
64.	**J_E12**	Organizational Behavior
65.	**J_E13**	Industrial Management
66.	**J_E14**	Production and Operations Management
67.	**J_E15**	Economic Policies of India
68.		
Note: '_' represents the applicable semester code		
Note: The syllabus of open elective subjects is kept available in the Departments and website		

COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J1501) PROGRAMMING FOR PROBLEM SOLVING

B.Tech. I Year I SEM: CSE (AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:

1.To introduces the basics of computers and information technology.
2. To educate problem solving techniques.
3. To impart programming skills in C language.
4. To practice structured programming to solve real life problems.
5. To study the concepts of Assembler, Macro Processor, Loader and Linker

Syllabus

UNIT-I

History and Classifications of Computers – Components of a Computer – Working Principle of Computer – Hardware – Software and its Types – Applications of Computers –Network and its Types – Internet and its services – Intranet– Extranet – Generations of Programming Languages Introduction to Number System.

UNIT-II

Problem solving techniques – Program development life-cycle – Algorithm – Complexities of Algorithm – Flowchart – Pseudo code. Introduction to C –C Program Structure – C tokens: Keyword, Identifiers, Constants, Variable, Data types (simple and user-defined) – Operators and its types – Operator Precedence – Expression Evaluation – Type Conversion – Input/output operations.

UNIT-III

Branching Statements – Looping Statements – Arrays – Multidimensional arrays. Functions: Function Prototype, Passing Arguments to Function – Call by Value and Call by Reference – Nested function call – Library Functions – User-defined Functions – Recursion. Strings – String I/O functions, String Library functions – Storage classes

UNIT-IV

Structures – Arrays and Structures – Nested structures – Structure as Argument to functions– Union Pointers – Declaration, Initialization and Accessing Pointer variable – Pointers and arrays – pointers as argument and return value – Pointers and strings - pointers and structures.

UNIT-V

Introduction to File Concepts in C – File types – I/O operations on files – File modes – Random access to files – Command line arguments. Dynamic Memory Allocation: MALLOC, CALLOC, FREE, REALLOC Introduction to preprocessor – Macro substitution directives – File inclusion directives –Compiler Control directives – Miscellaneous directives.

Text Books:

1. J. B. Dixit, “Computer Fundamentals and Programming in C”, Firewall Media, 2009.
2. Balagurusamy. E, “Programming in ANSI C”, Tata McGraw Hill, Sixth edition, 2012.

Reference Books:

1.Ashok N Kamthane, “Computer Programming”, Pearson education, Second Impression, 2008.
2. Venugopal.K and Kavichithra.C, “Computer Programming”, New Age International Publishers, First Edition, 2007.
3. Byron S Gottfried, “Programming with C”, Schaum’s Outlines, Second Edition, Tata McGraw-Hill, 2006.
4. Dromey R.G., “How to Solve it by Computer”, Pearson Education, Fourth Reprint, 2007.
5. Kernighan,B.W and Ritchie,D.M, “The C Programming language”, Second Edition, Pearson Education, 2006.

Course Outcomes:

1.Know the fundamentals of computers
2.Uunderstand applying logical skills for problem solving
3.Learn C programming language concepts
4.Apply C programming language concepts for problem solving
5.Gain knowledge in using memory management techniques in c programming

(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J1502) PROGRAMMING FOR PROBLEM SOLVING LABORATORY

B.Tech. I Year I SEM: CSE (AI&ML)

L	T	P	C
0	0	4	2

Course Objectives:

1.To study and understand the use of OS commands
2.To expose the undergraduate students to the practical implementation of C Programming concepts
3.To improve students capability in applying C Programming for problem solving.
4. To make students use effective memory management techniques in programming
5. To expose students to modular programming concepts in problem solving

LIST OF EXPERIMENTS

Week 1: Study of OS commands
Week 2: Study of Compilation and execution of simple C programs
Week 3: Basic C Programs
a.Arithmetic Operations
b. Area and Circumference of a circle
c. Swapping with and without Temporary Variables
Week 4: Programs using Branching statements
a. To check the number as Odd or Even
b. Greatest of Three Numbers
c. Counting Vowels
d. Grading based on Student's Mark
Week 5: Programs using Control Structures
a. Computing Factorial of a number
b. Fibonacci Series generation
c. Prime Number Checking
d. Computing Sum of Digit
Week 6: Programs using String Operations
a. Palindrome Checking
b. Searching and Sorting Names
Week 7: Programs using Arrays
Week 8: Programs using Functions
a. Computing nCr

b. Factorial using Recursion
c. Call by Value and Call by Reference

Week 9: Programs using Structure
a. Student Information System
b. Employee Pay Slip Generation
c. Electricity Bill Generation
Week 10: Programs using Pointers
a. Pointer and Array
b. Pointer to function
c. Pointer to Structure
Week 11: Programs using File Operation
a. Counting No. of Lines, Characters and Black Spaces
b. Content copy from one file to another
c. Reading and Writing Data in File

Text Books:

1. J. B. Dixit, "Computer Fundamentals and Programming in C", Firewall Media, 2009.
2. Balagurusamy. E, "Programming in ANSI C", Tata McGraw Hill, Sixth edition, 2012.

Course Outcomes:

1. Learn practical implementation of C programming language concepts.
2. Debug and document programs in C.
3. Know usage of logical skills in developing C programs.
4. Apply effective memory management techniques for problem solving
5. Understand the file management techniques

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J1507) IT AND ENGINEERING WORKSHOP

B.Tech. I Year I SEM: CSE (AI&ML)

L	T	P	C
1	0	2	2

Course Objectives:

1. The IT Workshop is a training lab course to get training on PC Hardware, Internet & World Wide Web and Productivity tools for documentation, Spreadsheet computations, and Presentation.
2. To introduce to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers, hardware and software level troubleshooting process.
3. To introduce connecting the PC on to the internet from home and workplace and effectively usage of the internet, Usage of web browsers, email, newsgroups and discussion forums.
4. To introduce the usage of Productivity tools in crafting professional word documents, excel Spread sheets and power point presentations using open office tools and LaTeX.
5. To provide hands on experience about use of different engineering materials, tools,equipments and processes those are common in the engineering field.

LIST OF EXPERIMENTS:

Machine Issues: (2 problems)

Problem 1: **Hardware Troubleshooting**: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor followed by a viva.

Problem 2: **a) Software Troubleshooting:** Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition. The work done should be verified by the instructor and followed by a viva.

b) OS Installation and Hard Drive Partitioning

Internet & World Wide Web (4 Problems)

Problem 3: **Orientation & Connectivity Boot Camp:** Students should get connected to their Local Area Network,access the Internet and transfer files from one system to another system across the LAN. In the process they configure the TCP/IP setting. Finally students should demonstrate, to the instructor, how to access the websites

and email. If there is no internet connectivity preparations need to be made by the instructors to simulate the WWW on the LAN.

Problem 4: **Web Browsers, Surfing the Web:** Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Problem 5: **Search Engines & Netiquette:** Students should know what search engines are and how to use the search engines. A few topics would be given to the students for which they need to search on Google. This should be demonstrated to the instructors by the student.

Problem 6: **Cyber Hygiene:** Students would be exposed to the various threats on the internet and would be asked to configure their computer to be safe on the internet. They need to first install antivirus software, configure their personal firewall and windows update on their computer. Then they need to customize their browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

Productivity Tools: LaTeX and FOSS Text Processing Tools (4 Problems)

Problem 7: **Document Preperation:** The mentor needs to give an overview of LaTeX and FOSS tools: Importance of LaTeX and FOSS tools for text processing, Details of the four tasks and features that would be covered in each, Using LaTeX and text Processor – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter.

Problem 8: **Using LaTeX and FOSS Text Processing Tools** to create project certificate. Features to be covered: Formatting Fonts, Drop Cap, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and FOSS Text Processing Tools.

Problem 9: **Text Layouts :**abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Problem 10: **Creating a Newsletter:** Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbars and text highlights, Formatting Images, Textboxes and Paragraphs using FOSS.

Spreadsheet: (3 Problems)

Problem 11: **Spreadsheet Orientation:** The mentor needs to tell the importance of FOSS Spreadsheet tools, give the details of the four tasks and features that would be covered in each.

Problem 12: Using Spreadsheet –Accessing, overview of toolbars, saving files, Using help and resources, Creating a Scheduler **,** Gridlines, Format Cells, Summation, auto fill and Formatting Text.

Problem 13: **Calculating GPA** - Features to be covered:- Cell Referencing, Formulae in spreadsheet – average, standard deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Aggregates and lookups, Sorting, , Conditional formatting

LaTeX and FOSS Slide shows (3 Problems)

Problem 14: Students will be working on basic slide show utilities and tools which help them create basic power point presentation. Topic covered during this problem includes: Slide Layouts, Inserting Text, Text high lighting Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows in both LaTeX and FOSS Tool. Students will be given model slide shows which need to be replicated. (Exactly how it's asked).

Problem 15: Second Problem helps students in making their presentations interactive. Topic covered during this problem includes: Hyperlinks, Inserting –Images, Image galleries, Audio, Video, Objects, Tables and Charts

Problem 16: Concentrating on the in and out of FOSS Slide shows and presentations in LaTeX. Helps them learn best practices in designing and preparing slide shows. Topic covered during this problem includes: Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), Inserting – Background, textures, Design Templates, Hidden slides.

Engineering Workshop

Workshop Practice: (Two exercises are required to perform from each trade)

1. Fitting
2. Carpentry
3. Tin – Smithy
4. House – wiring
5. Plumbing

Text Books:

1. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
2. LaTeX Companion – Leslie Lamport, PHI/Pearson.
3. Introduction to computers, Peter Norton, 6/e Mc Graw Hill.
4. Upgrading and Repairing, PC's 18th e, Scott Muller QUE, Pearson Education.
5. Complex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech
6. IT Essentials PC Hardware and Software Companion Guide Third Edition
7. PC Hardware and A+ Handbook – Kate J. Chase PHI
8. Workshop Manual – P.Kannaiah / K.L.Narayana/Scitech Publishers.

Course Outcomes:

1. Apply knowledge for computer assembling and software installation.
2. Ability how to solve the trouble shooting problems.
3. Apply the tools for preparation of PPT, Documentation and budget sheet etc.
4. Usage of Web browsers to access Internet, Search Engines
5. Ability to apply the knowledge of FOSS and Practice on manufacturing of components using workshop trades including plumbing,fitting, carpentry, tinsmithy and house wiring.

Note: Students should be able to use FOSS like Open Office, Zoho Docs, Libre Office, Soft Maker Free Office, Google Doss, Think Free Online, Live Document etc.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J1508) DATA STRUCTURES

B.Tech. I Year II SEM: CSE (AI&ML)

L	T	P	C
3	1	0	4

Course Objectives:
1. Basic data structures and its usage in handling real world applications
2. Representing the data using linear data structures such as queues, circular queues, dequeue, priority queue, and using non-linear data structures such as trees
3. Representing and retrieving the data in the form of various types of trees and graph data structures
4. Searching of data with the help of various search methods, to sort data using various sorting methods
5.Store and retrieve data effectively using various hashing methods

Syllabus:

UNIT - I
Basic Concepts: Algorithm specification- Introduction, Performance analysis and measurement- Performance analysis, Performance measurement.
Arrays: The arrays as an abstract data type, the polynomial abstract data type, sparse matrices- Introduction, Sparse matrix representation, transposing a matrix.
Stacks and Queues: The stack abstract data Type, The queue abstract data type,
Evaluation of expressions- Expressions, Postfix notations, Infix to postfix, Infix to prefix.
UNIT - II
Linked Lists: Singly linked lists and chains, Representing chains, Circular lists, Linked stacks and Queues, Polynomials, Doubly linked lists.
Trees: Introduction, Binary trees- The abstract data type, Properties of binary trees, Binary tree representations, Binary tree traversals and Tree iterator-Introduction, Inorder traversal, Preorder traversal, Postorder traversal, Iterative traversals. Threaded binary trees, Heaps, Binary search trees- Definition, Searching a binary search tree, Insertion into a binary search tree, Deletion from a binary search tree, Joining and Splitting binary search trees, Height of a binary search tree.
UNIT - III
Graphs: The graph abstract data type- Introduction, Definition, Graph representation, Elementary graph operations- Depth first search, Breadth first search, Connected components, Spanning trees, Minimum cost spanning trees- Kruskal's algorithm, Prim's algorithms, Shortest paths- All pairs shortest paths.
Efficient Binary Search Trees: Optimal binary search trees, AVL trees.
Multiway Search Trees: M-way search trees, B-trees, B+ trees.
UNIT - IV

Sorting and Searching: Searching, Search techniques- Binary search, Fibonacci search, Sorting-Types of sorting, General sort concepts, Bubble sort, Insertion sort, Selection sort, Quick sort, Heap sort, Merge sort, Comparison of all sorting methods.

UNIT-V

Hashing: Introduction, Key terms and issues, Hash functions, Collision resolution strategies, Hash table overflow, Extendible hashing.

Text Book:

1. Ellis Horowitz, Sartaj Sahani, Dinesh Metha, "Fundamentals of Data Structures in C++", *Universities Press*, 2nd Edition, ISBN-978 81 7371 606 5, 2008.
2. Varsha H.Patil, "Data Structures Using C++", *Oxford University Press*, 1st Edition, ISBN-10: 0-19-806623-6, ISBN-13: 978-0-19-806623-1, 2012 (Chapters: 9, 11)

Reference Books:

1. D. Samanta, "Classic Data Structures", *Prentice Hall India*, 2nd Edition, ISBN- 978-203-3731-2, 2009.
2. Mark Allen Weiss, "Data Structure & Algorithm Analysis in C++", *Pearson Education*, 3rd Edition, ISBN-10: 81-3171-474-8, ISBN-13:97-8813-1714-744, 2007.

Course Outcomes:

1. Implement the basics of data structures in handling real world applications
2. Represent data using linear data structures such as queues, circular queues, dequeue, priority queue, and using non-linear data structures such as trees and graphs
3. Represent and retrieve the data in the form of various non-linear data structures like trees and graphs
4. Search for data with the help of various searching techniques
5.Store and retrieve data using various hashing techniques

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J1509) DATA STRUCTURES LAB

B.Tech. I Year II SEM: CSE (AI&ML)

L	T	P	C
0	0	4	2

Course Objectives:
This course will develop students' knowledge in/on…
1. Concepts, operations and implementation details of various data structures
2. Implementing the different algorithms using C++ programming language
3. Improving the student capability in applying various data structures in different applications
4. Different types of sorting techniques
5.Different types of searching techniques

LIST OF EXPERIMENTS:

Experiment-I
1. Program to implement array operations.
2. Program to display sparse representation for a given m*n matrix.
3. Program to read a sparse matrix and display its transpose.

Experiment-II
4. Program to perform addition of two sparse matrices.
5. Program to implement stack operations using arrays

Experiment-III
6. Program to implement multiple stacks in single array.
7. Program to convert infix expression into postfix.
8. Program to convert given infix expression into prefix notation.
9. Program to evaluate given postfix expression.

Experiment-IV
10. Program to implement queue operations using arrays.
11. Program to implement circular queue operations using arrays.

Experiment-V
12. Program to create single linked list and implement its operations.
i. Insert
ii. Delete
iii. Search
iv. Reverse
13. Program to create single linked list and implement its operations with separate header node.
i. Insert
ii. Delete
iii. Search
iv. Reverse

Experiment-VI

14. Program to implement double linked list and its operations.
15. Program to implement double linked list and its operations with separate header node.

Experiment-VII

16. Program to implement circular single linked list and its operations.
17. Program to implement circular double linked list and its operations.

Experiment-VIII

18. Program to implement stack operations using linked list.
19. Program to implement queue operations using linked list.

Experiment-IX

20. Implementation of binary tree and its traversal techniques using recursive and non recursive methods.
21. Program to create a binary search tree and perform the tree operations.
a) Insertion of a node b) Deleting a node.

Experiment-X

22. Implement the following graph traversal techniques.
a) Depth first search b) Breadth first search.

Experiment-XI

23. Program to implement Fibonacci Search.
24. Program to implement insertion sort technique.
25. Program to implement selection sort technique.
26. Program to implement quick sort technique.

Experiment-XII

27. Program to implement merge sort technique.
28. Program to implement heap sort technique.

Text Book:

1. Ellis Horowitz, Sartaj Sahani, Dinesh Metha, "Fundamentals of Data Structures in C++", *Universities Press*, 2nd Edition, ISBN-978 81 7371 606 5, 2008.
2. Varsha H.Patil, "Data Structures Using C++", *Oxford University Press*, 1st Edition, ISBN-10: 0-19-806623-6, ISBN-13: 978-0-19-806623-1, 2012 (Chapters: 9, 11)

Course Outcomes (COs):

Upon completion of this course, students will be able to…
1. Know practical knowledge about implementing various data structures using C++
2. Uunderstand the knowledge about how various data structures will be implemented like Arrays, stacks, queues, linked list, trees, and graphs
3. Implement various sorting techniques
4. Implement various searching techniques
5. Apply these data structures efficiently to develop different software applications

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J3006) APPLIED PROBABILITY AND STATISTICS

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:

1. To learn the theory of Probability, and probability distributions of single and multiple random variables
2. To learn the sampling theory and testing of hypothesis and making inferences
3. To learn stochastic process and Markov chains.

UNIT - I

Probability: Sample Space, Events, Counting Sample Points, Probability of an Event, Additive Rules, Conditional Probability, Independence, and the Product Rule, Bayes' Rule. Random Variables and Probability Distributions: Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Statistical Independence.

UNIT - II

Mathematical Expectation: Mean of a Random Variable, Variance and Covariance of Random Variables, Means and Variances of Linear Combinations of Random Variables, Chebyshev's Theorem. Discrete Probability Distributions: Introduction and Motivation, Binomial, Distribution, Geometric Distributions and Poisson distribution.

UNIT - III

Continuous Probability Distributions: Continuous Uniform Distribution, Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution, Normal Approximation to the Binomial, Gamma and Exponential Distributions. Fundamental Sampling Distributions: Random Sampling, Some Important Statistics, Sampling Distributions, Sampling Distribution of Means and the Central Limit Theorem, Sampling Distribution of S2, t –Distribution, F-Distribution.

UNIT - IV

Estimation & Tests of Hypotheses: Introduction, Statistical Inference, Classical Methods of Estimation.: Estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance Limits, Estimating the Variance, Estimating a Proportion for single mean , Difference between Two Means, between Two Proportions for Two Samples and Maximum Likelihood Estimation. Statistical Hypotheses: General Concepts, Testing a Statistical Hypothesis, Tests Concerning a Single Mean, Tests on Two Means, Test on a Single Proportion, Two Samples: Tests on Two Proportions.

UNIT - V

Stochastic Processes and Markov Chains: Introduction to Stochastic processes- Markov process. Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, nstep transition probabilities, Markov chain, Steady state condition, Markov analysis.

TEXT BOOKS:

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Keying Ye, Probability & Statistics for Engineers & Scientists, 9th Ed. Pearson Publishers.
2. S C Gupta and V K Kapoor, Fundamentals of Mathematical statistics, Khanna publications.
3. S. D. Sharma, Operations Research, Kedarnath and Ramnath Publishers, Meerut, Delhi

REFERENCE BOOKS:

1. T.T. Soong, Fundamentals of Probability And Statistics For Engineers, John Wiley & Sons Ltd, 2004.
2. Sheldon M Ross, Probability and statistics for Engineers and scientists, Academic Press.

Course Outcomes: After learning the contents of this subject the student must be able to

1. Apply the concepts of probability and distributions to some case studies
2. Correlate the material of one unit to the material in other units
3. Resolve the potential misconceptions and hazards in each topic of study.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J3591) INTRODUCTION TO ARTIFICIAL INTELLIGENCE

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:
1.To learn the difference between optimal reasoning vs human like reasoning
2.To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
3.To learn different knowledge representation techniques
4.To understand the applications of AI: namely Game Playing, Theorem Proving,Expert Systems, Machine Learning and Natural Language Processing

UNIT - I
Introduction, History, Intelligent Systems, Foundations of AI, Sub areas of AI, Applications. Problem Solving - State-Space Search and Control Strategies: Introduction, General Problem Solving, Characteristics of Problem, Exhaustive Searches, Heuristic Search Techniques, Iterative-Deepening A*, Constraint Satisfaction. Game Playing, Bounded Look-ahead Strategy and use of Evaluation Functions, Alpha-Beta Pruning

UNIT - II
Logic Concepts and Logic Programming: Introduction, Propositional Calculus, Propositional Logic, Natural Deduction System, Axiomatic System, Semantic Tableau System in Propositional Logic, Resolution Refutation in Propositional Logic, Predicate Logic, Logic Programming. Knowledge Representation: Introduction, Approaches to Knowledge Representation, Knowledge Representation using Semantic Network, Extended Semantic Networks for KR, Knowledge Representation using Frames.

UNIT - III
Expert System and Applications: Introduction, Phases in Building Expert Systems, Expert System Architecture, Expert Systems Vs Traditional Systems, Truth Maintenance Systems, Application of Expert Systems, List of Shells and Tools. Uncertainty Measure - Probability Theory: Introduction, Probability Theory, Bayesian Belief Networks, Certainty Factor Theory, Dempster-Shafer Theory.

UNIT-IV
Machine-Learning Paradigms: Introduction. Machine Learning Systems. Supervised and Unsupervised Learning. Inductive Learning. Learning Decision Trees (Text Book 2), Deductive Learning. Clustering, Support Vector Machines. Artificial Neural Networks: Introduction, Artificial Neural Networks, Single-Layer FeedForward Networks, Multi-Layer Feed-Forward Networks, Radial-Basis Function Networks, Design Issues of Artificial Neural Networks, Recurrent Networks.

UNIT-V

Advanced Knowledge Representation Techniques: Case Grammars, Semantic Web Natural Language Processing: Introduction, Sentence Analysis Phases, Grammars and Parsers, Types of Parsers, Semantic Analysis, Universal Networking Knowledge.

TEXT BOOKS:

1. Saroj Kaushik. Artificial Intelligence. Cengage Learning. 2011
2. Russell, Norvig: Artificial intelligence, A Modern Approach, Pearson Education, Second Edition. 2004

REFERENCE BOOK:

1. Rich, Knight, Nair: Artificial intelligence, Tata McGraw Hill, Third Edition 2009.
2. Introduction to Artificial Intelligence by Eugene Charniak, Pearson.
3. Introduction to Artificial Intelligence and expert systems Dan W.Patterson. PHI.
4. Artificial Intelligence by George Fluger rearson fifth edition.

Course Outcomes:

1.Possess the ability to formulate an efficient problem space for a problem expressed in English.
2.Possess the ability to select a search algorithm for a problem and characterize its time and space complexities.
3.Possess the skill for representing knowledge using the appropriate technique
4. Possess the ability to apply AI techniques to solve problems of Game Playing, Expert Systems, Machine Learning and Natural Language Processing

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J3591) OOPS THROUGH JAVA

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:

1. To introduce the object oriented programming concepts.
2. To understand object oriented programming concepts, and apply them in solving problems.
3. To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes
4. To introduce the implementation of packages and interfaces
5. To introduce the concepts of exception handling and multithreading.
6. To introduce the design of Graphical User Interface using applets and swing controls.

UNIT - I

Object-oriented thinking- A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes, Methods and Classes, String handling.

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

UNIT - II

Packages- Defining a Package, CLASSPATH, Access protection, importing packages.

Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Stream based I/O(java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

UNIT - III

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

UNIT - IV

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque.
Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hashtable ,Properties, Stack, Vector More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

UNIT - V

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

A Simple Swing Application, **Applets –** Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls- JLabel and Image Icon, JText Field, The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs.

TEXT BOOKS

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

REFERENCE BOOKS

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object oriented Application Development, R. A. Johnson, Cengage Learning.

Course Outcomes:

1. Able to solve real world problems using OOP techniques.
2. Able to understand the use of abstract classes.
3. Able to solve problems using java collection framework and I/o classes.
4. Able to develop multithreaded applications with synchronization.
5. Able to develop applets for web applications.
6. Able to design GUI based applications

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J3592) OOPS THROUGH JAVA LAB

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
0	0	2	1

Course Objectives:
1.To write programs using abstract classes.
2. To write programs for solving real world problems using java collection frame work.
3. To write multithreaded programs.
4. To write GUI programs using swing controls in Java.
5. To introduce java compiler and eclipse platform.
6. To impart hands on experience with java programming.

Note:
1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform.
2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

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

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

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

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

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

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

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

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

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

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

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

12. Write a Java program that correctly implements the producer – consumer problem using the concept of interthread communication.

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

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

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

REFERENCE BOOKS

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition *Pearson* education.
2. Thinking in Java, Bruce Eckel, *Pearson* Education.
3. Java Programming, D. S. Malik and P. S. Nair, *Cengage* Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, *Pearson*.

Course Outcomes:

1. Able to write programs for solving real world problems using java collection frame work.
2. Able to write programs using abstract classes.
3. Able to write multithreaded programs.
4. Able to write GUI programs using swing controls in Java.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J3593) PYTHON PROGRAMMING

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:

1. To be able to introduce core programming basics and program design with functions using Python programming language.
2. To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.
3. To understand the high-performance programs designed to strengthen the practical expertise.

UNIT - I

Introduction to Python, Installing Python. How a Program Works, Using Python, Program Development Cycle, Input, Processing, and Output, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations,Operators. Type conversions, Expressions, More about Data Output. Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables. Repetition Structures: Introduction, while loop, for loop, Calculating a Running Total, Input Validation Loops, Nested Loops.

Data types and Expressions: Strings, Assignment and Comments, Numeric Data Types and Character Sets, Expressions, Functions and Modules.

UNIT - II

Control Statements: Definite Iteration, Formatting Text for Output, Selection, Conditional Iteration.

File and Exceptions: Introduction to File Input and Output, Using Loops to Process Files, Processing Records, Exceptions.

Functions: Introduction, Defining and Calling a Void Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions-Generating Random Numbers, The math Module, Storing Functions in Modules.

UNIT - III

Strings and Text Files: Accessing Characters and Substrings in a String, Strings and Number System, String Methods, Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings. Text Files, Data Encryption, Lists, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two-Dimensional Lists, Tuples Sequences, Tuples. Dictionaries and Sets: Dictionaries, Sets, Serializing Objects. Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms.

UNIT - IV

Design with Classes: Classes and Objects, Classes and Functions, Classes and Methods, Working with Instances, Inheritance and Polymorphism.

Object-Oriented Programming: Procedural and Object-Oriented Programming, Classes, techniques for Designing Classes.

UNIT - V

Graphical User Interfaces: Behavior of terminal based programs and GUI-based programs,Coding simple GUI-based programs, other useful GUI resources. GUI Programming:Graphical User Interfaces, Using the tkinter Module, Display text with Label Widgets,Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes, Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons.

Simple Graphics and Image Processing: Overview of Turtle Graphics, Two dimensional Shapes, Colors and RBG System, Image Processing.

TEXT BOOKS:

1. Kenneth A. Lambert, The Fundamentals of Python: First Programs, 2011, Cengage Learning.
2. Think Python First Edition, by Allen B. Downey, Orielly publishing

REFERENCE BOOKS:

1. Introduction to Computation and Programming Using Python. John V. Guttag, The MIT Press.
2. James Payne, Beginning Python using Python 2.6 and Python 3, Wrox publishing
3. Paul Gries, Practical Programming: An Introduction to Computer Science using Python 3,The Pragmatic Bookshelf, 2nd edition (4 Oct. 2013)
3. Charles Dierach, Introduction to Computer Science using Python

Course Outcomes:

1.Student should be able to understand the basic concepts scripting and the contributions of scripting language
2. Ability to explore python especially the object oriented concepts, and the built in objects of Python.
3. Ability to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete event simulations

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J3594) PYTHON PROGRAMMING LAB

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
0	0	2	1

Course Objectives:

1.To be able to introduce core programming basics and program design with functions using Python programming language.

2.To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.

3.To understand the high-performance programs designed to strengthen the practical expertise.

List of Programs:

1. Write a program to demonstrate different number data types 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 sub-string from a given string.
4. Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST 2017”
5. Write a 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 temperatures to and from Celsius, Fahrenheit. [Formula : c/5 = f-32/9]

10. Write a Python program to construct the following pattern, using a nested for loop

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

11. Write a Python script that prints prime numbers less than 20.
12. Write a python program to find factorial of a number using Recursion.

13. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
15. Write a python program to define a module and import a specific function in that module to another program.
16. Write a script named **copyfile.py**. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
18. Write a Python class to convert an integer to a roman numeral.
19. Write a Python class to implement pow(x, n)
20. Write a Python class to reverse a string word by word.

Course Outcomes:

- Student should be able to understand the basic concepts scripting and the contributions of scripting language
- Ability to explore python especially the object oriented concepts, and the built in objects of Python.
- Ability to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete event simulations

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

HSMC (H-102) UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY

B.Tech. II Year I SEM: CSE (AI&ML)

L	T	P	C
2	1	0	3

Pre-requisites: None. Universal Human Values 1 (desirable)

1. OBJECTIVE: The objective of the course is four fold:

1. Development of a holistic perspective based on self-exploration about themselves (human being), family, society and nature/existence.
2. Understanding (or developing clarity) of the harmony in the human being, family, society and nature/existence
3. Strengthening of self-reflection.
4. Development of commitment and courage to act.

2. COURSE TOPICS: The course has 28 lectures and 14 practice sessions in 5 modules:

MODULE 1: COURSE INTRODUCTION - NEED, BASIC GUIDELINES, CONTENT AND PROCESS FOR VALUE EDUCATION

1. Purpose and motivation for the course, recapitulation from Universal Human Values-I
2. Self-Exploration–what is it? - Its content and process; 'Natural Acceptance' and Experiential Validation- as the process for self-exploration
3. Continuous Happiness and Prosperity- A look at basic Human Aspirations
4. Right understanding, Relationship and Physical Facility- the basic requirements for fulfilment of aspirations of every human being with their correct priority
5. Understanding Happiness and Prosperity correctly- A critical appraisal of the current scenario
6. Method to fulfil the above human aspirations: understanding and living in harmony at various levels.

Include practice sessions to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking

MODULE 2: UNDERSTANDING HARMONY IN THE HUMAN BEING - HARMONY IN MYSELF!

7. Understanding human being as a co-existence of the sentient 'I' and the material 'Body'
8. Understanding the needs of Self ('I') and 'Body' - happiness and physical facility
9. Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer)
10. Understanding the characteristics and activities of 'I' and harmony in 'I'
11. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail
12. Programs to ensure Sanyam and Health.

Include practice sessions to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs dealing with disease

MODULE 3: UNDERSTANDING HARMONY IN THE FAMILY AND SOCIETY- HARMONY IN HUMANHUMAN RELATIONSHIP

13. Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship
14. Understanding the meaning of Trust; Difference between intention and competence
15. Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship
16. Understanding the harmony in the society (society being an extension of family): Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals
17. Visualizing a universal harmonious order in society- Undivided Society, Universal Order- from family to world family.

Include practice sessions to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives

MODULE 4: UNDERSTANDING HARMONY IN THE NATURE AND EXISTENCE - WHOLE EXISTENCE AS COEXISTENCE

18. Understanding the harmony in the Nature
19. Interconnectedness and mutual fulfilment among the four orders of naturerecyclability and selfregulation in nature
20. Understanding Existence as Co-existence of mutually interacting units in allpervasive space
21. Holistic perception of harmony at all levels of existence.

Include practice sessions to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology etc.

MODULE 5: IMPLICATIONS OF THE ABOVE HOLISTIC UNDERSTANDING OF HARMONY ON PROFESSIONAL ETHICS

22. Natural acceptance of human values
23. Definitiveness of Ethical Human Conduct
24. Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order
25. Competence in professional ethics: a. Ability to utilize the professional competence for augmenting universal human order b. Ability to identify the scope and characteristics of people friendly and eco-friendly production systems, c. Ability to identify and develop appropriate technologies and management patterns for above production systems.
26. Case studies of typical holistic technologies, management models and production systems
27. Strategy for transition from the present state to Universal Human Order: a. At the level of individual: as socially and ecologically responsible engineers, technologists and managers b. At the level of society: as mutually enriching institutions and organizations
28. Sum up.

Include practice Exercises and Case Studies will be taken up in Practice (tutorial) Sessions eg. To discuss the conduct as an engineer or scientist etc.

3. READINGS:

3.1 Text Book

1. Human Values and Professional Ethics by R R Gaur, R Sangal, G P Bagaria, Excel Books, New Delhi, 2010

3.2 REFERENCE BOOKS

1. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – PanditSunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj or Indian Home Rule - by Mohandas K. Gandhi
11. India Wins Freedom - Maulana Abdul Kalam Azad
12. Vivekananda - Romain Rolland (English)
13. Gandhi - Romain Rolland (English)

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES

(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4596) INTRODUCTION TO MACHINE LEARNING

B.Tech. II Year II SEM: CSE (AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:

1. This course explains machine learning techniques such as decision tree learning, Bayesian learning etc.
2. To understand computational learning theory.
3. To study the pattern comparison techniques.

UNIT - I

Introduction - Well-posed learning problems, designing a learning system, Perspectives and issues in machine learning Concept learning and the general to specific ordering – introduction, a concept learning task, concept learning as search, find-S: finding a maximally specific hypothesis, version spaces and the candidate elimination algorithm, remarks on version spaces

and candidate elimination, inductive bias. Decision Tree Learning – Introduction, decision tree representation, appropriate problems for decision tree learning, the basic decision tree learning algorithm, hypothesis space search in decision tree learning, inductive bias in decision tree learning, issues in decision tree learning.

UNIT - II

Artificial Neural Networks-1

Introduction, neural network representation, appropriate problems for neural network learning, perceptions, multilayer networks and the back-propagation algorithm.

Artificial Neural Networks-2

Remarks on the Back-Propagation algorithm, An illustrative example: face recognition, advanced topics in artificial neural networks. Evaluation Hypotheses – Motivation, estimation hypothesis accuracy, basics of sampling theory, a general approach for deriving confidence intervals, difference in error of two hypotheses, comparing learning algorithms.

UNIT - III

Bayesian learning – Introduction, Bayes theorem, Bayes theorem and concept learning, Maximum Likelihood and least squared error hypotheses, maximum likelihood hypotheses for predicting probabilities, minimum description length principle, Bayes optimal classifier, Gibs algorithm, Naïve Bayes classifier, an example: learning to classify text, Bayesian belief networks, the EM algorithm. Computational learning theory – Introduction, probably learning an approximately correct hypothesis, sample complexity for finite hypothesis space, sample complexity for infinite hypothesis spaces, the mistake bound model of learning. Instance-Based Learning- Introduction, k-nearest neighbour algorithm, locally weighted regression, radial basis functions, case-based reasoning, remarks on lazy and eager learning.

UNIT- IV

Genetic Algorithms – Motivation, Genetic algorithms, an illustrative example, hypothesis space search, genetic programming, models of evolution and learning, parallelizing genetic algorithms. Learning Sets of Rules – Introduction, sequential covering algorithms, learning rule sets: summary, learning First-Order rules, learning sets of First-Order rules: FOIL, Induction as inverted deduction, inverting resolution. Reinforcement Learning – Introduction, the learning task, Q–learning, non-deterministic, rewards and actions, temporal difference learning, generalizing from examples, relationship to dynamic programming.

UNIT - V

Analytical Learning-1

Introduction, learning with perfect domain theories: PROLOG-EBG, remarks on explanation-based learning, explanation-based learning of search control knowledge.

Analytical Learning-2

Using prior knowledge to alter the search objective, using prior knowledge to augment search operators. Combining Inductive and Analytical Learning – Motivation, inductive-analytical approaches to learning, using prior knowledge to initialize the hypothesis.

TEXT BOOKS:

1. Machine Learning – Tom M. Mitchell, - MGH

REFERENCES:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis

Course Outcomes:

1. Understand the concepts of computational intelligence like machine learning
2. Ability to get the skill to apply machine learning techniques to address the real time problems in different areas
3. Understand the Neural Networks and its usage in machine learning application

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES

(UGC-AUTONOMOUS)

COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4518) DATABASE MANAGEMENT SYSTEMS

B.Tech. II Year II-SEM CSE(AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:

This Course provides an emphasis on

1. How to organize, maintain and retrieve information efficiently and effectively from a Database
2. It presents an introduction to database management systems (DBMS) and relational data model.
3. Understanding the different issues involved in the design of a database system
4. Identifying functional dependencies to normalize the relations of database

5. Also the course introduces the concepts of transactions and transaction processing and the issues and techniques relating to concurrency and recovery in multi-user database environments.

Syllabus:

UNIT-I

Database System Applications, database System VS file System, View of Data – Data Abstraction –Instances and Schemas – Data Models, Database Languages, Database Architecture, Database Users and Administrators.

Database design, ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model.

UNIT-II

Introduction to the Relational Model – Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views – Destroying/altering Tables and Views. Relational Algebra – Selection and Projection, Set Operations, Renaming, Joins, Division, Examples of Algebra Queries, Relational calculus – Tuple relational Calculus – Domain relational calculus – Expressive Power of Algebra and calculus.

UNIT-III

Form of Basic SQL Query – Examples of Basic SQL Queries, Introduction to Nested Queries, Correlated Nested Queries, Set – Comparison Operators – Aggregate Operators, NULL values – Comparison using Null values – Logical connectives – AND, OR and NOT – Impact on SQL Constructs, Outer Joins, Disallowing NULL values, Complex Integrity Constraints in SQL Triggers and Active Data bases.Schema refinement – Problems Caused by redundancy, Decomposition – Problem related to decomposition - Reasoning about FDS - FIRST, SECOND, THIRD Normal forms – BCNF –Schema Refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form.

UNIT–IV

Transaction Concept- Transaction State- Implementation of Atomicity and Durability – Concurrent – Executions – Serializability- Recoverability – Implementation of Isolation.

Lock –Based Protocols – Timestamp Based Protocols- Validation- Based Protocols.

Recovery and Atomicity – Log – Based Recovery – Recovery with Concurrent Transactions – Buffer Management – Failure with loss of nonvolatile storage-Advance Recovery systems-Remote Backup systems.

UNIT–V

Data on External Storage, File Organization and Indexing – Clustered Indexes, Primary and Secondary Indexes, Index data Structures – Hash Based Indexing – Tree base Indexing, Comparison of File Organizations.

Intuitions for tree Indexes, Indexed Sequential Access Methods (ISAM) B+ Trees: A Dynamic Index Structure.

Text Books:

1. Raghurama Krishnan, Johannes Gehrke " Data base Management Systems" TATA McGraw-Hill 3rd Edition
2. Silberschatz, Korth "Data base System Concepts" McGraw hill, V Edition.

Reference Books:

1. Peter Rob and Carlos Coronel “Data base Systems design, Implementation, and Management” 7th Edition.
2. Elmasri Navrate “Fundamentals of Database Systems” Pearson Education

Course outcomes:

1: Differentiate database systems from file system by understanding the features of database system and design a ER model for a database system.

2: Develop solutions to a broad range of query and data update problems using relational algebra, relational calculus and SQL.

3: Apply the normalization theory in relational databases for removing anomalies.

4: Analyze the basic issues of transaction processing, concurrency control, deadlock and its recovery schemes

5: Compare database storage and access techniques for file organization, indexing methods and Query Processing.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4525) DATABASE MANAGEMENT SYSTEMS LAB

B.Tech. II Year II-SEM CSE(AI&ML)

L	T	P	C
0	0	4	2

Course Objectives:
This lab enables the students
1. To practice the concepts learnt in the subject DBMS by developing a database for an example company named "Roadway Travels" whose description is as follows.
2. The student is expected to practice the designing, developing and querying a database in the context of example database "Roadway travels". Students are expected to use "Mysql" database.
3. To create a database and query it using SQL, design forms and generate reports.
4. Understand the significance of integrity constraints, referential integrity constraints, triggers, assertions
5. Improve the database design by normalization.

LIST OF EXPERIMENTS:

Roadway Travels
"Roadway Travels" is in business since 1997 with several buses connecting different places in India. Its main office is located in Hyderabad. The company wants to computerize its operations in the following areas:
• Reservations and Ticketing
• Cancellations
Reservations & Cancellation: Reservations are directly handled by booking office. Reservations can be made 30 days in advance and tickets issued to passenger. One Passenger/person can book many tickets (to his/her family).Cancellations are also directly handed at the booking office. In the process of computerization of Roadway Travels you have to design and develop a Database which consists the data of Buses, Passengers, Tickets, and Reservation and cancellation details. You should also develop query's using SQL to retrieve the data from the database.
The above process involves many steps like
1. Analyzing the problem and identifying the Entities and Relationships
2. E-R Model
3. Relational Model
4. Normalization
5. Creating the database
6. Querying.

Students are supposed to work on these steps week wise and finally create a complete "Database System" to Roadway Travels. Examples are given at every experiment for guidance to students.

WEEK 1: E-R Model

Analyze the carefully and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, candidate attributes etc.Identify the primary keys for all the entities. Identify the other keys like keys, partial keys, if any.

Example: Entities:

1. BUS
2. Ticket
3. passenger

Relationships:

1. Reservation
2. Cancellation

PRIMARY KEY ATTRIBUTES:

1. Ticket ID (Ticket Entity)
2. Passport ID (Passenger Entity)
3. Bus_NO (Bus Entity)

Apart from the above mentioned entities you can identify more. The above mentioned are few.
Note: The student is required to submit a document by writing the Entities and Keys to the lab teacher.

WEEK 2: Concept design with E-R Model

Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any). Indicate the type of relationships (total / partial). Try to incorporate generalization, aggregation, specialization etc wherever required.
Note: The student is required to submit a document by drawing the E-R Diagram to the lab teacher.

WEEK 3: Relational Model

Represent all entities (Strong, Weak) in tabular fashion. Represent relationships in a tabular fashion. There are different ways of representing relationships as tables based on the cardinality. Represent attributes as columns in tables or as tables based on the requirement. Different types of Attributes (Composite, Multi-valued, and Derived) have different way of representation.
Example: The passenger tables look as below. This is an example. You can add more attributes based on E-R model. This is not a normalized table.
Passenger

Name	Age	Sex	Address	Ticket_id	Passport ID

Note: The student is required to submit document relationships in a tabular fashion to the lab teacher.

WEEK 4: Normalization

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. For example, when multiple instances of a

given piece of information occur in a table, the possibility exists that these instances will not be kept consistent when the data within the table is updated, leading to a loss of data integrity. A table that is sufficiently normalized is less vulnerable to problems of this kind, because its structure reflects the basic assumptions for when multiple instances of the same information should be represented by a single instance only.For the above table in the First normalization we can remove the multi valued attribute Ticket_id and place it in another table along with the primary key of passenger.
First Normal Form: The above table can be divided into two tables as shown below.
Passenger

Name	Age	Sex	Address	Passport_ID

Passport_ID	Ticket_id

You can do the second and third normal forms if re wired. Any ht)* given Normalized tables are at the end.

WEEK 5: installation of Mysql and practicing DDL commands

Installation of MySql. In this week you will learn Creating databases, How to create tables,altering the database, dropping tables and databases if not required. You will also try truncate, rename commands etc.Example for creation of a normalized "Passenger" table.

```
CREATE TABLE Passenger (
Passport_id INTEGER PRIMARY KEY,
Name VARCHAR (50) Not NULL,
Age Integer Not NULL,
Sex Char,
Address VARCHAR (50) Not NULL);
```

Similarly create all other tables.
Note: Detailed creation of tables is given at the end.

WEEK 6: Practicing DML commands

DML commands are used to for managing data within schema objects. Some examples:

- SELECT - retrieve data from the a database
- INSERT - insert data into a table
- UPDATE - updates existing data within a table
- DELETE - deletes all records from a table, the space for the records remain

Inserting values into "Bus" table:

```
Insert into Bus values (1234,'hyderabad', lirupathi');
Insert into Bus values (2345,1hyderabd,Banglore');
Insert into Bus values (23,'hyderabd','Kolkata');
Insert into Bus values (45,11rupathi,'Banglore');
```

Insert into Bus values (34,1h derab yc11,1Chennar);

Inserting values into "Passenger" table:

Insert into Passenger values (1, 45,'ramesh', 45,'M','abc123');
Insert into Passenger values (2, 78,'geetha', 36,'F','abc124');
Insert into Passenger values (45, 90,'ram', 30,'M',1abc12');
Insert into Passenger values (67, 89,'ravi', 50,'M','abc14');
Insert into Passenger values (56, 22,'seetha', 32,'F','abc55');

Few more Examples of DML commands:

Select * from Bus; (selects all the attributes and Display)
UPDATE BUS SET Bus No = 1 WHERE BUS NO=2;

WEEK 7: Querying

In this week you are going to practice queries (along with subquries) Using queries ANY,ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

Practice the following Queries:

1. Display unique PNR_no of all passengers.
2. Display all the names of male passengers.
3. Display the ticket numbers and names of all the passengers.
4. Find the ticket numbers of the passenger whose name start with and ends with 'h'.
5. Find the names of passengers, whose age is between 30 and 45,

6. Display all the passengers' names beginning with 'A'
7. Display the sorted list of passenger's names

WEEK 8 and WEEK 9: Querying (continued...)

You are going to practice queries using Aggregate functions (COUNT, Sum, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

1. Write a Query to display the Information present in the Passenger and cancellation tables. Hint: Use UNION Operator.
2. Display the number of days in a week on which the 9W01 bus is available.
3. Find number of tickets booked for each PNR_no using GROUP BY CLAUSE. Hint: Use GROUP BY on PNR No.
4. Find the distinct PNR numbers that are present.
5. Find the number of tickets booked by a passenger where the number of seats is greater than 1. Hint: Use GROUP BY, WHERE and HAVING CLAUSES.
6. Find the total number of cancelled seats.

WEEK 10: Triggers

In this week you are going to work on Triggers. Creation of insert trigger, delete trigger, update trigger. Practice triggers using the above database.

Eg: CREATE TRIGGER update check BEFORE UPDATE ON passenger FOR EACH ROW
BEGIN IF NEW.TickentNO > 60 THEN
SET New.Tickent no = Ticket no;
ELSE
SET New.Ticketno:at 0;
END IF;
END;

WEEK 11: Procedures

This session you are going to learn Creation of stored procedure, Execution of procedure and modification of procedure. Practice procedures using the above database.
Eg: CREATE PROCEDURE myProc ()
BEGIN
SELECT COUNT (Tickets) FROM Ticket WHERE age>=40;
End;

WEEK 12: Cursors

In this week you need to do the following: Declare a cursor that defines a result set.Open the cursor to establish the result set. Fetch the data into local variables as needed from the cursor, one row at a time. Close the cursor when done

```
CREATE PROCEDURE myProc (in_customer_id INT)
BEGIN
DECLARE v_id INT;
DECLARE v_name VARCHAR (30);
DECLARE c1 CURSOR FOR SELECT stdld,stdFirstname FROM students WHERE
stdId=in_customer_id;
OPEN c1;
FETCH cl into v_id, v_name;
Close c1;
END;
```

Tables

BUS

Bus No: Varchar: PK (public key)
Source: Varchar
Destination: Varchar

Passenger

PPNO: Varchar(15)) :
PK Name: Varchar(15)
Age int (4)
SexIChar(10) : Male / Female
Address: VarChar(20)

Passenger_Tickets

PPNO: Varchar(15)) :
PK Ticket_No: Numeric (9)

Reservation

PNR_No: Numeric (9) :
FK Journey_date : datetime(8)
No_of_seats : int (8)
Address: Varchar (50)
Contact_No: Numeric (9) --> should not be less than 9 and should not accept any other Character other than Integer
Status: Char (2): Yes / No

Cancellation

PNR_No: Numeric (9): FK

Journey_date : datetime(8)
No_of_seats : int (8)
Address: Varchar (50)
Contact_No: Numeric (9) --> should not be less than 9 and should not accept any other Character other than Integer
Status: Char (2): Yes / No

Ticket

Ticket_No: Numeric (9): PK
Journey date: datetime(8)
Age: int (4)
Sex:Char(10) : Male / Female
Source: Varchar
Destination: Varchar
Dep_time: Varchar

Reference Books:

1. Introduction to SQL, Rick F.Vander Lans, Pearson education.
2. Oracle PL/SQL, B.Rosenzweig and E.Silvestrova, Pearson education
3. Oracle PL/SQL Programming, Steven Feuerstein, SPD.
4. SQL & PL/SQL for Oracle 10g, Black Book, Dr. P. S. Deshpande, Dream Tech.
5. Oracle Database 11g PL/SQL Programming, M. Mc Laughlin, TMH.
6. SQL Fundamentals, J.J. Patrick, Pearson Education.

Course Outcomes:

1. Ability to design and implement a database schema for given problem.
2. Be capable to Design and build a UI application.
3.Apply the normalization techniques for development of application software to realistic problems.
4. Ability to formulate queries using SQL DML/DDL/DCL commands.
5.Ability to design cursors and procedures.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4004) DISCRETE MATHEMATICS

B.Tech. II Year II SEM: CSE(AI&ML)

L	T	P	C
3	0	0	3

Course Objectives

1. To introduce Mathematical Logic, especially First Order Logic to students intending to graduate in Computer Science.
2. To introduce proof techniques such as Mathematical Induction and Contradiction.
3. To Develop an understanding of counting, functions and relations.
4. To make the students familiar with fundamental notions and applicability of algebraic systems
5. To make the students familiar with fundamental notions of graph theory.

UNIT-I

Fundamental Principles of counting: The Rules of Sum and Product, permutations, Combinations: Binomial Theorem

Introduction to Propositional Calculus: Basic Connectives and Truth tables, Logical Equivalence: Laws of Logic, Logical Implication: Rules of Inference.

Predicates: The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems

UNIT-II

Sets: Sets and Subsets, Operations on sets and the Laws of Set Theory, Counting and Venn Diagrams

Relations and Functions: Cartesian Products and Relations, Functions:one-one and Onto Pigeonhole principle, partial ordering relations, POSET, hasse diagrams, Equivalence relations.

UNIT-III

Generating function: Generating Functions, Function of Sequences, Calculating Coefficient of generating function.

Recurrence Relations: The First Order Linear Recurrence Relation, Second Order Linear. Homogenous Recurrence relations with constant coefficients, NonHomogenousRecurence relations.

UNIT-IV

Introduction to graphs: Graphs and their basic properties - degree, path, cycle, Sub graphs, Complements and Graph Isomorphism, Euler trails and circuits, planar graphs, Hamiltonian paths and cycles, Graph Coloring and Chromatic polynomial

Trees: Definitions, Properties, Rooted Trees, Spanning Trees, Minimum Spanning trees : The Algorithms of Kruskal and Prim.

UNIT-V

Algebraic Structures: Algebraic Systems: Examples and General Properties, Semigroups and Monoids, Groups: Definitions and Examples, Subgroups and Homomorphisms.

Lattices: Lattices as Partially Ordered Sets, Lattices as Algebraic Systems.

Text books:

1. Ralph P. Grimaldi, Discrete and Combinatorial Mathematics, An Applied Introduction, 4th edition, Pearson Education, 2003.
2. J.P. Tremblay, R.Manohar, Discrete Mathematical Structures with Applications to Computer Science, TATA McGraw-Hill Edition, 1995.

Suggested Readings:

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, 7th edition, Tata McGraw-Hill, 2005.
2. Joe L.Mott, Abraham Kandel, Theodore P. Baker, Discrete Mathematics for Computer Scientists & mathematicians, 2nd Edition , PHI ,1986.
3. David D.Railey,Kenny A.Hunt, Computational Thinking for the modern problem solving, CRC Press,2014.
4. Uwe Naumann,Olaf Scherk, Combinatorial Scientific Computing,CRC Press,2012.

Course Outcomes:

1. Distinguish between Propositional Logic and Predicate Logic.
2. Apply induction and other proof techniques towards solving recurrences and other problems in elementary algebra.
3. Have an understanding of elementary combinatorics.and distinguish between functions and relations.
4. Deal with problems which may arise in Computer Science and Engineering in near future.
5. Better equipped for examinations involving placement opportunities.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4513) OPERATING SYSYTEMS

B.Tech II Year II-Sem: CSE(AI&ML)

L	T	P	C
3	0	0	3

Course Objectives:
1. To understand main components of OS and their working
2. To study the operations performed by OS as a resource manager
3. To understand the different scheduling policies of OS
4. To understand the different memory management techniques
5.To understand process concurrency, synchronization, input/output, storage and file management.

UNIT-I
Computer System and Operating System Overview: Overview of computer operating systems, operating systems functions, protection and security. Distributed systems, special purpose systems, operating systems structures, systems calls and operating systems generation.**Process Management:** Process concepts, threads, scheduling-criteria algorithms, their evaluation, thread scheduling, case studies UNIX, Linux, Windows.
UNIT-II
Concurrency: Process synchronization, the critical- section problem, Peterson's Solution, synchronization Hardware, semaphores, and classic problems of synchronization, monitors, Synchronization examples, and atomic transactions. Case studies UNIX, Linux, and Windows.**Memory Management:** Swapping, contiguous memory allocation, paging, structure of the page table, segmentation, virtual memory, demand paging, page-Replacement, algorithms, case studies UNIX, Linux, Windows.
UNIT–III
Principles of Deadlock: System model, deadlock characterization, deadlock prevention, detection and avoidance, recovery form deadlock.
File system Interface: The concept of a file, Access Methods, Directory structure,File system mounting, file sharing,protection.
File System implementation: File system structure, file system implementation, directory implementation, directory implementation, allocation methods, free-space management, efficiency and performance, case studies. UNIX, Linux, Windows.
UNIT–IV
Mass-Storage Structure: Mass-storage structure overview of Mass-storage structure, Disk structure, disk attachment disk scheduling, swap-space management, RAID structure, stable-storage implementation, Tertiary storage structure.

UNIT–V

Protection: Protection, Goals of Protection, Principles of Protection, Domain of protection Access Matrix, Implementation of Access Matrix, Access control, Revocation of Access Rights, Capability- Based systems,Language–Based Protection.

Security- The Security problem, program threats, system and network threats cryptography as a security tool, user authentication, implementing security defenses, firewalling to protect systems and networks, computer –security classifications, case studies UNIX, Linux, Windows.

Text Books:

1. Abraham Silberchatz, Peter B. Galvin "Operating System Concepts" Greg Gagne 7th Edition, John Wiley,ISBN:-10:04->1694665
2. Stallings, "Operating Systems – Internal and Design Principles", Fifth Edition–2005, Pearson education/PHI,ISBN:0-13-147954-7

Reference Books:

1. D.M.Dhamdhere "Operating systems- A Concept based Approach" 2nd Edition, TMH,ISBN:13:9780070611948
2. Andrew S Tanenbaum "Modern Operating Systems" 2nd Edition, Pearson/PHI,ISBN:-10:0132392275

Course outcomes:

1. Understand the basics of operating systems like kernel, shell, types and views of operating systems
2. Describe the various CPU scheduling algorithms and remove deadlocks.
3. Explain various memory management techniques and concept of thrashing.
4. Use disk management and disk scheduling algorithms for better utilization of external memory.
5. Recognize file system interface, protection and security mechanisms.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4511) DESIGN AND ANALYSIS OF ALGORITHMS

L	T	P	C
3	0	0	3

B.Tech II Year II-Sem: CSE(AI&ML)

Course Objectives:

This course will develop students' knowledge in/on

1. Techniques for effective problem solving in computing.
2. Analyzing the algorithms and calculating their complexity
3. Designing algorithms using greedy strategy, divide and conquer approach and dynamic programming
4. Backtracking and least cost search
5. Fundamental computability concepts and the complexity of classes P, NP and NP-complete

Syllabus:

UNIT-I

Introduction: Algorithm analysis, Performance analysis, Space complexity and time complexity, Big 'O' notation, Omega notation, Theta notation, Different mathematical approach's for solving Time complexity of Algorithms.

Sets and disjoint set union: Introduction, Union, Find operations.

UNIT-II

Divide and conquer: General method, Binary search, Merge sort, Quick sort, Strassen's matrix multiplication.

Greedy method: General method, Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Optimal storage on tapes, Optimal merge patterns, Single source shortest paths.

UNIT-III

Dynamic programming: General method, Multistage graphs, All pairs shortest paths, Single source shortest paths**.** Optimal binary search trees, String editing, *0/1* Knapsack problem, Reliability design problem, Travelling sales person problem.

UNIT-IV

Back tracking: General method, *N-Queens* problem, Sum of subsets, Graph coloring problem, Hamiltonian cycles.

Branch and bound: General method, Least cost (*LC*) search, the 15-puzzle problem, Control abstractions for *LC search*, *0/1* Knapsack problem, Travelling salesperson problem.

UNIT-V

NP Hard and NP complete problems: Basic concepts - Nondeterministic algorithms, The classes *NP* hard and *NP* complete; *COOK's* Theorem, NP hard graph problems - Clique decision problem, Node cover decision problem, Traveling salesperson decision problem.

Text Books:

1. E.Horowitz, S.Sahni, S.Rajasekaran, ”Fundamentals of Computer Algorithms*”, 2nd Edition, Universities Press,* ISBN: 978-8173716126, 2008

Reference Books:

1. Thomas H.Cormen, Charles E.Leiserson, Ronald L.Rivest, Cliford Stein, “Introduction to Algorithms”, *3rd Edition, Prentice-Hall of India,* ISBN: 978-81-203-4007-7, 2010

2. S.Sridhar, “Design and Analysis of Algorithms”, *Oxford University Press, India,* ISBN -13: 978-0-19-809369-5, ISBN-10: 0-19-809369-1, 2015

Course Outcomes:

Upon completion of this course, students will be able to…

1.Argue the correctness of algorithms using inductive proofs and invariants

2. Analyze the time and space complexity of an algorithms

3.Design algorithms using greedy strategy and dynamic programming

4.Identify algorithm design methodology to solve problems

5.Analyze the classes P, N and NP Complete and be able to prove that a certain problem is NP complete

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4516) OPERATING SYSYTEMS LAB

B.Tech II Year II-Sem: CSE(AI&ML)

L	T	P	C
0	0	3	1.5

Course Objectives:
1. To Know LINUX environment and basic OS commands
2.To use LINUX operating system for study of operating system concepts.
3. To write the code to implement and modify various concepts in operating systems using Linux.
4. To implement different CPU scheduling algorithms, page replacement algorithms and dead lock avoidance algorithm
5.To learn different types of file organization techniques

LIST OF EXPERIMENTS:

1. Simulate the following CPU scheduling algorithms
    a. Round Robin
    b. SJF
    c. FCFS
    d. Priority
2. Simulate all file allocation strategies
    a. Sequential
    b. Indexed
    c. Linked
3. Simulate MVT and MFT
4. Simulate all File Organization Techniques
    a. Single level directory
    b. Two level
    c. Hierarchical
    d. DAG
5. Simulate Bankers Algorithm for Dead Lock Avoidance
6. Simulate Bankers Algorithm for Dead Lock Prevention
7. Simulate all page replacement algorithms
    a. FIF
    b. LRU
    c. LFU etc.
8. Simulate Paging technique of memory management.

Text Books:

1. Abraham Silberchatz, Peter B. Galvin "Operating System Concepts" Greg Gagne 7th Edition, John Wiley,ISBN:-10:04->1694665

2. Stallings, "Operating Systems – Internal and Design Principles", Fifth Edition–2005, Pearson education/PHI,ISBN:0-13-147954-7

Reference Books:

1. D.M.Dhamdhere "Operating systems- A Concept based Approach" 2nd Edition, TMH, ISBN: 13:9780070611948

2.Andrew S Tanenbaum "Modern Operating Systems" 2nd Edition, Pearson/PHI, ISBN:-10:0132392275

Course Outcomes:

1. Upon completing the course the student is capable of explaining the basic structure and functioning of operating system.
2. Student is able to point the problems related to process management and synchronization as well as is able to apply learned methods to solve basic problems.
3. Student is capable of explaining the cause and effect related to deadlocks and is able to analyze them related to common circumstances in operating systems.
4. The student is able to explain the basics of memory management.
5. Student is able to know the structure of the most common file-systems.

JAYAMUKHI INSTITUTE OF TECHNOLOGICAL SCIENCES
(UGC-AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING)

(J4597) MACHINE LEARNING LAB

B.Tech. II Year II SEM: CSE (AI&ML)

L	T	P	C
0	0	3	1.5

Course Objectives: The objective of this lab is to get an overview of the various machine learning techniques and can able to demonstrate them using python.

List of Experiments

1. The probability that it is Friday and that a student is absent is 3 %. Since there are 5 school days in a week, the probability that it is Friday is 20 %. What is the probability that a student is absent given that today is Friday? Apply Baye's rule in python to get the result. (Ans: 15%)

2. Extract the data from database using python

3. Implement k-nearest neighbours classification using python

4. Given the following data, which specify classifications for nine combinations of VAR1 and VAR2 predict a classification for a case where VAR1=0.906 and VAR2=0.606, using the result of kmeans clustering with 3 means (i.e., 3 centroids)

VAR1	VAR2	CLASS
1.713	1.586	0
0.180	1.786	1
0.353	1.240	1
0.940	1.566	0
1.486	0.759	1
1.266	1.106	0
1.540	0.419	1
0.459	1.799	1
0.773	0.186	1

5. The following training examples map descriptions of individuals onto high, medium and low credit-worthiness.

medium skiing design single twenties no -> highRisk
high golf trading married forties yes -> lowRisk
low speedway transport married thirties yes -> medRisk
medium football banking single thirties yes -> lowRisk
high flying media married fifties yes -> highRisk
low football security single twenties no -> medRisk
medium golf media single thirties yes -> medRisk

medium golf transport married forties yes -> lowRisk
high skiing banking single thirties yes -> highRisk
low golf unemployed married forties yes -> highRisk
Input attributes are (from left to right) income, recreation, job, status, age-group, home-owner. Find the unconditional probability of `golf' and the conditional probability of `single' given `medRisk' in the dataset?

6. Implement linear regression using python.
7. Implement Naïve Bayes theorem to classify the English text
8. Implement an algorithm to demonstrate the significance of genetic algorithm
9. Implement the finite words classification system using Back-propagation algorithm

Course Outcomes:
After the completion of the course the student can able to:
1. understand complexity of Machine Learning algorithms and their limitations;
2. understand modern notions in data analysis-oriented computing;
3. be capable of confidently applying common Machine Learning algorithms in practice and implementing their own;
4. Be capable of performing experiments in Machine Learning using real-world data