GMR Institute of Technology
An Autonomous Institute Affiliated to JNTU-GV

GMRIT
Training Tomorrow's Engineers Today

1.1.3 (A) Syllabus copy of the courses highlighting Focus on Employability/Entrepreneurship/ Skill development along with their course outcomes

DEPARTMENT OF CSE-AI & ML

INDEX

GMR Institute of Technology
An Autonomous Institute Affiliated to JNTU-GV

GMRIT Training Tomorrow's Engineers Today

Department of CSE-AI&ML

Minimum Credits to be earned: 160 (for Regular Students)
123 (for Lateral Entry Students)

Fifth Semester							
1	21IT405	Web Technologies (Integrated)	3,5,PSO1	3	-	2	4
2	21ML502	Neural Networks	1,2,4,5,12	3	-	-	3
3	21DS503	Data Analytics &Visualization Techniques (Integrated)	1,4,PSO1	3	-	2	4
4	21ML504	Computer Networks	1, 2,3	3	-	-	3
5		Elective I (Professional Elective)		3	-	-	3
6		Elective II (Open Elective I)		3	-	-	3
7	21ML507	Neural Networks Lab	4,5,8	-	-	3	1.5
8	21TPX01	Term Paper	1,4,10,12	-	-	3	1.5
9	21ESX02	Employability Skills II	1,2,5,8,10,12	0	-	2	-
10	21HSX12	CC & EC Activities II	6,7, 9,10	-	-	1	-
11	21SIX01	Summer Internship I	1,2,8,10,12	-	-	-	1
			Total	**18**	**-**	**12**	**24**
Sixth Semester							
1	21ML601	Deep Learning Techniques	1, 2,4,12,PSO1	3	-	-	3
2	21ML602	Automata Theory and Language Processors	1,2,3	3	-	-	3
3	21CS603	Software Engineering	4,5,8,11, PSO1	3	-	-	3
4		Elective III (Professional Elective)		3	-	2	4
5		Elective IV (Open Elective II)		3	-	-	3
6	21ML606	Deep Learning Techniques Lab	4, 5,08	-	-	3	1.5
7	21MPX01	Mini Project	All POs & PSOs	-	-	3	1.5
8	21ESX02	Employability Skills II	1,2,3,5,6,8,10,12	0	-	2	2

9	21HSX12	CC & EC Activities II	6,7,9,10	-	-	1	1
10	21ATX01	Environmental Studies	1,7	-	-	-	-
11	21ATX02	Professional Ethics and Human Values	----	-	-	-	-
12	21ATX---	Audit Course	**12**	-	-	-	-
			Total	**15**	**-**	**8**	**22**
Seventh Semester– AI & ML							
1		Elective V (Professional Elective)		3	-	-	3
2		Elective VI (Professional Elective)		3	-	-	3
3		Elective VII (Open Elective III)		3	-	-	3
4	21SIX02	Summer Internship II	1,2,5,6,10,12	-	-	-	1
5	21PWX01	Project	All POs& PSOs	-	-	16	8
			Total	**9**	**-**	**16**	**18**
Eighth Semester-AI & ML							
1		Elective VIII (Professional Elective)		-	-	-	3
2		Elective IX (Open Elective IV)		-	-	-	3
3	21FIX01	Full Semester Internship (FSI)	1,2,5,8,9,10, PSO1,PSO2	-	-	-	8
			Total	**6**	**-**	**-**	**14**

List of Electives

Language Electives							
No.	**Course Code**	**Course**	**POs**	**Contact Hours**			
				L	**T**	**P**	**C**
1	21HSX03	Advanced Communicative English	10,12	2	-	-	2
2	21HSX04	Communicative German		2	-	-	2
3	21HSX05	Communicative French		2	-	-	2
4	21HSX06	Communicative Japanese		2	-	-	2
5	21HSX07	Communicative Spanish		2	-	-	2
6	21HSX08	Communicative Korean		2	-	-	2
7	21HSX09	Communicative Hindi		2	-	-	2

Elective I							
Career Path I, II, III							
1	**21MLC11**	**Computer Vision & Pattern Recognition**	1,3,PSO1,PSO2	3	-	-	3

2	**21CSC21**	**Web Programming Languages (Full Stack Development)**	1,2,7,12	3	-	-	3
3	**21MLC31**	**Fundamentals of Cloud Computing**	2,6,7,8	3	-	-	3
4	21CS004	Principles of Programming Languages	1, 2, 3, 4	3	-	-	3
5	21CS005	Mobile Computing	3, 8	3	-	-	3
6	21CS006	Distributed Operating Systems	1,2	3	-	-	3
7		MOOCs/Honors		3	-	-	3
Elective II: Open Elective I							
1	21CE001	Disaster Management	2, 7	3	-	-	3
2	21EE001	Electrical Installation and Safety Measures	2,3,6,8	3	-	-	3
3	21DS001	Fundamentals of Data Science	1,2	3	-	-	3
4	21ME001	Fundamentals of Optimization Techniques	1, 2	3	-	-	3
5	21EC001	Sensors for Engineering Applications	1, 2	3	-	-	3
6	21CS001	Fundamentals of Artificial Intelligence	1, 2, 3	3	-	-	3
7	21IT001	Fundamentals of Multimedia	1, 5, 7	3	-	-	3
Elective III							
Career Path I, II, III							
1	**21MLC12**	**Machine Learning for Business Intelligence**	2,3,PSO1,PSO 2	3	-	2	4
2	**21CSC22**	**Web Application Developments Framework (Full Stack Development)**	1, 3,4	3	-	2	4
3	**21MLC32**	**Cloud Services using AWS**		3	-	2	4
4	21CS007	Cloud Computing Essentials	2,5,6,7,8	3	-	2	4
5	21CS008	Cryptography and Network Security	3, 6,8	3	-	2	4
Elective IV: Open Elective II							
1	21CE002	Air Pollution and Environmental Impact Assessment	6, 7,12	3	-	-	3
2	21EE002	Renewable Energy Sources	2, 7	3	-	-	3
3	21ME002	Principles of Entrepreneurship	1,11	3	-	-	3
4	21EC002	Electronics for Agriculture	1, 2	3	-	-	3
5	21CS002	Fundamentals of Machine Learning	2, 3	3	-	-	3
6	21CH002	Industrial Safety and Hazard Management	2, 6, 7,8	3	-	-	3
7	21IT002	Fundamentals of Cloud Computing	1, 7	3	-	-	3
8	21BS002	Advanced Numerical Techniques	1, 2	3	-	-	3
9	21BS003	Functional Materials and Applications	1, 2	3	-	-	3
Elective V							
Career Path I, II, III							
1	**21MLC13**	**Conversational AI**	1,2,4,12,PSO1	3	-	-	3
2	**21CSC23**	**Web Application Databases (Full Stack Development)**	2,3	3	-	-	3
3	**21MLC33**	**Cloud Security Essentials**	2,3	3	-	-	3
4	21IT010	Social Network Analysis	2, 4, 5	3	-	-	3
5	21ML001	Human Computer Interaction	2, 3	3	-	-	3
6	21CS012	Wireless Adhoc Networks	2, 3	3	-	-	3
7		MOOCs/Honors		3	-	-	3

Elective VI							
1	21ML002	Large Language Models	2,3,5,6	3	-	-	3
2	21CS015	Software Project Management	3,6	3	-	-	3
3	21ML003	Reinforcement Learning	6,7	3	-	-	3
4		MOOCs/Honors		3	-	-	3
Elective VII: Open Elective III							
1	21CE003	Solid Waste Management	2,3,12	3	-	-	3
2	21EE003	Fundamentals of Electrical Vehicle Technology	2,3,12	3	-	-	3
3	21ME003	Industrial Engineering and Management	1,11	3	-	-	3
4	21EC003	Interfacing and Programming with Arduino	1,2	3	-	-	3
5	21CS003	Data Science for Engineering Applications	2,3,4	3	-	-	3
6	21CH003	Industrial Ecology for Sustainable Development	2,6,7	3	-	-	3
7	21IT003	Fundamentals of Mobile Computing	1,7	3	-	-	3
8	21BS004	Advanced Materials of Renewable Energy	1,7	3	-	-	3
9	21BS005	Applied Linear Algebra for Engineers	1,12	3	-	-	3

Elective VIII: Professional Elective							
1	21CS018	Information Retrieval Systems	1,2,3,4	-	-	-	3
2	21CS019	Fundamentals of Devops	1,3, 5,8,10	-	-	-	3
3	21DS003	Cyber Security	1,3,4,5,PSO1				3
4		MOOCs/Honors		-	-	-	3

Elective IX: Open Elective IV							
1	21CE019	Green Buildings	2,3,4,5,7	-	-	-	3
2	21EE017	Sustainable Energy	1,2,12	-	-	-	3
3	21ME004	Total Quality Management	1,11	-	-	-	3
4	21EC011	Communication Technologies	1,2	-	-	-	3
5	21CS020	Applications of Artificial Intelligence	2,3,6,7	-	-	-	3
6	21CH016	Green Technologies	1,6,7	-	-	-	3
7	21IT015	Human Computer Interaction	1,7	-	-	-	3
8	21BS006	Handling of Industrial Waste and Wastewater	1,7	-	-	-	3

Audit Course							
1	21AT001	Communication Etiquette in Workplaces	-	-	-	-	-
2	21AT002	Contemporary India: Economy, Policy andSociety	-	-	-	-	-
3	21AT003	Design The Thinking	-	-	-	-	-
4	21AT004	Ethics and Integrity	-	-	-	-	-
5	21AT005	Indian Heritage and Culture	-	-	-	-	-
6	21AT007	Intellectual Property Rights and Patents	-	-	-	-	-
7	21AT008	Introduction to Journalism	-	-	-	-	-
8	21AT009	Mass Media Communication	-	-	-	-	-
9	21AT010	Science, Technology and Development	-	-	-	-	-
10	21AT011	Social Responsibility	-	-	-	-	-
11	21AT012	The Art of Photography and Film Making	-	-	-	-	-
12	21AT013	Gender Equality for Sustainability	-	-	-	-	-
13	21AT014	Women in Leadership	-	-	-	-	-
14	21AT015	Introduction to Research Methodology	-	-	-	-	-

15	21AT016	Climate Change and Circular Economy	-	-	-	-	-

B. Tech. (Honors)							
Domain I (Data Engineering)							
1	21CSH11	Advanced Data Structures	2,3,4	4	-	-	4
2	21CSH12	Advanced Databases	2,3,4	4	-	-	4
3	21CSH13	Programming, Data Structures and Algorithms Using Python	2,3,4,5	4	-	-	4
4	21CSH14	Bioinformatics	2,3	4	-	-	4
Domain II (Modern Software Engineering)							
1	21CSH21	DevOps	1,3,5,8,10	4	-	-	4
2	21CSH22	Design Patterns	2,3	4	-	-	4
3	21CSH23	Advanced Software Engineering	1,3,4, PSO1	4	-	-	4
4	21CSH24	Robotic Process Automation	3,5, 8, PSO2	4	-	-	4
Domain III (Security)							
1	21CSH31	Fundamentals of Systems Security	1,2	4	-	-	4
2	21CSH32	Python Programming for Security	2,3,4	4	-	-	4
3	21CSH33	Management of Information Security	3,6,7	4	-	-	4
4	21CSH34	Computer Forensics	2,3	4	-	-	4
Domain IV (User Interface Design)							
1	21CSH41	Computer Graphics	1,2,3,4	4	-	-	4
2	21CSH42	Multimedia Systems	3,4	4	-	-	4
3	21CSH43	Human Computer Interaction	2,3	4	-	-	4
4	21CSH44	Mobile Programming	3,4	4	-	-	4

B. Tech. (Minors)							
Energy Science & Technology							
1	21CHM11	Foundation of Energy Science and Technology	1,2,3,5,7,12	4	-	-	4
2	21CHM12	Energy Generation from Waste	1,2,3,4,5	4	-	-	4
3	21CHM13	Energy Storage Systems	1,2,3,6,7	4	-	-	4
4	21CHM14	Hydrogen Energy and Fuel Cells	1,2,3,7	4	-	-	4
Nano Science & Technology							
1	21CHM21	Introduction and Characterization of Nano Materials	1,2,3,7	4	-	-	4
2	21CHM22	Carbon Nanostructures and Applications	1,3,4,5	4	-	-	4
3	21CHM23	Energy, Environment & Biomedical Nanotechnology	1,2,3,7	4	-	-	4
4	21CHM24	Industrial Applications of Nano Technology	2,3,5,,7	4	-	-	4
Environmental Engineering							
1	21CEM11	Watershed Management	6,7	4	-	-	4
2	21CEM12	Industrial Pollution Control and Engineering	3,6,7,12	4	-	-	4
3	21CEM13	Solid and Hazardous Waste Management	1,3,6,7	4	-	-	4
4	21CEM14	Ecology and Environmental Assessment	1,3,6,7	4	-	-	4
Artificial Intelligence & Machine Learning							
1	21CSM11	Fundamentals of AI & Machine Learning	1,12	4	-	-	4

2	21CSM12	Feature Engineering for Machine Learning	1,2,3	4	-	-	4
3	21CSM13	Exploratory Data Analytics	1,4	4	-	-	4
4	21CSM14	Foundations of Deep Learning	1,2, 4	4	-	-	4
Cyber Security							
1	21CSM21	Fundamentals of Security	1,2	4	-	-	4
2	21CSM22	Management of Information Security	3,6,7	4	-	-	4
3	21CSM23	Cyber Security	1,3,4	4	-	-	4
4	21CSM24	Fundamentals of Cloud Security	2,3	4	-	-	4
Data Science & Analytics							
1	21CSM31	Data Cleaning	2,3,4	4	-	-	4
2	21CSM32	Data Engineering	1,2,3,4	4	-	-	4
3	21CSM33	Text Analytics	1,2,4	4	-	-	4
4	21CSM34	Social Network and Semantic Analysis	2, 4	4	-	-	4
Computer Systems Programming							
1	21CSM41	Programming Fundamentals	1,2,3	4	-	-	4
2	21CSM41	Data Structures & Algorithms	1,2,3,4	4	-	-	4
3	21CSM41	Fundamentals of Databases	1,4	4	-	-	4
4	21CSM41	Fundamentals of Computer Networks & Operating Systems	1,2,3	4	-	-	4
Digital IC Design							
1	21ECM11	Fundamentals of VLSI Design	1,2,3	4	-	-	4
2	21ECM12	Digital Design using HDL	1,2,3	4	-	-	4
3	21ECM13	FPGA Technology	1,2	4	-	-	4
4	21ECM14	Analog and Mixed Signal Design	1,2	4	-	-	4
Industrial Automation							
1	21ECM21	Microcontrollers and Interfacing	1,2,3	4	-	-	4
2	21ECM22	Sensors and Data Acquisition System	1,2	4	-	-	4
3	21ECM23	Fundamentals of Labview	1,2	4	-	-	4
4	21ECM24	Medical Robotics	1,2,3	4	-	-	4
Communications and Networking							
1	21ECM31	Principles of Communications	1,2	4	-	-	4
2	21ECM32	Coding Theory and Practice	1,2	4	-	-	4
3	21ECM33	Ad-hoc and Wireless Sensor Networks	1,2,3	4	-	-	4
4	21ECM34	Fundamentals of Multimedia Networking	1,2,3	4	-	-	4
Avionics							
1	21ECM41	Principles of Aerodynamics	1,2	4	-	-	4

2	21ECM42	Aircraft Electrical Systems	1,2	4	-	-	4
3	21ECM43	Aircraft Instrument Systems	1,2	4	-	-	4
4	21ECM44	Aircraft Communication and Navigational Systems	1,2	4	-	-	4
Geographic Information System							
1	21ECM51	Sensors and Sensing Technology	1,2	4	-	-	4
2	21ECM52	Geographic Information Systems	1,2	4	-	-	4
3	21ECM53	Digital Image Processing	1,2	4	-	-	4
4	21ECM54	Lidar Systems	1,2	4	-	-	4
Electric Vehicles Technology							
1	21EEM11	Introduction to Electric Vehicles Technologies	2,3	4	-	-	4
2	21EEM12	Electrical Drives and Controllers for Electric Vehicles	2,3	4	-	-	4
3	21EEM13	Charging Technology in Electric Vehicles	2,3	4	-	-	4
4	21EEM14	Computer Vision in Electric Vehicles	2,3	4	-	-	4
Electric Vehicles Technology							
1	21EEM21	Fundamentals of Smart City	2,3	4	-	-	4
2	21EEM22	Smart City Infrastructure	2,3	4	-	-	4
3	21EEM23	Computational Methods for Smart City Management	2,3	4	-	-	4
4	21EEM24	Communication Technologies and Mobility for Smart City	2,3	4	-	-	4
Electric Vehicles Technology							
1	21EEM31	Modelling and Simulations of Industrial Applications	2,3	4	-	-	4
2	21EEM32	Industrial Sensors and Actuators	2,3	4	-	-	4
3	21EEM33	Programmable Logic Controllers	2,3	4	-	-	4
4	21EEM34	Control Design for Industrial Applications	2,3	4	-	-	4
Cloud Application Development							
1	21ITM11	Introduction to Cloud Computing	6, 7, 12	4	-	-	4
2	21ITM12	Introduction to Web Development with HTML, CSS, JavaScript	1, 2, 3, 9, 12	4	-	-	4
3	21ITM13	Developing Cloud Native Applications	5, 8, 10	4	-	-	4
4	21ITM14	Developing Cloud Apps with Node.js and React	5, 8, 10	4	-	-	4
Robotics and Automation							

1	21MEM11	Introduction to Robotics	1,2,3	4	-	-	4
2	21MEM12	Drives and Sensors	1,2,3,4	4	-	-	4
3	21MEM13	Control Systems for Robotics	1,2,3,4	4	-	-	4
4	21MEM14	Machine Learning for Robotics	2,5	4	-	-	4
Industrial Systems Engineering							
1	21MEM21	Industrial Management	1,10,11,12	4	-	-	4
2	21MEM22	Fundamentals of Operations Research	1,2,3,5	4	-	-	4
3	21MEM23	Enterprise Resource Planning	1,2,3,5,11,12	4	-	-	4
4	21MEM24	Production Planning and Control	1,2,3,5,11,12	4	-	-	4

5th Semester

21IT405 Web Technologies

3 0 2 4

Course Outcomes

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

1. Illustrate client-side scripting tools to create and modify web pages.
2. Infer the knowledge of web programming to develop and maintain web pages.
3. Develop portable and dynamic web pages.
4. Demonstrate server-side scripting languages to develop business logic.
5. Make use of database connectivity to communicate database server from web server.
6. Test for client-server web application development in real time internet domains.

CO–PO Mapping

COs O	**PO$_3$ O**	**PO$_4$ O**	**PO$_5$**	**PSO$_1$**
1	3	3	2	3
2	3	3	2	3
3	2	2	3	3
4	2	2	3	2
5	3	3	3	3
6	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Syllabus

Unit- I **12+8 Hours**

HTML Tags

Introduction, Links, Lists, Tables, Forms, Canvas, Semantics, Plug-ins.**CSS:** Introduction, CSS Properties, Selectors, Combinators, Controlling Fonts, Forms, Pseudo classes, Transitions, Animations (data-aos), 2D and 3D Transforms, Navigation Bar, Layouts,

Java Script: Introduction, Functions, Events, Validations, Objects, Document Object Model (DOM), Division Replacement, Browser Object Model (BOM)

CSS Tooltips, Font Combinations.

Practical Components

1. Design the following static web pages required for Online Examination System web site. a) Home Page
   b) Login Page: a. Student Login Page b. Admin Login Page
   c) Registration page
   d) Test Page
   e) Results Page.
2. For Specific pages of Online Examination System like Test page and Result page Apply different font styles, font families, font colors, animations and other formatting styles to the above static web pages.
3. Validate login page and registration page using Java Script.
4. Create an HTML page with any required JavaScript that takes a number from text field in the range of 0 to 999 and shows it in words. It should not accept four and above digits, alphabets and special characters.

Unit- II **12+8 Hours**

Bootstrap

Responsive Design, Layouts, Grids, Media Queries, **Components-** Forms, Drop Downs, Cards, Input Groups, Collapse, List Groups, Navbar, Popovers, Flex Box. **Utilities-** Borders, Positions, Scroll spy, Visibility, CSS Components, SVG, Java Script Components. **JSON:** Introduction, Parsing, Objects, JSON arrays, Dynamic HTML Data. **jQuery:** Selectors, Filters, Form Plug-ins, Auto Validations

Bootstrap icons, Jumbotron.

Practical Components

1. Design an interactive area using HTML and CSS. Use JavaScript to implement the color change effect during mouseover and reversion during mouse out.
2. **Inner Function with Constant Addition**: Design a function "createAdderWithConstant" that takes a constant as a parameter and returns an inner function. The inner function, when given a number, adds the constant to it and returns the result. Use the inner function to add the same constant to different numbers.
3. Make the static pages Responsive and attractive using Bootstrap components (***Mobile View)
4. Validate login page and registration page using JQuery**.**

Unit- III **12+8 Hours**

PHP

Introduction, Modal View Architecture (MVC), Creating PHP script, Running PHP script, Introduction to PHP, Arrays, Functions, sending parameters in URL, working with Forms, **working with Database:** Running SQL Queries, connecting with Databases, Uploading Files, PHP Sessions, Redirecting, Routing, Connecting PHP with JavaScript and jQuery, Authentication.

PHP Cookies, PHP Exception Handling, Introduction to PHP Laravel

Practical Components

1. Write a PHP to connect to the database, Insert the details of the student who registered through Online Examination System student login page including photograph.
2. Develop and demonstrate PHP Script for the following problems:
   a. Write a PHP Script to find out the Sum of the Individual Digits.
   b. Write a PHP Script to check whether the given number is Palindrome or not
3. Write a PHP program to insert the questions and respective answers into the database through admin page, sending mails to registered students through admin.
4. Write a PHP program, assuming four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Create a Cookie and add these four user ID‟s and passwords to this Cookie. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.
   a. If he is a valid user (i.e., user-name and password match) you should welcome him by name (username) else you should display "You are not an authenticated user"

Unit- IV **12+8 Hours**

Flask

Introduction to Flask, Virtual Environment, features of flask, url building, routing, Templates and Jinja Code, Rendering Templates, Static files, Building Forms, Sending Form data to Templates, Template Inheritance (header, footer, etc), Session Tracking, connecting database (SQLite), Retrieving database values to templates, File uploading, Sending Mails, Deployment of website.

wsgi file, csrf token, http methods.

Practical Components

1. Write a Flask code to connect to the database, and authenticate Login pages with jQuery and start a Session for Student, Change password for Student
2. After user login, display Questions from the database into test page in shuffled manner using flask and store in database.
3. Write a Flask code to do the following
   a) Evaluate the answers of the test which is given by the student in the Test page.
   b) Calculate the total score of the student and store it into the database and display score in dashboard of student.
4. Using flask retrieve results in a responsive table format of all students with filtering. Deploy the application developed in real time environment

Total: 48+32 Hours

Textbook(s):

1. Programming the World Wide Web, 8th edition Robert W. Sebesta, pearson.
2. Bootstrap: Responsive Web Development, 1st Edition, jake spurlock foreword by dave winer, O'Reilly publications.

Reference Book(s):

1. Web programming with HTML, XHTML and CSS, 2e, Jon Duckett, Wiley India
2. Web programming Bai, Michael Ekedahl, CENAGE Learning, India edition.
3. An Introduction to Web Design + Programming, Paul S.Wang, India Edition
4. Flask Web Development 2e Miquel Grinberg O'Reilly publications.

21ML502 Neural Networks

3 0 0 3

Course Outcomes

1. Illustrate the knowledge of the Neural Networks and their roles in building intelligent machines
2. Identify and apply supervised learning methods to different neural networks models for pattern classification and regression problems.
3. Demonstrate different data representations for Neural Networks.
4. Evaluate and apply unsupervised learning methods to different neural networks models for storing and pattern classification.
5. Exemplify different Deep unsupervised models.
6. Understand the basic concepts of Deep Learning and its Application.

COs–POs Mapping

COs	**PO1**	**PO2**	**PO4**	**PO5**	**PO12**
1	3	2	2	1	1
2	3	2	2	1	1
3	3	3	3	1	1
4	3	3	3	1	2
5	3	3	3	3	2
6	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Neural Networks

Introduction to Neural Networks, Benefits of Neural Networks, Human Brain, Evolution of Neural Networks Artificial Neural Networks: Introduction to Artificial Neural Networks, Neural Networks Terminologies, Basic Models of Neural Networks , McCulloch-Pitts Neuron and Hebb Network.

Basics of Artificial Intelligence and Knowledge Representation.

Unit II **12 Hours**

Supervised Learning Network: Perceptron Networks, Multilayer Perceptron Networks, Back Propagation Network. Cross Validation – Model Selection, Early Stopping Method of Training, Variance of Cross Validation.

Associative Memory Networks: Auto-associative Memory Network, Hetero auto-associative Memory Network , Bidirectional Associative Memory (BAM)

Graphical Neural Network and Batch Normalization, Dropout

Unit III **12 Hours**

Unsupervised Learning Network

Introduction -Kohonen Self-Organizing Feature Maps - Learning Vector Quantization - Counter Propagation Networks - Adaptive Resonance Theory Networks.

Deep unsupervised models: Auto-Encoder in NN and variational Auto-encoders.

Tensor flow, data augmentation and Approximation of Functions.

Unit IV **12 Hours**

Deep Learning: Introduction to deep learning, Convolutional Neural Network, Recurrent Neural Network, Applications of Deep Learning, Image segmentation and object detection.

End-to-End deep learning and Transfer Learning

Total: 48 Hours

Textbook (s)

1. Simon Haykin, "Neural Networks - A comprehensive foundation", Pearson Education, 2003.
2. Laurene Fausett, "Fundamentals of Neural Networks" , Pearson Education, 2004.
3. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation,2015. **Reference (s)**

1. Satish Kumar,"Neural Networks: A Classroom Approach" Tata McGraw Hill Education, 2004.
2. Simon Haykin, " Neural networks and Learning Machines", Prentice Hall, 2008
3. Josh Patterson and Adam Gibson "Deep Learning A Practitioner's Approach" O'Reilly Media, Inc. 2017

21DS503 Data Analytics and Visualization Techniques

3 0 2 4

Course Outcomes

1. Understand the Data Analysis Fundamentals
2. Apply data transformation techniques for preprocessing and feature engineering in datasets.
3. Analyze and visualize data using descriptive statistics, correlation measures, and advanced plotting techniques.
4. Apply data transformation, statistics, and plots like violin and heatmaps for data analysis.
5. Analyze and visualize associations using scatterplots, correlograms, and treemaps.
6. Analyze and visualize individual and multivariate time series data.

COs–POs Mapping

COs	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO12**	**PSO1**
1	2	3	-	-	2	1	3
2		2	-	2	2	-	3
3	2	3	-	1	3	-	3
4		2	-	1	3	-	3
5		2	3	-	2	2	3
6	2	3	3	-	2	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I 12+8 Hours

Data Analysis Fundamentals

Understanding data science, significance of EDA, steps in EDA. types of analysis (univariate, bivariate, multivariate). Making sense of data: Numerical data-Discrete &continuous data, categorical data, Measurment scales-Nominal, Ordinal, Interval, Ratio. Comparing EDA with classical and Bayesian analysis, getting started with EDA: Numpy, Pandas,Scipy, and Matplotlib.

Grouping of data: Groupby mechanics, rearranging, reshaping data structures, data aggregation methods, and cross-tabulation methods.

Objectives of Exploratory data Analysis, The applications of EDA

Practical Components
1: Explore Scipy Library
2: Explore Stat Library
3: Experiment with all group by functionalities & cross tabulation methods 4: Analyzing and Visualizing Real-World Data.

Unit II 12+8 Hours

Data transformation and descriptive statistics

Transformation techniques- ı, replacing values, handling missing data, renaming indexes, outlier detection and filtering. Permutation and randc axis variables, string manipulations.

Correlation and Inferences: covariance, Pearson’s Correlation, Spearman’s Rank Correlation.

Descriptive statistics: central tendenacy, measure of dispersion (standard deviation, variance, skewness, kurtosis, percentiles, quartiles)
Violin Plots, Inter Quartile Range, Discretization & binning, heatmaps

Practical Components
5: Experiment with various data pre-processing & Transformation techniques
6:Experiment and Analyse the correlations among attributes. 7: Experiment with the statistical measures 8: Outlier Detection and Handling.

Unit III 12+8 Hours

Visualizing Distributions

Visualizing Amounts-Bar Plots, Grouped and Stacked Bars, Dot Plots and Heatmaps, Visualizing Distributions: Histograms and Density Plots- Visualizing a Single Distribution, Visualizing Multiple Distributions at the Same Time, Visualizing Distributions: Empirical Cumulative Distribution Functions and Q-Q Plots-Empirical Cumulative Distribution Functions, Highly Skewed Distributions, Quantile Plots, Visualizing Many Distributions at OnceVisualizing Distributions Along the Vertical Axis, Visualizing Distributions Along the Horizontal Axis

Kernel Density Estimation (KDE), Geospatial Distributions, Interactive Visualization

***Practical* Components**
Download the House Pricing dataset from Kaggle
9: Use different Color scales on the Rainfall Prediction dataset
10: Create different Bar plots for variables in any dataset
11: Visualization of different Distributions in various ways
12: Visualizing Empirical Cumulative Distribution Functions (ECDF) for Real-world Data.

Unit IV 12+8 Hours

Visualizing Associations & Time Series

Stacked Densities, Visualizing Proportions Separately as Parts of the Total ,Visualizing Nested Proportions- Nested Proportions Gone Wrong, Mosaic Plots and Treemaps, Nested Pies ,Parallel Sets. Visualizing Associations Among Two or More Quantitative Variables-Scatterplots, Correlograms, Dimension Reduction, Paired Data. Fundamentals on Time Series Data, Visualizing Time Series and Other Functions of an Independent Variable-Individual Time Series , Multiple Time Series and Dose–Response Curves, Time Series of Two or More Response Variables

Multidimensional Scaling, MANOVA vs ANOVA

Practical Components
Experiment 13: Visualization through Pie charts, Stacked Bars & Stacked Densities
Experiment 14: Show an example of Skewed data and Experiment with different skewness removal techniques
Experiment 15: For a sales dataset do a Time Series Visualization
Experiment 16: Geospatial Visualization for Regional Sales Analysis

Total: 45+32 Hours

Textbook (s)

1. Daniel J. Denis: Univariate, Bivariate, and Multivariate Statistics Using R: Quantitative Tools for Data Analysis and Data Science, Wiley,2020
2. Mukhiya Suresh Kumar Mukhiya, Ahmed Usman Ahmed: Hands-On Exploratory Data Analysis with Python: Perform EDA techniques to understand, summarize, and investigate your data, Packt, 2020
3. Claus Wilke, "Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures", 1st edition, O'Reilly Media Inc, 2019.
4. Downey, Allen. Think stats: exploratory data analysis. " O'Reilly Media, Inc.", 2014.
5. Neil H. Spencer: Essentials of Multivariate Data Analysis, CRC Press,2014 **Reference (s)**

1. Wes McKinney : Python for Data Analysis 2nd Edition,Wiley,2013
2. Glenn J. Myatt, Wayne P. Johnson: Making Sense of Data I: A Practical Guide to Exploratory Data Analysis and Data Mining, 2nd Edition,Wiley,2014
3. 1 Tony Fischetti, Brett Lantz, R: Data Analysis and Visualization,O'Reilly ,2016 2 OssamaEmbarak, Data Analysis and Visualization Using Python: Analyze Data to Create Visualizations for BI Systems,Apress, 2018
4. Wendy L. MartinezAngel R. MartinezJeffrey L. Solka: Exploratory Data Analysis with MATLAB, 2nd Edition,CRC Press,2011
5. Radhika Datar, Harish Garg : Hands-On Exploratory Data Analysis with R , Packt,2020
6. Joseph F Hair, Barry J. Babin, Rolph E. Anderson, William C. Black: Multivariate Data Analysis Cengage, 2018

21ML504 Computer Networks

3 0 0 3

Course Outcomes

1. Understand the basic network infrastructure to learn the overall functions of networking systems
2. Classify various wired and wireless transmission media for data communication networks
3. Apply knowledge of different techniques of error detection and correction to detect and solve error bit during data transmission.
4. Compare various routing algorithm and select an appropriate one for a routing design.
5. Design a proper routing for IP networks.
6. Understand the internal functionalities of main protocols such as HTTP, FTP, SMTP, TCP, UDP, IP.

CO-PO Mapping:

COs	PO1	PO2	PO3	PSO2
1	3	2	1	1
2	3	3	1	1
3	1	3	2	1
4	2	3	1	1
5	1	3	3	1
6	3	1	1	1

3- Strongly linked, 2- Moderately linked, 1- Weakly linked **SYLLABUS:**

UNIT I **12 Hours**

Network Hardware-Network Software-and Transmission Media

OSI-TCP/IP reference models, ARPANET, Network Topologies,
Physical Layer: Transmission media: Guided Media-Twisted Pair-Coaxial Cable-Fiber optics-Unguided Media: Electromagnetic Spectrum, Radio Transmission and Microwave Transmission. Switching Techniques: Circuit Switching-Packet Switching-Message Switching.
Taxonomy of networking devices.

Unit II Data link layer

12 Hours

Design Issues: Framing-error detection and correction-CRC-Elementary Data link Protocols: Stop and wait-Sliding Window protocols: Go-back-n-Selective Repeat ARQ
Medium Access sub layer and Multiple Access protocols: ALOHA-CSMA-IEEE Standard 802.3 and Ethernet-IEEE Standard 802.4: Token bus.
Data Link Control Protocols: HDLC-SLIP-PPP

Unit III Network Layer

12 Hours

Network Layer design issues-Virtual circuit and Datagram Subnets-Routing algorithms: shortest path routing-Flooding-Hierarchical Routing-Distance vector routing-Broad cast and Multi cast routing, Congestion Control: Congestion prevention policies.
Internet control protocols

Unit IV Transport Layer

12 Hours

Transport Services-Connection management - Elements of Transport Protocols-Internet Transport Protocols: UDP and TCP. IPv4 & IPv6, Application Layer-Domain name system - Electronic Mail-WWW
ATM Transport protocol-Proxy Servers-Multimedia: Data compression-digital representation of analog signals

48 Hours

Textbook (s)

1. Andrew S Tanenbaum, Computer Networks, 4th Edition, Pearson Education /PHI, 2002.
2. Behrouz A. Forouzan, Data Communications and Networking, 3rdEdition, Tata McGraw Hill Higher Education, 2003

Reference (s)

1. Willam Stallings, Data and Computer Communications,8th Edition, Pearson Prentice Hall, 2007.
2. W.A. Shay, Thomson, Understanding communications and Networks, 3rd Edition, Cengage Learning, 2005

21MLC11 Computer Vision & Pattern Recognition

3 0 0 3

Course Outcomes

1. Explain fundamental computer vision concepts, including applications and challenges.
2. Understand image processing techniques and their application in computer vision.
3. Apply feature detection and matching techniques to analyze image data.
4. Understand pattern recognition and machine learning concepts and their application in image analysis.
5. Describe and implement segmentation techniques to partition images into meaningful regions.
6. Understand and apply object detection and recognition techniques for identifying and analyzing objects in images or sequences.

CO–PO Mapping

CO	PO1	PO2	PO3	PO12	PSO1
1	3	2	1	1	1
2	3	2	2	1	1
3	3	3	3	1	3
4	3	3	3	2	3

5	2	3	3	2	3
6	2	3	3	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Computer Vision and Image Processing

Computer Vision: Introduction to computer vision - applications of computer vision - Image formation, Geometric primitives, 2D transformations, and photometric image formation - Sampling and aliasing in images: Image processing Techniques: Point operators: Thresholding, Image Enhancement - Linear filters - Non-Linear filters - Multi-resolution representations : Gaussian pyramid, Laplacian Pyramid - wavelets: 1D Haar wavelet transform , 2D Haar wavelet transform.

Neighborhood operators (morphological operations) and content-based image retrieval.

Unit II **12 Hours**

Feature Detection and Matching

Feature detection : Image features - Edge Detection - Overview of edge detection methods, Prewitt, Sobel, and Canny Edge detectors - Line detection: Hough Transform for line detection - Feature descriptors: Histogram of Oriented Gradients (HOG), Local Binary Pattern (LBP), SIFT and SURF- Feature Matching techniques - Performance Analysis.

Harris corner detection, and Piecewise Linear Approximation

Unit III **12 Hours**

Pattern Recognition and Machine Learning

Introduction to pattern recognition: Pattern, Features, RST invariant features - Pattern recognition systems - Different approaches to pattern recognition - Statistical Pattern Recognition: Bayesian decision theory and discriminant functions. Supervised learning: Parametric methods (linear regression), Nonparametric methods (knearest neighbors, decision trees) - Unsupervised learning: Clustering algorithms (k-means, mean shift clustering). Neural pattern recognition: Structure and types of neural networks, Learning algorithms for neural networks. *Dimensionality reduction methods- PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis)*

Unit IV **12 Hours**

Image Analysis and Object Recognition

Segmentation techniques: Region-based segmentation: Split and merge, region growing, Contour-based segmentation: Active contours, Clustering-based segmentation: K-means, mean shift. Object detection: Techniques for object detection - Face recognition: Methods and challenges in face recognition - Instance recognition Techniques – Image Classification – Semantic segmentation - Motion estimation: Optical flow, activity recognition, motion estimation.

Medical image segmentation and Deep learning object detection

Total: 48 Hours

Textbook (s)

1. Richard Szeliski, Computer Vision: Algorithms and Applications , Springer, 2022, ISBN:978-1848829343
2. Rober.J. Shelkoff, John Wiley &Sons,Pattern Recognition- Statistical, Structural and Neural Approaches,Wiley, 2007, ISBN: 978-8126513703.

Reference (s)

1. R.O.Duda, P.E.Hart and D.G.Stork, Pattern Classification, Wiley India, 2006, ISBN: 978-8126511167.
2. Simon J. D. Prince, Computer Vision: Models, Learning, and Inference, Cambridge University Press, 2012,78-1107011793
3. Milan Sonka, Vaclav Hlavac, Roger Boyle, Image Processing, Analysis, and Machine Vision, Brooks/Cole, 2007, ISBN:978-0495082521.

21CSC21 Web Programming Languages

3 0 0 3

Course Outcomes

1. Understand web applications.
2. Analyze the templates and common scenarios of web development.
3. Examine various web programming languages and their usages.
4. Analyze the flow of data through various layers of web.
5. Identify the use of data management and compatibility of web browsers.
6. Identify the need of security and performance for a web application.

COs–POs Mapping

COs	PO1	PO2	PO3	PO5	PSO1	PSO2

1	3	2	2	1	3	2
2	2	3	3	2	3	1
3	3	2	3	3	1	3
4	2	3	2	2	1	3
5	2	2	2	3	2	2
6	1	2	3	1	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Overview of the web

Web application Overview: Basic concepts of web, Hyper Text Mark-up Language (HTML) and Cascading style sheets (CSS), Web programming languages

Web server: Introduction, Anatomy of HTTP transaction, Request and response structures, REST APIs. Data management: Browser storage - Local storage & Session storage, Database storage *HTML, CSS, HTTP, Status codes*

Unit II

12 Hours

Web programming in JavaScript

Introduction: Basic concepts – Arrays, objects, functions, classes and interfaces, ES6 JavaScript standard. Node JS: Introduction, Node packages, HTTP web server – Creating a HTTP server that supports Create, Read, Update and Delete operations via various HTTP methods, Adding common request parser

Express: Creating a web server with express and body parser packages *NodeJS, ES6, Node packages.*

Unit III **12 Hours**

Introduction: Basic concepts – Arrays, objects, methods, classes and interfaces, Extensible Markup Language (XML).

Web programming techniques – Servlets, Java Server Pages (JSP)

Servlets: Introduction, creating a HTTP servlet for Create, Read, Update and Delete operations

Java Server Pages (JSP): Introduction, creating a JSP for Create, Read, Update and Delete operations

Servlet, XML, JSP

Unit IV **12 Hours**

Web programming in Python

Introduction: Basic concepts – Arrays, objects, methods, classes and interfaces, Python's HTTP package

HTTP server: Introduction, creating a server that supports Create, Read, Update and Delete operations

HTTP server, Python packages

Total: 48 Hours

Textbook (s)

1. Learn Web Development with Python, Fabrizio Romano, Gaston C. Hillar, Arun Ravindran, 1st Edition, 2018.
2. Learn Java for Web Development, Vishal Layka, 1st Edition, 2014.
3. Node.js, John Bach, Alexander Aronowitz, 3rd Edition, 2021

Reference (s)

1. HTML 5 in simple steps, Kogent Learning Solutions Inc, Dreamtech Press.
2. Beginning HTML, XHTML, CSS, and JavaScript, John Duckett, Wiley India.
3. Beginning CSS: Cascading Style Sheets for Web Design, Ian Pouncey, Richard York, Wiley India.
4. Web Designing & Architecture-Educational Technology Centre, University of Buffalo.

21MLC31 Fundamentals of Cloud Computing

3 0 0 3

Course Outcomes

1. Interpret the architecture and infrastructure models of cloud computing, strengths, and limitations of cloud computing.
2. Understand the virtualization concepts of machines and datacenters.
3. Infer the design concepts of cloud ready applications
4. Compare different cloud centre's implementation
5. Understand the concepts of cloud scaling and disaster recovery

6. Interpret the security and risk issues in cloud computing

COs–POs Mapping

COs	PO2	PO6	PO7	PO8
1	3	2	3	1
2	3	1	2	3
3	2	3	1	2
4	2	3	3	1
5	2	3	1	2
6	2	3	3	3

3–Stronglylinked|2–Moderatelylinked|1–Weaklylinked

Unit I **12 Hours**

Cloud Computing

Cloud computing: Introduction, SOA, Cloud computing architectures, Value of cloud computing, Cloud Infrastructure models including SaaS, PaaS, IaaS, Storage-as-a-Service, Database-as-a-Service, Information-asa-Service,Process-as-a-Service,Application-as-a-Service, Integration-as-a-Service, and Security-as-a-Service. Public cloud, private cloud and hybrid cloud, Cloud Services.

Before moving into the cloud: Know Your Software Licenses, The Shift to a Cloud Cost Model. History of Cloud Computing, Advantages of Cloud Computing, and Disadvantages of Cloud Computing.

Companies in the Cloud Today, Amazon Web Services, Google services, IBM Cloud, Windows Azure, Tata Cloud, Salesforce.com

Unit II **13 Hours**

Virtualization & Design

Virtualization,Virtualmachine, Implementation Levels of Virtualization,Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Datacentre, Virtualization for Data-Centre Automation. Service Levels for Cloud Applications Ready for the cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management.

*Various hypervisors like VMware, KVM,*oracleVM,

Unit III **12 Hours**

Cloud centres, Scaling a Cloud Infrastructure and Disaster Recovery:

Cloud centres in detail: Comparing approaches, Xen, Eucalyptus, CloudStack, and OpenStack. Cloud Scaling:Capacity Planning, Cloud Scale, Types of cloud scaling.

Disaster Recovery:Disasters in cloud,Disaster Recovery Planning, Cloud Disaster Management.

Requirements for modern data centres-high availability and Service Orientated-Infrastructures(SOI).Modern data centre use case studies.

Unit IV **11 Hours**

Cloud Computing Software Security Fundamentals

Data Security, Network Security, Host Security, Compromise Response.

Cloud information Security Objectives, Cloud Security Services, Relevant Cloud Security Design Principles, Secure Cloud Software Requirements, Approaches to Cloud Software Requirements Engineering, Cloud Security Policy Implementation.

Cloud Computing Risk Issues: The CIA Triad, Privacy and Compliance Risks, Threats to Infrastructure Data and Access Control, Cloud Access Control Issues, Cloud Service Provider Risks.

MicrosoftAzure, IBMBluemix

Total:48 Hours

Textbook(s)

1. GeorgeReese,Cloud ApplicationArchitectures,1st Edition O'ReillyMedia,2009
2. RonaldL.KrutzandRussell DeanVines,CloudSecurity,1st Edition,WileyPublishing,2010

Reference(s)

1. MichaelMiller,CloudComputing-WebBasedApplicationsthatchangethewayyouworkandcollaborateonline, 1st Edition,PearsonEducation,Publishing, 2011
2. KaiHwang,GeoffreyCFoxandJackJ.Dongarra,Distributed&CloudComputingfromParallelProcessingto theInternet ofThings,1st Edition, MK Publishing,2010
3. DavidSLinthicum,CloudComputingandSOAConvergenceinYourEnterprise:AStep-by-StepGuide,1st Edition, Addison-Wesley, 2009

21CS004 Principles of Programming Languages

3 0 0 3

Course Outcomes

1. Explain the concepts of programming languages.
2. Describe syntax and semantics of programming languages.
3. Explain data types, and basic statements of programming languages.
4. Illustrate the concept of subprogram.
5. Demonstrate the concepts of ADT and OOP.
6. Summarize the concept of exception handling and event handling.

CO–PO Mapping

COs	PO1	PO2	PO3	PO4
1	3	1	2	2
2	3	2	1	2
3	3	1	3	1
4	3	2	2	2
5	3	2	2	3
6	3	2	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked **Unit I**

Preliminary Concepts 12 Hours

Reasons for studying concepts of programming languages, programming domains, language evaluation criteria, influences on language design, language categories, language design trade-offs, implementation methods, programming environments, Evolution of Major Programming Languages.

Syntax and Semantics: General problem of describing syntax, formal methods of describing syntax. *attribute grammars, describing the meanings of programs.*

Unit II

Data types, Expressions and Statements, Control Structure 12 Hours

Names, Bindings, and Scopes: Introduction, names, variables, concept of binding, scope, scope and lifetime, referencing environments, named constants

Data types: Introduction, primitive, character, string types, user defined ordinal types, array, associative arrays, record, tuple types, list types, union types, pointer and reference types, type checking, strong typing, type equivalence

Expressions and Statements: Arithmetic expressions, overloaded operators, type conversions, relational and Boolean expressions, short- circuit evaluation, assignment statements, mixed-mode assignment **Control Structures:** introduction, selection statements, iterative statements, unconditional branching. *Guarded commands.*

Unit III

Subprograms 12 Hours

Subprograms: Fundamentals of subprograms, design issues for subprograms, local referencing environments, parameter passing methods, parameters that are subprograms, calling subprograms indirectly, overloaded subprograms, generic subprograms, design issues for functions, user defined overloaded operators, closures, co routines

Implementing subprograms: General semantics of calls and returns, implementing simple subprograms, implementing subprograms with stack-dynamic local variables, nested subprograms, blocks. *Implementing dynamic scoping.*

Unit IV

ADT, OOP, Exception and Event Handling 12 Hours

Abstract Data types: The concept of abstraction, introductions to data abstraction, design issues, language examples, parameterized ADT, encapsulation constructs, naming encapsulations

Object Oriented Programming: Design issues for OOP, OOP in Smalltalk, C++, Java, Ada 95, C#, Implementation of Object-Oriented constructs.

Exception Handling and Event Handling: Introduction, exception handling in Ada, C++, Java. *Introduction to event handling, event handling with Java and C#.*

Total: 48 Hours

Textbook (s)

1. Robert W. Sebesta, "Concepts of Programming Languages", Tenth Edition, Addison Wesley, 2012.
2. Programming Languages, Principles & Paradigms, 2ed, Allen B Tucker, Robert E Noonan, TMH

Reference (s)

1. R. Kent Dybvig, "The Scheme programming language", Fourth Edition, MIT Press, 2009.
2. Jeffrey D. Ullman, "Elements of ML programming", Second Edition, Prentice Hall, 2098.
3. Richard A. O'Keefe, "The craft of Prolog", MIT Press, 2009.
4. W. F. Clocksin and C. S. Mellish, "Programming in Prolog: Using the ISO Standard", Fifth Edition, Springer, 2003

21CS005 Mobile Computing

3 0 0 3

Course Outcomes

1. Explain the basic concepts and fundamentals of mobile computing and telecommunication systems along with various standards.
2. Illustrate the techniques, protocols related to GSM and GPRS architecture to perform requirements analysis.
3. Explain major components of Mobile IP to improve the service qualities of a network
4. Compare various ad hoc routing protocols to examine the performance of network
5. Explain the architecture of Wireless Sensor Network and WLAN design issues and limitations.
6. Outline the basic knowledge in developing smart phone applications using various platforms, toolkits, APIs and third party libraries

CO–PO Mapping

COs	PO1	PO2	PO3	PO8
1	1	1	3	2
2	1	1	2	2
3	2	2	1	2
4	1	1	2	2
5	2	2	2	2
6	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction to Mobile Communications **12 Hours**

Overview: Mobile Communication and Mobile Computing – Architecture; Generations of Telecommunications – 1G, 2G, 3G, 4G and 5G; Wireless Technologies – WPAN, WLAN, WMAN; Wireless Medium Access Control (MAC) – Overview, SDMA, TDMA, FDMA, CDMA;

2G Technologies: GSM – Services, Architecture, Radio Interface, Protocol Stack, Localization, Call Handling, Handover, Security; 2.5G – GPRS and 3G – EDGE.

Networking: Communication Modes, Basic Network Designs, Cellular Infrastructures

Unit II

Mobile Network Layer and Transport Layer **12 Hours**

Mobile Network Layer: Mobile IP Overview, IP Packet delivery, Agent discovery and advertisement, Registration, Tunneling and Encapsulation, Optimizations, Security, and Dynamic Host Configuration Protocol (DHCP).

Mobile Transport Layer – Motivation, Traditional TCP, Classical TCP Approaches: Indirect TCP, Snooping TCP, Mobile TCP, Transaction-oriented TCP; Optimizations, TCP for 2.5G/3G.

Multi Task gadget: wide area mobile data - air link standards for data - wireless application environment

Unit III

Mobile Ad-hoc Network (MANET) **12 Hours**

Introduction to Mobile ad-hoc networks, Characteristics and features, Applications, Limitations; Routing protocols – Design Issues, Routing algorithms: Proactive (DSDV & OLSR) and Reactive (DSR & AODV), Security in ad hoc networks; Wireless LAN – IEEE 802.11 – System Architecture, Protocol Layers.

Wireless Sensor Network (WSN): Introduction, Architecture, Applications, Properties and Security.

Satellite systems: history - applications - basics - broadcast systems

Unit IV

Mobile Platforms and Applications **12 Hours**

Mobile OS: Overview on Mobile Device Operation Systems (Android, iOS, Black Berry) – Architecture, App development Kit; Introduction to Network Simulators: Characteristics, Applications, Limitations, Types of Simulators: Wireless Application Protocol (WAP): Introduction, Architecture, Applications.
Application layer Protocols – FTP, SMTP, HTTP, DNS; Windows 10.

Total: 48 Hours

Textbook (s)

1. Raj Kamal, Mobile Computing, Oxford press, Third Edition, 2018
2. Jochen Schiller, Mobile Communications, Pearson Eduction, Secon Edition, 2020

Reference (s)

1. Asoke K Talukder, Hasan Ahmad and RoopaYavagal, Mobile Computing, Second Edition, McGraw Hill, 2010 2. Prasant Kumar Pattnail and Rajib Mall, Fundamentals of Mobile Computing, Second Edition, PHI Learning Pvt. Ltd., 2015
3. Frank Adelstein, et al., Fundamentals of Mobile and Pervasive Computing, McGraw Hill, 2005
4. http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf (NS2 manual)

21CS006 Distributed Operating Systems

3 0 0 3

Course Outcomes

1. Summarize the fundamental concepts of Distributed Operating Systems.
2. Illustrate the Concepts of Message passing system and Remote Procedure Calls.
3. Design and Implementation of Distributed Shared Memory and Structures.
4. Utilize the Synchronization and Distributed File Systems.
5. Make use of Resource Management and Process Management Concepts.
6. Outline the concept of Naming and Security in Distributed Operating Systems.

CO–PO Mapping

CO	PO1	PO2	PO3	PO12
1	3	2	2	1
2	3	2	2	1
3	3	2	2	1
4	2	2	2	1
5	3	2	2	1
6	3	1	2	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction and Communication in Distributed System **12 Hours**

Introduction: Design Issues - Distributed Computing Environment - Message Passing - Features of Good Message Passing System-Issues in IPC by Message Passing-Synchronization- Buffering

Remote Procedure Calls: The RPC Model - Transparency of RPC- Implementing RPC Mechanism - Stub Generation - RPC Messages - Marshaling Arguments and Results - Server Management - Parameter-Passing Semantics- Call Semantics

Communication Protocols for RPCs - Complicated RPCs - Client-Server Binding-Case Study: Sun RPC

Unit II

Distributed Shared Memory and Synchronization **12 Hours**

Distributed Shared Memory: General Architecture of DSM Systems- Design and Implementation Issues of DSM – Granularity - Structure of Shared Memory Space

Synchronization: Clock Synchronization - Event Ordering - Mutual Exclusion –Deadlocks- Election Algorithms

Consistency Models-Replacement Strategy - Thrashing

Unit III

Resource Management and Distributed File Systems **11 Hours**

Resource Management: Features of a Good Global Scheduling Algorithm - Task Assignment Approach – Load Balancing Approach – Load Sharing Approach

Distributed File Systems: Introduction -Desirable Features of a Good Distributed File System - File Models – File Accessing Models – File Sharing Semantics – File Caching Schemes - File Replication

Process Management - Process Migration- Threads- Case Study: DCE Distributed File Service

Unit IV

Naming and Security **12 Hours**

Naming: Desirable Features of a Good Naming System - Fundamental Terminologies and Concepts - System-Oriented Names - Object-Locating Mechanisms - Human-Oriented Names - Name Caches Security: Potential Attacks to Computer Systems - Cryptography – Authentication

Access Control - Digital Signatures - Design Principles- Case Study: DCE Directory Service

Total: 48 Hours

Textbook (s)

1. Pradeep K Sinha, “Distributed Operating Systems: Concepts and Design”, Prentice Hall of India, 2009
2. Andrew S Tannebaum, ”Distributed Operating Systems”, Pearson Education, 2007

Reference (s)

1. George Coulouris, Jean Dollimore and Tim Kindberg, “Distributed Systems Concepts and Design”, Fifth Edition, Pearson Education, 2012
2. Tanenbaum A.S., Van Steen M., “Distributed Systems: Principles and Paradigms”, Pearson Education, 2007
3. Liu M.L., “Distributed Computing, Principles and Applications”, Pearson Education, 2004.
4. Nancy A Lynch, “Distributed Algorithms”, Morgan Kaufman Publishers, USA, 2003

21ML507 Neural Networks Lab

0 0 3 1.5

Course Outcomes

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

1. Understand and demonstrate building blocks of neural networks
2. Design and implement the basic neural network model
3. Design and implement multilayer neural network models
4. Demonstrate forward and backward propagation in neural network models.
5. Illustrate the working of Convolutional Neural Networks and auto encoders.
6. Apply and analyze the performance of neural network models on real time applications

CO–PO Mapping

COs	PO4	PO5	PO8	PSO1
1	3	2	2	2
2	3	3	2	2
3	3	3	2	3
4	2	2	3	3
5	2	2	2	3
6	1	1	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked **Lab Manual**

1. Neural Networks Laboratory– AIML Department, GMRIT.

Textbook (s)

1. Simon Haykin, "Neural Networks - A comprehensive foundation", Pearson Education, 2003.
2. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation,2015. **Reference (s)**

1. SatishKumar,"Neural Networks: A Classroom Approach" Tata McGraw Hill Education, 2004.
2. Josh Patterson and Adam Gibson "Deep Learning A Practitioner's Approach" O'Reilly Media, Inc. 2017

List of Experiments

1. Importing and Exporting Datasets, Splitting Datasets to Training Dataset and Test Dataset, Define the Structure and Parameter's Initialization in a Neural Network Model.

2. Implement the following types of Activation Functions in neural network.
   a. Binary Step Activation Function
   b. Linear Activation Function
   c. Sigmoid Activation Function
   d. Tanh Activation Function
   e. RELU Activation Function
   f. Softmax Activation Function

3. Implement the following Cost Functions
   a. Mean Square Error
   b. Root Mean Square Error
   c. Mean Absolute Error
   d. Cross-Entropy Loss

4. Implementation of ANN with McCulloch Pitts Neuron Model.

5. Implementation of Bidirectional Associative Memory (BAM).

6. Implement the Single-layer perceptron network and test the same using appropriate dataset.
7. Implement the multi-layer perceptron network and test the same using the appropriate dataset.
8. Implement K-fold cross validation by following the below steps. (On any dataset)
   a) Randomly split your entire dataset into k-folds
   b) For each k-fold in your dataset, build your model on k – 1 folds of the dataset. Then, test The model to check the effectiveness for kth fold
   c) Record the error you see on each of the predictions
   d) Repeat this until each of the k-folds has served as the test set
   e) The average of your k recorded errors is called the cross-validation error and will serve as your performance metric for the model
9. Implement Back propagation algorithm in neural network.
10. Implement the Auto-Encoder in neural network .
11. Implement the Variational Auto-encoder in neural network.
12. Explore the Convolutional Neural Networks (Convolutional Layers, Pooling Layer, fully connected layer and Hyper parameters).
13. Digit and Character recognition using CNN.
14. Explore the Simple Recurrent Neural Network.
15. Perform Image segmentation by taking single image as input.
16. Perform object detection using Region based CNN (R-CNN).

Augmented Experiments: (Students shall opt any one of the Augmented Experiments in addition to the regular experiments)

1. Human Facial Recognition.
2. Stock Market Prediction.
3. Social Media.
4. Aerospace.
5. Defense.
6. Healthcare.
7. Signature Verification.
8. Handwriting Analysis
9. Speech Recognition
10. Weather Forecasting

21TPX01 Term Paper

0 0 3 1.5

Course Outcomes

1. Interpret the literature to link the earlier research with the contemporary technologies
2. Communicate effectively as an individual to present ideas clearly and coherently
3. Review the research findings and its correlation to the latest applications
4. Prepare documents and present the concepts clearly and coherently
5. Inculcate the spirit of enquiry for self-learning
6. Identify interdisciplinary oriented topics

COs – POs Mapping

COs	PO1	PO4	PO10	PO12
1	-	2	-	-
2	-	-	3	3
3	3	-	-	-
4	-	-	3	-
5	-	-	-	3
6	1	-	-	-

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Term Paper: The term paper is a self-study report with an objective enhancing the students' written technicalcommunication and shall be carried out in 5th semester. Every student will take up this either individually or as a batch of maximum four members and submit a report. The scope of the term paper could be an exhaustive literature review choosing any engineering concept with reference to standard research papers or an extension of the concept of earlier course work in consultation with the term paper supervisor.

The report will be evaluated by a committee nominated by HoD with the approval of the Controller of Examinations (CoE)

21ESX02 Employability Skills II

0 0 2 0

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs – POs Mapping

COs	PO1	PO2	PO5	PO 8	PO10	PO12
CO1					3	2
CO2				1	2	2
CO3	2	1		2		
CO4	2		2			
CO5	2		2			
CO6	2		2			

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Soft Skills:

Sl No.	5th Semester (Topic & Content)	No. of Periods

1.	**Introduction to Campus Placements:** Stages of Campus Placement, Skills assessed in Campus Placements & How to get ready?	02
2.	**Motivational Talk on Positive Thinking:** Beliefs, Thoughts, Actions, Habits & Results (Success)	02
3.	**Resume Preparation:** Resume? Templates? Mistakes to be avoided in a Resume, Steps to be followed in preparing it.(with examples)	03
4.	**Group Discussions** (Recap)**:** GD? Stages of a GD, Skills assessed in a GD, Blunders to be avoided, How to excel in a GD? (through Practice Sessions)	03
5.	**Psychometric Tests:** Definition, Types of Psychometric Tests: Numerical Computation, Data Interpretation, Verbal Comprehension, Verbal Critical Reasoning and Personality Questionnaires	03
6.	**Exercises related to Communication:** Story Writing, TAT etc	03
	Total Periods	**16**

Quantitative Aptitude:

Sl No.	Semester-V (Topic)	No. of Periods
1.	Square &Cube roots	02
2.	Partnership	02
3.	Partnership	02
4.	Logarithms	02
5.	Progressions	02
6.	Mensuration	02
7.	Mensuration	02
8.	Data Sufficiency	02
	Total Periods	**16**

Domain Specific: (Data Structures and Algorithms (DSA)-Phase-1) 16 Hours

Module-1:
Introduction: Importance of Competitive Programming, Conditional Statements, Loops, Nested Loops. Strings, Arrays, Recursion and pointers

Practice Problems:

Week	Topic	Practice Problem Links
1	Introduction to CP	Creating accounts on Globla Coding Platforms
2	Conditional Statements	https://www.codechef.com/practice/tags/conditional-statements
3	Loops	https://www.codechef.com/practice/tags/loops https://leetcode.com/tag/math/

		https://www.interviewbit.com/practice/#topics[]=math
4	Nested Loops	https://www.codechef.com/practice/topics/math
		https://www.interviewbit.com/practice/#topics[]=math
5	Strings	https://www.codechef.com/practice/topics/strings
		https://leetcode.com/tag/string/
		https://www.interviewbit.com/practice/#topics[]=strings
6	Arrays	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=arrays
		https://leetcode.com/tag/array/
		https://www.codechef.com/practice/topics/arrays
		https://www.interviewbit.com/practice/#topics[]=arrays
7	Two Pointer	https://www.interviewbit.com/practice/#topics[]=two-pointers
		https://leetcode.com/tag/two-pointers/
8	Recursion	https://leetcode.com/tag/recursion/
		https://www.codechef.com/practice/tags/recursion

Module-2:
Searching, Sorting, Linked List, Linked List: Double Linked List, Circular Linked List, Stack, Queue, hashing

Practice Problems:

Week	**Topic**	**Practice Problem Links**
9	Searching-I	https://www.codechef.com/practice/topics/binary-search https://leetcode.com/tag/binary-search/ https://www.interviewbit.com/practice/#topics[]=binary-search https://www.codechef.com/practice/tags/searching
10	Searching-II	https://www.codechef.com/practice/topics/binary-search https://leetcode.com/tag/binary-search/ https://www.interviewbit.com/practice/#topics[]=binary-search https://www.codechef.com/practice/tags/searching
11	Sorting-I	https://www.codechef.com/practice/topics/sorting https://leetcode.com/tag/sorting/ https://leetcode.com/tag/merge-sort/
12	Sorting-II	https://www.codechef.com/practice/topics/sorting https://leetcode.com/tag/sorting/
13	LinkedList: Double LinkedList, Circular LinkedList	https://www.hackerrank.com/domains/data-structures?filters%5Bsubdomains%5D%5B%5D=linked-lists https://leetcode.com/tag/linked-list/ https://www.interviewbit.com/practice/#topics[]=linked-lists https://leetcode.com/tag/doubly-linked-list/
14	Stack	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=stacks https://leetcode.com/tag/stack/ https://www.interviewbit.com/practice/#topics[]=stacks-and-queues

15	Queue	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=queues
		https://leetcode.com/tag/queue/
		https://www.interviewbit.com/practice/#topics[]=stacks-and-queues
16	Hashing	https://www.interviewbit.com/practice/#topics[]=hashing

References:

1. https://leetcode.com
2. https://www.codechef.com
3. https://www.hackerearth.com
4. https://www.interviewbit.com
5. https://www.hackerrank.com

21HSX12 CC & EC Activities II

0 0 1 0

Course Outcomes

1. Interpret and present the abstractive technical information through an activity
2. Think critically in providing solutions to the generic and common problems
3. Demonstrate the creative thinking in dealing with liberal arts
4. Instill team sprit through active engagement with the peer
5. Develop programs of common interest having social impact
6. Empower the under privileged through motivational activities

COs – POs Mapping

COs	PO6	PO7	PO9	PO10
1	-	-	-	3
2	3	2	-	-
3	3	-	-	-
4	-	-	3	-
5	3	-	-	-
6	3		-	-

Co-Curricular and Extra Curricular (CCEC) Activities:

Students shall acquire 1 credit each in 2nd and 3rd years with the following scheme:
Scheme of evaluation for the CCEC activities:

- No. of slots in each Semester @ 2 slots every week : 24
- No. of Stream (2-CC + 1-EC) : 3
- No. of slots allotted for each stream : 8

Requirement for the award of 1- Credit

- Students shall choose at least two streams of events in each semester
- Students shall secure 75% attendance in each stream of events to obtain a certificate
- Students shall obtain 2 certificates of Participation in each semester.

The credits earned through these courses will be indicated in the grade sheet and will not beaken into account for CGPA calculation.

21SIX01 Summer Internship I

0 0 0 1

Course Outcomes

1. Demonstrate the application of knowledge and skill sets acquired from the course and workplace in the assigned job function/s
2. Solve real life challenges in the workplace by analyzing work environment and conditions, and selecting appropriate skill sets acquired from the course
3. Articulate career options by considering opportunities in company, sector, industry, professional and educational advancement
4. Communicate and collaborate effectively and appropriately with different professionals in the work environment through written and oral means

5. Demonstrate the ability to harness resources by examining challenges and considering opportunities
6. Demonstrate appreciation and respect for diverse groups of professionals by engaging harmoniously with different company stakeholders

COs – POs Mapping

COs	PO1	PO2	PO8	PO10	PO12
1	3	-	-	-	-
2	3	-	-	-	-
3	-	-	-	-	3
4	-	-	-	3	-
5	-	2	-	-	-
6	-	-	3	-	-

Summer Internship: As a part of curriculum in all branches of Engineering, it is mandatory for all students to undergo summer internship Programme at industries (core or allied) / R & D organization to get practical insight of their subject domain during summer break after the 4th semester. This summer internship Programme shall be availed to a maximum duration of 4 weeks and the assessment shall be carried out with both internal and external experts leading to “Satisfactory” and “Non-Satisfactory Performance”, and it will not be accounted for the calculation of CGPA.

Audit Course

0 0 0 0

Course Outcomes

1. Interpret the meaning of values and select their goals by self- Investigation based on personal values activity
2. Interpret the major events and issues related to a period in Indian history
3. Assess the benefits and limitations of science and its application in technological developments towards human welfare
4. Check the awareness regarding basic human rights and to uphold the dignity of every individual
5. Assess the individual and group behaviour, and understand the implications of organizational behaviour on the process of management
6. Determine the appropriateness of various leadership styles and conflict management strategies used in organizations

COs – POs Mapping

COs	PO12
1	3
2	3
3	3
4	2
5	3
6	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Audit Courses: Audit courses are among the compulsory courses and do not carry any credits. All the students shall register for one Audit courses in the beginning of 3rdsemester. List of the courses will be notified at the beginning of the third semester for all students and the student has to choose one audit course under self-study mode at the beginning of third semester. All the students (regular and lateral entry students) shall complete the audit course similar to other regular courses and the results will be indicated with “Satisfactory” or “Not Satisfactory” performance.

6th Semester

21ML601 Deep Learning Techniques

3 0 0 3

Course Outcomes

1. Explain the fundamental of Artificial Neural Networks 2. Identify different data representations for Neural Networks
3. Make use of different CNN models.
4. Compare different character encoding techniques.
5. Exemplify different Deep unsupervised models.
6. Examine different deep learning applications.

CO–PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PSO1
1	3	2	2	1	3	2
2	3	3	2	1	2	2
3	3	3	3	2	3	2
4	3	2	3	2	3	2
5	3	2	2	2	2	2
6	3	3	3	3	3	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Neural Networks

Artificial Neural Networks: Introduction, Neuron Model, Neural Network Architecture, Learning Rules, Single Layer Perceptrons, Multilayer Perceptrons, adaptive resonance theory (ART), Back propagation Networks: Kohnen's self-organizing networks, Hopfield network, Applications of NN, Data representations for neural networks : 0D tensors, 1D tensors, 2D tensors, tensor attributes, data tensors (Vector data, Timeseries data, images, video), tensor operations.

Model Parameters vs Hyper parameters, Types of activation functions, Gradient descent, delta rule

Unit II **12 Hours**

Convolution networks: Building blocks of CNNs, Architectures, Filters and Feature Maps, pooling layers, Convolutions over volumes, SoftMax regression, Deep Learning frameworks, Training and testing on different distributions, Bias and Variance with mismatched data distributions, Transfer learning, Multi-task learning, end-to-end deep learning.

Keras , tensorflow, data augmentation, Batch Normalization, Dropout

Unit III **12 Hours**

CNN models: Alex Net, VGG –16, Residual Networks, YOLO

Recurrent Networks: One-hot encoding of words and characters, using word embeddings, Recurrent Neural Network Model, vanishing gradients with RNNs, Gated Recurrent Unit (GRU), LSTM (long short-term memory).

Bidirectional RNN, Exploding gradient.

Unit IV **12 Hours**

Sequence Models: Encoder Decoder sequence to sequence architectures

Deep Unsupervised Learning: Autoencoders, variational Autoencoders, Generative adversarial network, Deep Boltzmann Machines.

Applications of Deep Learning: Image segmentation, object detection, etc.,

Batch Gradient Descent vs Stochastic Gradient Descent, limitations of deep learning.

Total: 48 Hours

Textbook (s)

1. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation. (2015).
2. S.N. Sivanandam, S.N. Deepa. “Principals of Soft Computing”, An WILLY publications 3rd edition 2009.
3. Fundamentals of Deep Learning: Designing Next-generation Machine Intelligence Algorithms by Nicholas Locascio and Nikhil Buduma O'Reilly Media; 1 edition (June 29, 2017) 4. Simon S. Haykin, Neural Networks, Prentice Hall, 2nd edition

5. B. Yegnanrayana , “Artificial Neural Networks” , PHI.

Reference (s)

1. Franchois Chollet, Deep Learning with Python
2. Deep Learning: A Practitioner's Approach by Adam Gibson and Josh Patterson Shroff/O'Reilly; First edition (2017)
3. Python Deep Learning by Daniel Slater and Gianmario Spacagna, Packt Publishing; 2/e (January 16, 2019)
4. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006
5. Kevin P. Murphy.,Machine Learning:A Probabilistic Perspective

21ML602 AUTOMATA THEORY AND LANGUAGE PROCESSORS

3 0 0 3

Course Outcomes

1. Understand State machines, languages and computations
2. Understand regular grammars and regular languages
3. Design Pushdown automata for Context free languages
4. Acquire knowledge of compiler & its Phases
5. Construct parse table for a given context free grammar
6. Apply code optimization techniques to improve the performance of a program

CO–PO Mapping

COs	PO1	PO2	PO3	PO5	PO12
1	3	2	2	1	2
2	3	3	2	--	1
3	3	3	3	1	2
4	2	3	1	2	2
5	3	2	2	2	2
6	3	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Finite Automata: Finite automata model-Deterministic Finite Automata – Nondeterministic Finite automata - Recognition of a language by an Automaton - Equivalence of DFA and NFA, Minimization of Finite Automata **Regular sets:** Regular Expressions-Arden's theorem –Pumping Lemma for Regular Languages - Closure Properties of Regular sets.

Applications of finite automata, Equivalence of finite automata.

Unit II **12 Hours**

Context free language- Chomsky Classification of languages- Context-free grammar -simplification of Context free Grammar- Chomsky Normal Form-Closure Properties of Context free Languages.

Pushdown Automata: PDA Model-Design-Acceptance by empty stack and final state- Non-deterministic PDA.

Context Sensitive Languages, Linear bounded automata

Unit III **12 Hours**

Language Processing-Phases of a Compiler-Top-down Parsing: Derivation of a string- ambiguous grammar-Left recursion-Left factoring-First and Follow-Non-Recursive Predictive Parsing- LL(1) Grammar-Bottom-up parsing: Model of an LR Parsers-Construction of SLR Parsing Table.

Shift-Reduce parsing – Regular definitions

Unit IV **12 Hours**

Intermediate Code generation and optimization: Three Address Code- Quadruples-Triples-Indirect TriplesMachine Independent Code Optimization: Common Sub-expression Elimination-Constant Folding-Copy Propagation-Dead Code Elimination-Strength Reduction-Loop Optimization -Basic Blocks-Flow Graph-DAG for basic Blocks-Machine Dependent Code Optimization: Peephole Optimization

Syntax directed translation – DAG for expressions **Total 48 Hours**

Text Books:

1. J. E. Hopcroft and J. D. Ullman, Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson/Addison Wesley
2. Mishra & Chandra Sekharan, Theory of Computer Science& Automata Language and Computation, 3rd Edition, Prentice Hall of India
3. Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers, Principles Techniques an Tools,2nd Edition, Pearson.

Reference books:

1. P. Linz, Introduction to Formal Language and Computation, 2nd Edition, Narosa, 2006.
2. V. Raghavan, Principles of Compiler Design, 2nd Edition, TMH, 2011.

Internal Assessment Pattern

Cognitive Level	**Int. Test 1 (%)**	**Int. Test 2 (%)**	**Open Book Exam (%)**
Remember	20	20	--
Understand	30	40	--
Apply	50	40	60
Analyze			40
Evaluate	--	--	--
Create	--	--	--
Total (%)	100	100	100

21CS603 Software Engineering

3 0 0 3

Course Outcomes

1. Explain the need of Software Life Cycle Models
2. Build end-user requirements into system and software requirements,
3. Summarize the system models of software engineering
4. Identify and apply appropriate software architectures and patterns to carry out high level design
5. Choose various testing techniques during software development
6. Categorize Risk management and Software quality for software products

COs–POs Mapping

COs	PO1	PO2	PO3	PO5	PO11	PSO1
1	3	2	2	2	2	3
2	3	2	3	2	3	3
3	3	2	3	3	3	3
4	3	2	3	2	3	3
5	3	2	3	2	3	3
6	3	2	2	2	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Software Engineering and SDLC, Software Myths, CMMI, Process models: Linear Sequential model, Prototyping model, Evolutionary models: Spiral model, Agile developmental methodologies-Scrum & XP

Incremental model, software development : Product based and application based

Unit II **12 Hours**

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification. Software Requirements Engineering Process, Feasibility studies, Requirements elicitation and analysis, requirements validation.
System models: Context models, behavioral models, data models, object models.

Structure of Software Requirements Document, Structured analysis methods

Unit III **12 Hours**

Design concepts, data design, software architecture, Architectural styles and patterns, User interface design - Golden rules, User interface analysis and design and steps. Conceptual model of UML, basic structural modeling, Static and Dynamic UML diagrams : class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, etc.,

Data Acquisition System - Monitoring and Control System

Unit IV **12 Hours**

Testing strategies and Risk Management: Testing levels: Unit testing, integration testing, system testing – alpha and beta testing, Testing Types: black box and white box testing techniques, Cyclomatic Complexity, debugging, Risk management - Risk types, strategies, estimation and Planning. Software Quality - Quality assurance and its techniques

Software measurement, metrics for software quality

Total: 48 Hours

Textbook (s)

1. Roger S. Pressman, Software Engineering, A practitioner's Approach, 8th Edition, McGraw-Hill International Edition, 2015
2. I. Sommerville, Software Engineering, 7thEdition, Pearson education, 2004.
3. Rajib Mal, Fundamentals of software Engineering, 4th Edition, Eastern Economy Edition, 2014.

Reference (s)

1. K K Aggarwal and Yogesh singh, Software engineering,3rd Edition, New age international publication,2008

21MLC12 Machine Learning for Business Intelligence

3 0 2 4

Course Outcomes

1. Understand the basic concepts of business analytics.
2. Identify the application of business analytics and use tools to analyze business data.
3. Understand deriving meaning from huge volume of data and information and knowledge discovering process is used in business decision making.
4. Explain various metrics, measures used in business analytics.
5. Illustrate various descriptive, predictive, and prescriptive methods and techniques. 6. Model the business data using various business analytical methods and techniques.

CO–PO Mapping

CO	PO1	PO2	PO3	PO4	PSO1	PSO2
1	3	1	2	1	2	2
2	3	1	3	1	3	3
3	3	3	1	2	2	2
4	1	3	1	3	2	2
5	1	1	3	3	3	3
6	1	2	2	1	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **11+ 9 Hours**

Introduction to Business Analytics: Introduction to Business Analytics, need and science of da decision making, Descriptive, predictive, prescriptive analytics and techniques, big data analytics, \ media analytics, Machine Learning algorithms, framework for decision making, challenges in DD de and future.

der Systems, Automated Decision-Making, Reinforcement Learning for Decision-Making

omponents

1. scriptive, Predictive analysis on numeric data
2. plementation of central tendency models on an appropriate dataset
3. mple case study on Types of Social Media Analytics Report
4. mple case study on Decision making

UNIT II **12+ 9 Hours**

Data mining, Text mining, Web mining, Spatial mining, Process mining, Data warehouse and datamar
process KDD, CRISP-DM, SEMMA and Domain-Specific, Classification and Prediction performance m
-RSME, MAD, MAP, MAPE, Confusion matrix, Receiver Operating Characteristic curve & AUC; Validat
Techniques

- hold-out, k-fold cross-validation, LOOCV, random subsampling, and bootstrapping.

Nested Cross-Validation, Stratified Cross-Validation, Time Series Cross-Validation

Practical Components

1. Measure Prediction performance with confusion matrix on sample dataset
2. Implement ROC curve and AUC
3. Implementation of K-fold cross validation for sample dataset
4. Implementation of bootstrapping with suitable datasets

UNIT III **11+ 6 Hours**

Forecasting Techniques: Introduction, time-series data and components, forecasting accuracy, moving average method, single exponential smoothing, Holt's method, Holt-Winter model, Croston's forecasting method, regression model for forecasting, Auto regression models, autoregressive moving process, ARIMA, Theil's coefficient.

Coefficient of Variation, Shannon Diversity Index, Simpson's Diversity Index, Gini Coefficient

Practical Components

1. Implementation of various smoothing techniques on sample datasets
2. Implement Moving Average forecasting technique
3. Implement Holt-Winter forecasting technique
4. Implementation of regression for forecasting
5. Measure the Theil's coefficient for suitable data

UNIT IV **11+ 6 Hours**

Six Sigma: Introduction, origin, 3-Sigma Vs Six-Sigma process, cost of poor quality, sigma score, industry applications, six sigma measures, DPMO, yield, sigma score, DMAIC methodology, Six Sigma toolbox

Statistical Process Control, Design of Experiments, Theory of Constraints (TOC

Practical Components

1. Illustrate difference between 3-sigma vs six-sigma process
2. Implementation of DMAIC method with suitable scenario
3. Case study on Enhancing Employee Management using Lean Six Sigma toolbox

48+32 Hours

Textbooks:

1. Jaiwei Ham and Micheline Kamber, Data Mining concepts and techniques, Kauffmann Publishers 2006 2. Efraim Turban, Ramesh Sharda, Jay E. Aronson and David King, Business Intelligence, Prentice Hall, 2008.
3. W.H.Inmon, Building the Data Warehouse, fourth edition Wiley India pvt. Ltd. 2005
4. Marc J. Schniederjans, Dara G. Schniederjans, Christopher M. Starkey, "Business Analytics Principles, Concepts, and Applications with SAS", Associate Publishers,2015.

References:

1. S. Christian Albright, Wayne L. Winston, "Business Analytics - Data Analysis andDecision Making", 5th Edition, Cengage, 2015.
2. U Dinesh Kumar, "Data Analytics", Wiley Publications, 1st Edition, 2017.
3. Ralph Kimball and Richard Merz, The data warehouse toolkit, John Wiley, 3^{rd}edition,2013.
4. Michel Berry and Gordon Linoff, Mastering Data mining, John Wiley and Sons Inc,2nd Edition, 2011
5. https://machinelearningmastery.com/time-series-forecasting-methods-in-python-cheat-sheet/
6. https://leansixsigmatoolbox.com/2023/06/08/ongoing-education-lean-six-sigma-employeeengagement/

21CSC22 Web Application Developments Framework

3 0 2 4

Course Outcomes

1. Understand the fundamentals of web framework.
2. Classify model, view and controller layers of a web application.
3. Design a web application using a framework.
4. Know the concept of Java web framework.
5. Understand and analyze how modern-day web applications are different from web sites.
6. Learn the technologies of Python web framework.

COs–POs Mapping

COs	PO1	PO3	PO5	PSO2
1	3	3	1	1
2	3	2	2	2
3	3	3	3	3
4	3	3	2	3
5	3	2	2	3
6	2	3	3	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT- I **10+4 Hours**

Fundamentals of Web Framework

Web framework-History, Types of framework architectures, Model–view–controller (MVC), Three-tier organization Introduction to frameworks-Framework applications, General-purpose website frameworks-Server-side, Clientside features

MVC, Three-tier organisation, Framework

Practical Components

1. Realization of separation of data, data fetch and representation logics
2. Realization of markup- language and it's styling

UNIT- II **14+10 Hours**

Angular – JavaScript web framework

Introduction – Angular MVC, Model, View, Controller, Ajax, Data binding

Angular concepts - Directives, Scopes, Controllers, Modules, Expressions

Developing a simple To-Do application using AngularJS (Developing a single page application) *MVC, Ajax, Data binding*

Practical Components

1. Implementation of angular directives
2. Implementation of angular components
3. Implementation of custom/user-defined directive
4. Create a TODO Application using Angular JS
5. Implement routing and navigation in Angular JS

Unit III **13+12 Hours**

React framework

Introduction to React: What is Full-Stack Web Development?, Node.js and NPM, Front-end JavaScript Frameworks and Libraries Overview, Introduction to React, React App Overview, Introduction to JSX, React Components, React Components: State and Props, React Components: Lifecycle Methods Part 1

React Router and Single Page Applications: Presentational and Container Components, React Components: Lifecycle Methods Part 2, Functional Components, React Virtual DOM, React Router, Single Page Applications, React Router: Parameters

React Forms, Flow Architecture and Introduction to Redux: Controlled Forms, Uncontrolled Components, The

Model-View-Controller Framework, The Flux Architecture, Introduction to Redux, React Redux Forms

Practical Components

1. React Components
2. React Router and Single page applications
3. Controlled Form Validation, Uncontrolled Forms
4. Create a TODO Application using React JS
5. Creating a movie list Application in React JS
6. Creating a Progressive web application with Reusable React Components

Unit IV **11+6 Hours**

Django – Python web framework

Introduction to Django- History-Django Components-Alternate Components-MVC Architecture in Django MVC creation in Django – Configuring Django, Creating model, view and controller in Django, REST in Django and templates *MVC, Django, REST*

Practical Components

1. Creating models for database queries
2. Writing Django template for rendering data
3. Implementation of REST API using Django

Total: 48+32 Hours

Textbook (s)

1. Angular: Up and Running, Shyam Seshadri, 1st Edition, O'Reilly, 2018
2. Struts the Complete Reference, James Holmes, 2nd Edition, Mc. Graw Hill Professional, 2006.
3. Programming with Django, Wiley Publishing **Reference (s)**

1. Angular 6 for Enterprise-Ready Web Applications, Doguhan Uluca, 1st edition, 2018
2. The Definitive Guide to Django, Adrian Holovaty, Jacob Kaplan-Moss, Apress, 2009.
3. Struts 2 In Action, Donald Brown, Chad Michael Davis, Scott Stanlick, Dreamtech press, 2008.

21MLC32 Cloud Services using AWS

3 0 2 4

Course Outcomes

1. Comprehend different services provided AWS
2. Understand Identify and Access Management in AWS
3. Demonstrate AWS Directory services and AWS Artifact
4. Make use of Amazon Elastic Compute Cloud (EC2) to scalable computing capacity
5. Understand about Elastic File Systems and its features
6. Describe about AWS Storage Options

COs–POs Mapping

COs	PO2	PO3	PO4	PO5	PO12
1	2	3	2	1	2
2	3	2	1	2	3
3	3	3	2	3	1
4	2	1	3	1	2
5	3	3	1	3	2
6	1	2	3	2	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit-I **12+8 Hours**

Introduction to AWS:

Classic Data Center , Virtualization, Service Comparison: AWS, Azure, and GCP, Amazon Web Services (AWS) and its Benefits, AWS Global Infrastructure, Signup an AWS Free Tier Account.

Different Amazon Web Services, Ways to access AWS: CLI, Console, and SDKs, Explore Management Console and Configure AWS CLI

AWS CloudShell

Practical Components

1. Signup an AWS Free Tier Account
2. Explore Management Console and configure AWS CLI
3. Access AWS: CLI, Console, and SDKs
4. Working with AWS services in AWS CloudShell

Unit-II **12+8 Hours**

Identity and Access Management in AWS:

Identity and Access Management (IAM), Managing Users with IAM, Managing Permissions with Groups, IAM Policy and its Elements, IAM Roles ,Password Policy, Best Practices for IAM

AWS Security Token Services, AWS Single Sign-on, AWS Resources Access Manager, Active directory, Microsoft Active directory, AWS Directory Services, AWS Artifact, AWS Audit Manager, Accessing Billing and Alerts.

AWS Certificate Manager

Practical Components

1. Creating New Users, Roles, and Policies on AWS Console
2. Creating an user group for a collection of IAM users
3. Login to AWS Console via MFA
4. AWS Artifact

Unit-III **12+8 Hours**

Amazon Elastic Compute Cloud (EC2):

Amazon Elastic Compute Cloud (EC2) and Its Benefits, Amazon Machine Image (AMI), Security Groups in AWS, Authentication through Key-pair, Hardware Tenancy – Shared vs. Dedicated, Networking Layer in EC2: VPC, Elastic Network Interface (ENI) and Its Attributes

Public IP vs. Elastic IP, Instance Store

Elastic Block Store (EBS), Its Features and Volume Types

Solid State Drive: General Purpose SSD and Provisioned IOPS

Hard Disk Drive: Throughput Optimized HDD and Cold HDD
Snapshots, Elastic File System (EFS) and Its Features, Amazon Batch, Windows File Server *Amazon FSx for Windows File Server*

Practical Components

1. Creating an EC2 instance and Custom AMI
2. Host the Website Inside EC2 instance
3. Create an Elastic IP and attaching an EBS Volume Externally
4. Creating a Snapshot

Unit-IV **12+8 Hours**

AWS Storage Options:

Traditional Storage, Need to Move to Cloud Storage, Traditional vs. Cloud Storage Cost, Cloud Storage, Different Storage Options Available on AWS, Simple Storage Service (S3) and Its Components, Working of S3, Bucket Policy, Access Control List (ACL)

Cross-Region Replication (CRR) and Its Use Case, Amazon S3 Transfer Acceleration, Choice of Storage Classes on S3, Lifecycle Policy of S3 Bucket, AWS Backup.

CDN: CloudFront, Storage Gateway

Practical Components

1. Hosting a Static Website on Amazon S3
2. Replicating Data Across Regions
3. Transfer and Retrieve Data from Glacier
4. Accessing Website Using CloudFront

48+32 Hours

Textbook(s):

1. Amazon Web Services (AWS) 2022 Everything You Need To Know About The Amazon Web Service From Start To End, 2022, Maxwell Harris, Maxwell Harris.
2. Implementing AWS: Design, Build, and Manage your Infrastructure, Yohan Wadia, Rowan Udell, Lucas Chan, Udita Gupta, packt publishers.

21CS007 Cloud Computing Essentials (Elective III)

3 0 2 4

Course Outcomes

1. Interpret the architecture and infrastructure models of cloud computing, strengths, and limitations of cloud computing.
2. Understand the virtualization concepts of machines and data centers.
3. Infer the design concepts of cloud ready applications
4. Compare different cloud centre's implementation
5. Understand the concepts of cloud scaling and disaster recovery
6. Analyze the performance, scalability, and availability of the underlying cloud technologies and software

CO–PO Mapping

CO	PO2	PO5	PO6	PO7	PO8
1	3	3	2	3	1
2	3	1	1	2	3
3	2	2	3	1	2
4	2	1	3	3	1
5	2	1	3	1	2
6	2	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+6 Hours**

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges. System Models: Introduction, Architectural Models- Software Layers, System Architecture, Variations, Interface and Objects, Design Requirements for Distributed Architectures, Fundamental Models- Interaction Model, Failure Model, Security Model.

Communication between Distributed Objects- Object Model, Distributed Object Modal.

Practical Components:

1. Study of Cloud Computing & Architecture.
2. Install KVM/VMware Workstation for creating a virtual environment on windows platform. Concept: Virtualization
3. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.

Unit II **12+8 Hours**

Introduction to Cloud Computing

Overview of Computing Paradigm: Recent Trends in Computing, Evolution of Cloud Computing. Introduction to Cloud Computing: Cloud Computing (NIST Model), Properties, Characteristics & Disadvantages, Role of Open Standards. Cloud Computing Architecture: Cloud Computing Stack, Service Models (XaaS), Deployment Models. Infrastructure as a Service (IaaS): Introduction to IaaS, Resource Virtualization. Platform as a Service (PaaS): Introduction to PaaS, Cloud Platform and Management. Software as a Service (SaaS): Introduction to SaaS, Web services, Web 2.0, Web OS.

Companies in the Cloud Today, Amazon Web Servies, Google services, IBM Cloud, Windows Azure, Tata Cloud, Salesforce.com

Practical Components:

1. Install Google App Engine. Create hello world app and other simple web applications using python/java.
2. Use GAE launcher to launch the web applications.
3. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
4. Signup an AWS Free Tier Account and Explore Management Console and configure AWS CLI

Unit III **12+10 Hours**

Virtualization & Design

Virtualization, Virtual machine, Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Data centre, Virtualization for Data-Centre Automation. Service Levels for Cloud Applications Ready for the cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management.

various hypervisors like VMware, KVM, oracle VM, HTTPS,Electronic Payment

Practical Components:

1. Working on to transfer the files from one virtual machine to another virtual machine.
2. Working on to launch virtual machine using trystack (Online Openstack Demo Version)
3. Set up a pseudo-distributed, single-node Hadoop cluster backed by the Hadoop Distributed File System, running on Ubuntu Linux. After successful installation on one node, configuration of a multi-node Hadoop cluster(one master and multiple slaves).Hadoop file management: Adding files and directories, Retrieving files, Deleting files
4. Run Wordcount application using Hadoop single node cluster.
5. Move files between regular Linux file-system and HDFS using put and get commands **Unit IV**

Cloud Service Providers **12 + 8 Hours**

EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue, service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud.

Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform **Practical Components:**

1. Working and installation of Microsoft Azure
2. Working with Mangrasoft Aneka Software
3. Installation and Configuration of Justcloud
4. Working with AWS services in AWS CloudShell

Total: 48+32 Hours

Textbook (s)

1. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, Pearson Publication
2. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010
3. Cloud Computing: Principles and Paradigms, Editors: RajkumarBuyya, James Broberg, Andrzej M. Goscinski, Wile, 2011
4. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012
5. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010

Reference (s)

a) Michael Miller, Cloud Computing-Web Based Applications that change the way you work and collaborate online, 1st Edition, Pearson Education, Publishing, 2011

b) Kai Hwang, Geoffrey C Fox and Jack J.Dongarra, Distributed & Cloud Computing from Parallel Processing to the Internet of Things , 1st Edition, MK Publishing, 2010

c) David S Linthicum, Cloud Computing and SOA Convergence in Your Enterprise: A Step-by-Step Guide, 1st Edition, Addison-Wesley, 2009

d) Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, SubraKumaraswamy, ShahedLatif, O'Reilly, SPD, rp2011.

21CS008 Cryptography and Network Security (Elective III)

3 0 2 4

Course Outcomes

1. Explain the fundamentals of Cryptography, encryption and decryption algorithms
2. Make use of the symmetric and public key cryptographic algorithms 3. Choose the various authentication applications for security
4. Interpret the functionalities of IP and web security.
5. Demonstrates the functionalities of firewalls
6. Explain various non-cryptographic protocol vulnerabilities

CO–PO Mapping

COs	PO1	PO3	PO6	PO8
1	1	2	2	3
2	1	2	2	3
3	2	2	1	3
4	2	2	1	2
5	2	2	2	3
6	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12+10 Hours**

Introduction: Security Attacks, Security Goals, Computer criminals, Methods of Defence, Security Services, Security Mechanisms. **Basics of Cryptography:** Symmetric Cipher Model, Substitution Techniques, Transportation, Techniques, Other Cipher Properties- Confusion, Diffusion, Block and Stream Ciphers. Block Cipher Design Principles and Modes of Operations, **Symmetric Key Cryptosystems:** Principles of Private Key System, Data Encryption Standard (DES), Strength of DES, Triple DES, International Data Encryption Algorithm(IDEA), Advanced Encryption Standard (AES) *Blowfish, CAST-128.*

Practical Components:

1. Write a program that contains a string (char pointer) with a value “Hello world”. The program should XOR each character in this string with 0 and displays the result.
2. Write a program that contains a string (char pointer) with a value “Hello world”. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a program to perform encryption and decryption using the following algorithms a. Ceaser cipher
   b. Substitution cipher
4. Write a program to perform encryption and decryption using the following algorithms
   a. Mono-alphabetic cipher
   b. Hill Cipher
5. Write a program to perform encryption using the following algorithms
   a. Railfence Technique
   b. Transposition Technique

Unit II **12+8 Hours**

Public Key Cryptography: Principles of Public Key Cryptosystems, RSA Algorithm, Diffie-Hellman Key Exchange. Cryptographic Hash Functions: Principles of Cryptographic Hash functions, Applications of Cryptographic Hash Functions, Secure Hash Algorithm (SHA), Message Authentication Codes – Message Authentication Requirements and Functions, HMAC, Digital Signatures, Elgamal Digital Signature Schemes

Digital Signature Standards

Practical Components:

1. Write a program to implement the Play-fair Cipher algorithm.
2. Write a program to implement the DES algorithm logic.
3. Write a program to implement the Blowfish algorithm logic.

Unit III **12+8 Hours**

Authentication Applications: Kerberos, Key Management and Distribution, X.509 Directory Authentication service, Public Key Infrastructure, Electronic Mail Security: Pretty Good Privacy, S/MIME. **IP Security:** Overview, Architecture, Authentication Header, Encapsulating Security Payload, Combining security Associations, Internet Key Exchange,

Web Security: Web Security Considerations, Secure Sockets Layer and Transport Layer Security,.

HTTPS, Electronic Payment **Practical Components:**

1. Write a Java program to implement RSA algorithm.
2. Calculate the message digest of a text using the SHA-1 algorithm.
3. Write a program to implement the Triple-DES algorithm logic.
4. Write a program to implement the AES algorithm logic. **Unit IV**

Locking, Recovery Systems, Indexing, Different Types of Data 12 + 6 Hours

IDS and Firewalls: Intruders, Intrusion Detection, Password Management, Firewalls-Characteristics, Types of Firewalls, Placement of Firewalls, Firewall Configuration, Trusted Systems. Non-cryptographic protocol Vulnerabilities: DoS, DDoS, Session Hijacking and Spoofing, Software Vulnerabilities- Phishing, Buffer Overflow. *Format String Attacks, SQL InjectionCybercrime and Computer Crime, Intellectual Property.*

Practical Components:

1. Implement the Diffie-Hellman Key Exchange mechanism.
2. Write a program that can encrypt and decrypt using a general substitution block cipher.
3. Write a program that can encrypt and decrypt in 4-bit in cipher clock chaining mode using DES.

Total: 48+32Hours

Textbook (s)

1. William Stallings, “Cryptography And Network Security – Principles and Practices”, 7th edition, Pearson Education Limited 2017.
2. AtulKahate, "Cryptography and Network Security", 2nd edition Tata McGraw-Hill, 2003.
3. Behourz A Forouzan, Cryptography and Network Security, 2nd edition, Tata McGraw-Hill 2011

Reference (s)

1. Matt Bishop ,“Computer Security art and science ”, Second Edition, Pearson Education, 2002
2. Wade Trappe and Lawrence C. Washington, “Introduction to Cryptography with Coding Theory” Second Edition, Pearson Education, 2007
3. Jonathan Katz, and Yehuda Lindell, Introduction to Modern Cryptography, CRC Press, 2007
4. Douglas R. Stinson, “Cryptography Theory and Practice”, Third Edition, Chapman & Hall/CRC, 2006 5. Wenbo Mao, “Modern Cryptography – Theory and Practice”, Pearson Education, First Edition, 2006.

6. OWASP top ten security vulnerabilities: http://xml.coverpages.org/OWASPTopTen.pdf

21ML606 Deep Learning Techniques Lab

0 0 3 1.5

Course Outcomes

1. Design and implement the basic neural network model.
2. Design and develop the various convolutional Neural Network Models for classification problems.
3. Apply the transfer learning techniques on various standard DL models.
4. Demonstrate use of different RNN model for text processing.
5. Develop various CNN and RNN models for understanding visual and text data.
6. Understanding Generative and Transformers based neural network models.

CO–PO Mapping

COs	PO4	PO5	PO8	PSO1
1	3	3	2	2

2	3	3	2	2
3	3	3	2	3
4	2	3	3	3
5	2	3	2	3
6	1	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Lab Manual

1. Deep Learning Laboratory– AIML Department, GMRIT. **Textbook**

(s)

1. Bengio, Yoshua, Ian J. Goodfellow, and Aaron Courville. "Deep Learning." An MIT Press book in preparation. (2015).
2. Fundamentals of Deep Learning: Designing Next-generation Machine Intelligence Algorithms by Nicholas Locascio and Nikhil Buduma O'Reilly Media; 1 edition (June 29, 2017) 3. Simon S. Haykin, Neural Networks, Prentice Hall, 2nd edition

4. B. Yegnanrayana , "Artificial Neural Networks" , PHI.

Reference (s)

1. Franchois Chollet, Deep Learning with Python
2. Deep Learning: A Practitioner's Approach by Adam Gibson and Josh Patterson Shroff/O'Reilly; First edition (2017)
3. Python Deep Learning by Daniel Slater and Gianmario Spacagna, Packt Publishing; 2/e (January 16, 2019)
4. Bishop, C., M., Pattern Recognition and Machine Learning, Springer, 2006
5. Kevin P. Murphy.,MachineLearning:A Probabilistic Perspective

List of Experiments

Week 1: Introduction to Tensors and Various Operations on Tensors

Week 2: Building a Simple Perceptron for Binary Classification

Week3: Implementing a Multilayer Perceptron (MLP) for MNIST Classification

Week 4: Convolutional Neural Networks (CNN) for CIFAR-10 Classification

Week 5: Exploring CNN Model: VGG16

Week 6: Exploring CNN Model: VGG19

Week 7: Exploring AlexNet and Transfer Learning with Pretrained CNN Models (VGG-16)
Week 8: Object Detection with YOLOv5

Week 9: One-Hot Encoding of Words and Characters using Word Embedding

Week 10: Study the Construction and Working of Recurrent Neural Networks (RNNs)

Week 11: Exploring Long Short-Term Memory
(LSTM) Week 12: Exploring Gated Recurrent Unit
(GRU). Week 13: Explore Image Segmentation with
CNN

Week 14: Image Captioning using CNN and LSTM

Week 15: Exploring Generative Adversarial Networks (GANs)

Week 16: Transformer-based Models (e.g., BERT) for Text Classification

Augmented Experiments:
(Students shall opt any one of the Augmented Experiments in addition to the regular experiments)

1. Human Facial Recognition.
2. Stock Market Prediction.
3. Social Media.
4. Aerospace.
5. Defense.
6. Healthcare.
7. Signature Verification.
8. Handwriting Analysis
9. Speech Recognition
10. Weather Forecasting

21MPX01 Mini Project

0 0 3 1.5

Course Outcomes

1. Identify a contemporary engineering application to serve the society at large
2. Use engineering concepts and computational tools to get the desired solution
3. Justify the assembled/fabricated/developed products intended
4. Organize documents and present the project report articulating the applications of the concepts and ideas coherently
5. Demonstrate ethical and professional attributes during the project implementation
6. Execute the project in a collaborative environment

COs – POs Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
1	3	2	-	-	-	3	2	-	-	-	-	-
2	3	3	-	-	3	-	-	-	-	-	-	-
3	3	3	3	2	-	-	-	-	-	-	2	-
4	-	-	-	-	-	-	-	-	-	3		2
5	-	-	-	-	-	-	-	3	-	-	-	-
6	-	-	-	-	-	-	-		3	-	-	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Mini Project: The curriculum offers Mini Projects in two different forms viz: (i) Mini Project as a mandatory component in all lab courses (ii) 2 credit Mini Project during 5th or 6th semester. With respect to second one (ii) student will take mini project batch wise and the batches will be divided as similar to lab courses. The report will be evaluated by a committee as nominated by CoE constituted with internal & external panels

21ESX02 Employability Skills II

0 0 2 2

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs – POs Mapping

COs	PO1	PO2	PO5	PO 8	PO10	PO12
CO1					3	2
CO2				1	2	2
CO3	2	1		2		
CO4	2		2			
CO5	2		2			
CO6	2		2			

3–Strongly linked | 2–Moderately linked| 1–Weakly linked **Soft Skills:**

Sl No.	Topic & Content	No. of Periods
1.	**Resume (Recap):** Resume? Templates? Mistakes to be avoided in a Resume and Steps to be followed in preparing it.	**01**
2.	**Group Discussions (Recap) & Practice:** GD? Stages of a GD, Skills assessed in a GD, Blunders to be avoided, How to excel in a GD? Practice sessions and sharing Feedback. (Screening sample Videos)	**01**
3.	**Interview Skills:** Interview? Types of Interviews, Dos & Don'ts, Skills assessed in an Interview, Mistakes to be avoided, How to equip oneself to excel? How to handle the Typical Interview Questions? (with Examples)	**03**
4.	**Mock Interviews**: Practice sessions with Feedback.	**02**
5.	**Exercises related to Communication:** Email Writing, Voice Versant., etc.	**01**
	Total Periods	**08**

Quantitative Aptitude:

Sl. No.	Topic	No. of Periods
1.	Time and Distance	01
2.	Time and Distance	01
3.	Problems on Trains	01
4.	Problems on Trains	01
5.	Blood relations	01
6.	Ratio and Proportions	01
7.	Calendars	01
8.	Clocks	01
	Total Periods	**08**

Domain Specific: (Advanced Competitive Programming) **16 Hours**

Module-1:
Non-Linear Data Structures: Heaps - Maps- Trees- Graphs - Tries.

Practice Problems:

Week	**Topic**	**Practice Problem Links**
1	Searching	https://www.hackerearth.com/practice/algorithms/searching/linear-search/practiceproblems/
		https://www.codechef.com/practice/topics/binary-search
		https://leetcode.com/tag/binary-search/
		https://www.interviewbit.com/practice/#topics[]=binary-search
2	Sorting	https://www.codechef.com/practice/topics/sorting
		https://leetcode.com/tag/sorting/
		https://leetcode.com/tag/merge-sort/
		https://www.codechef.com/practice/topics/sorting
3	Heaps & Maps	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=heap
		https://www.interviewbit.com/courses/programming/heaps-and-maps
		https://workat.tech/problem-solving/topics/bst-heaps-and-map/practice
4	Trees - I	https://www.codechef.com/practice-old/tags/trees
		https://leetcode.com/problemset/?page=1&topicSlugs=binary-tree
		https://www.hackerearth.com/practice/data-structures/trees/binary-and-narytrees/practice-problems/
		https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=trees
5	Trees - II	https://www.codechef.com/practice-old/tags/tree-data-structure
		https://leetcode.com/problemset/?page=1&topicSlugs=binary-search-tree
6	Trees - III	https://leetcode.com/problemset/?page=1&topicSlugs=segment-tree
		https://www.interviewbit.com/courses/programming/tree-data-structure
7	Graphs - I	https://www.codechef.com/practice-old/tags/graphs
		https://www.hackerrank.com/domains/algorithms?filters%5Bsubdomains%5D%5B%5D=graph-theory
8	Graphs - II	https://leetcode.com/problemset/?page=1&topicSlugs=graph
		https://www.interviewbit.com/courses/programming/graph-data-structure-algorithms

Module-2:
Algorithm design Techniques: Backtracking – Dynamic Programming – Greedy Approach.

9	Backtracking - I	https://www.codechef.com/practice-old/tags/backtracking
		https://leetcode.com/tag/backtracking/
10	Backtracking - II	https://www.interviewbit.com/courses/programming/backtracking#p roblems
		https://workat.tech/problem-solving/topics/backtracking/practice
		https://www.codechef.com/practice-old/tags/dynamic-programming

11	Dynamic Programming - I	https://www.hackerrank.com/domains/algorithms?filters%5Bsubdomains%5D%5B%5D=dynamic-programming
		https://www.hackerearth.com/practice/algorithms/dynamicprogramming/introduction-to-dynamic-programming-1/practiceproblems/
12	Dynamic Programming - II	https://leetcode.com/problemset/?page=1&topicSlugs=dynamicprogramming
		https://www.interviewbit.com/courses/programming/dynamicprogramming#problems
13	Greedy Algorithms - I	https://www.codechef.com/practice-old/tags/greedy
		https://www.hackerrank.com/domains/algorithms?filters%5Bsubdomains%5D%5B%5D=greedy
		https://www.hackerearth.com/practice/algorithms/greedy/basics-ofgreedy-algorithms/practice-problems/
14	Greedy Algorithms - II	https://www.interviewbit.com/courses/programming/greedyalgorithm#problems
		https://leetcode.com/problemset/?page=1&topicSlugs=greedy
15	Tries - I	https://www.codechef.com/practice-old/tags/tries
		https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=trie
		https://www.hackerearth.com/practice/data-structures/advanceddata-structures/trie-keyword-tree/practice-problems/
16	Tries - II	https://leetcode.com/problemset/?page=1&topicSlugs=trie
		https://workat.tech/problem-solving/topics/string-and-tries/practice

References:

1. https://leetcode.com
2. https://www.codechef.com
3. https://www.hackerearth.com
4. https://www.interviewbit.com
5. https://www.hackerrank.com **21HSX12 CC & EC Activities II**

0 0 1 1

Course Outcomes

1. Interpret and present the abstractive technical information through an activity
2. Think critically in providing solutions to the generic and common problems
3. Demonstrate the creative thinking in dealing with liberal arts
4. Instill team sprit through active engagement with the peer
5. Develop programs of common interest having social impact
6. Empower the under privileged through motivational activities

COs – POs Mapping

COs	PO6	PO7	PO9	PO10
1	-	-	-	3
2	3	2	-	-
3	3	-	-	-
4	-	-	3	-
5	3	-	-	-
6	3		-	-

Co-Curricular and Extra Curricular (CCEC) Activities:

Students shall acquire 1 credit each in 2nd and 3rd years with the following scheme:
Scheme of evaluation for the CCEC activities:

- No. of slots in each Semester @ 2 slots every week : 24
- No. of Stream (2-CC + 1-EC) : 3
- No. of slots allotted for each stream : 8

Requirement for the award of 1- Credit

- Students shall choose at least two streams of events in each semester
- Students shall secure 75% attendance in each stream of events to obtain a certificate
- Students shall obtain 2 certificates of Participation in each semester.

The credits earned through these courses will be indicated in the grade sheet and will not beaken into account for CGPA calculation.

AUDIT COURSE

Course Outcomes

At the end of the Audit Course the students will be able to
CO1: Interpret the meaning of values and select their goals by self- Investigation based on personal values.
CO2: Interpret the major events and issues related to a period in Indian history.
CO3: Assess the benefits and limitations of science and its application in technological developments towards human welfare
CO4: Check the awareness regarding basic human rights and to uphold the dignity of every individual.
CO5: Assess the individual and group behaviour and understand the implications of organizational behaviour on the process of management.
CO6: Determine the appropriateness of various leadership styles and conflict management strategies used in organizations.

Cos-POs Mapping:

COs	PO12
CO1	3
CO2	3
CO3	3
CO4	2
CO5	3
CO6	2

7th Semester
Career Path III (ML Ops)

21MLC13 Conversational AI (Elective V)

3 0 0 3

Course Outcomes

1. Understand the fundamentals & building blocksof Natural Language Processing.
2. Understand the basic Vectorization approaches.
3. Understand the basics of Conversational AI.
4. Understand the Dialog Management and Modeling.
5. Apply different rule based dialog management.
6. Make use of End-to-End Neural Dialogue Systems.

CO–PO Mapping

COs	PO1	PO2	PO4	PO12	PSO1
1	3	2	1	1	1
2	2	3	1	3	1
3	3	2	2	3	2
4	2	3	1	3	3
5	1	3	2	3	3
6	1	2	3	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to NLP:

Definition, History, Building blocks of language, Approaches to NLP, NLP Pipeline, Deep learning in NLP and Transformers in NLP.

Basic Vectorization approaches- One-Hot Encoding, Bag of Words, Bag of N-Gram, TF-IDF; Neural language models, N-gram language model Sequence labeling for POS and Named Entities: POS tagging, Named Entities tagging.

Evaluation of Named Entity Recognition, Markov chains.

Unit II **12 Hours**

Introduction to Conversational AI: Introduction to AI assistants and their platforms: Types of AI assistants and their platforms, Primary use cases for AI assistant technology: self-service assistant, agent assist, and classification and routing.

Building your first conversational AI: Building a conversational AI for Fictitious Inc, What's the user's intent and Responding to the user.

The applications of Conversational AI and Objective of Chatbots.

Unit-III **12 Hours**

Dialog Management and Modelling

Introducing Dialogue Systems: Dialogue System, History, Present-day Dialogue System, Modelling conversation in dialogue systems.

Rule-Based Dialogue Systems: A typical dialogue systems architecture and Tools for developing dialogue systems and Evaluating Dialogue Systems

Challenges of the dialogue systems, Conversational Agents and Multimodal dialogue systems.

Unit IV **12 Hours**

End-to-End Neural Dialogue Systems

Neural Network Approaches to Dialogue Modelling, A Neural Conversational Model, Introduction to the Technology of Neural Dialogue and Open-Domain Neural Dialogue Systems.

Case Study: Chatbots in healthcare and mental health support, Voice-enabled devices and smart home applications.

Objective of Conversational AI with Rasa and Design Principles of the Chatbot.

Total: 48 Hours

Textbook (s)

1. Andrew R. Freed. Conversational AI: Chatbots that work, Manning; 1st edition (12 October 2021)
2. Yoav Goldberg. Neural Network Methods for Natural Language Processing, Morgan and Claypool Life Sciences, 30 April 2017
3. Michael McTear . Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots (Synthesis Lectures on Human Language Technologies), Morgan & Claypool Publishers, 30 October 2020
4. Dan Jurafsky and James H. Martin. Speech and Language Processing (3rd ed. draft), Pearson, 22 Nov 2023.

Reference (s)

1. Xiaoquan Kong , Guan Wang . Conversational AI with Rasa by Packt., 1st edition, Oct 2021.
2. Stephan Bisser . Microsoft Conversational AI Platform for Developers End-to-End Chatbot Development from Planning to Deployment.1st edition, 17 Feb 2021.
3. Lee Boonstra . The Definitive Guide to Conversational AI with Dialog flow and Google Cloud build advanced enterprise chatbots, voice., 1st edition, 24 June 2021.
4. https://hao-cheng.github.io/ee596_spr2019/

Career Path II (Full Stack Developer)

21CSC23 Web Application Databases (Elective V)

3 0 0 3

Course Outcomes

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

1. Understand how web-based client-server applications work.
2. Analyze architecture of various web-based applications.
3. Examine the requirements of structured and un-structured data.
4. Identify the structure of modern-day web applications.
5. Identify the use advanced querying techniques to enable faster data transfer.
6. Analyze the security issues in data transfer via the web.

CO – PO Mapping

COs	PO1	PO3	PO5	PSO2
1	3	3	1	1
2	3	2	2	2
3	3	3	3	3
4	3	3	2	3
5	3	2	2	3
6	2	3	3	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Syllabus

Unit I **12 Hours**

Overview of Web databases

Web application Overview: Basic concepts of web, Web database architectures, Web database gateways, Web-database programming: Client-side and server-side web database programming *Database gateways, CGI, Browser extensions*

Unit II **12 Hours**

Structured and unstructured Web databases

Structured databases: Structured data, Data models, Relational database management systems, MySQL – Introduction, querying, updating and deleting data, Challenges of structured data.

Unstructured databases: Unstructured data and its usages, NoSQL databases, MongoDB- Introduction, querying, updating and deleting data, Challenges of unstructured data- Cassandra, GraphQL *Structured data, unstructured data, Relational database, Document-based database*

Unit III **12 Hours**

Database connections & data operations

Connecting to database: Native database APIs, Database-independent APIs, Template-driven database access packages, Third-party class libraries.

CRUD operations: Querying web database, User driven querying, writing to web databases – create, delete and update. *JDBC, ODBC, template parsing.*

Unit IV **12 Hours**

Authentication, Authorization and Communication in web services

Web requests: Structure of web requests, JavaScript Object Notation (JSON), Request methods – GET, POST, PUT, DELETE, response status codes
State management: Session management, cookies, request and response headers.
Security: User authentication, User authorisation, Proxy servers, Digital signatures, Digital certificates, SSL and HTTPS.
Kafka, RabbitMQ
JSON, User authentication and authorisation

Total: 48 Hours

Textbook (s)

1. Web Database Applications with PHP and MySQL, Hugh E. Williams, David Lane, 2nd Edition, O'Reilly, 2004.
2. NoSQL Distilled, PramodSadalage, Martin Fowler, 1st Edition, 2012

Reference (s)

1. Getting Started with NoSQL, GauravVaish, Paperback, Packt, 2013.
2. Fundamentals of Database Systems, ElmasriNavathe Pearson Education. 6th edition, 01 Jan 2013.
3. An Introduction to Database systems, C.J. Date, A.Kannan, S.SwamiNadhan, Pearson, 8th Edition.2004.
4. Web Application Security, Andrew Hoffman, O'Reilly, 2020.

Career Path III (Cloud Computing)

21MLC33 Cloud Security Essentials (Elective V)

3 0 0 3

Course Outcomes

1. Understand core concepts of the cloud computing paradigm
2. Learn about characteristics, advantages and challenges brought by the various models and services in cloud computing.
3. Identify the known threats, risks, vulnerabilities and privacy issues associated with Cloud based services.
4. Understand security challenges, threats and risks involved in the cloud application
5. Anlayze the Security Requirements for the Architecture in cloud services
6. Understand Data Security and best practices involved in securing the cloud.

CO–PO Mapping

CO	PO!	PO2	PO3	PO12
1	1	3	1	1
2	1	2	2	1
3	1	2	1	1
4	1	3	2	1
5	1	2	3	1
6	1	3	2	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Cloud Computing Software Security Fundamentals- Cloud Information Security Objectives, Cloud Security Services, Relevant Cloud Security Design Principles,
Secure Cloud Software Requirements, Security Concerns, Risk Tolerance, Legal and Regulatory Issues

Unit II **12 Hours**

Cloud Computing Risk Issues and Security Challenges:
The CIA Triad,Privacy and Compliance Risks,Threats to Infrastructure, Data, and Access Control, Cloud Service Provider Risks, Disaster Recovery-Disaster Recovery Planning, Disasters in cloud, Cloud Disaster Management. Security Challenges- Security Policy Implementation, Virtualization Security Management, VM Security Recommendations, VM-Specific Security Techniques.

Unit III **12 Hours**

Securing the cloud: Architecture and Data Security
Security Requirements for the Architecture, Security Patterns and Architectural Elements, Cloud Security Architecture, Planning Key Strategies for Secure Operation.

Unit IV **12 Hours**

Overview of Data Security in Cloud Computing, Data Encryption: Applications and Limits, Cloud Data Security: Sensitive Data Categorization, Cloud Data Storage, Cloud Lock-in, Key strategies to secure the cloud, Best practices for cloud computing, security monitoring

Total: 48 Hours

Textbook (s)

1. Russell Dean Vines and Ronald L. Krutz, Cloud Security: A Comprehensive Guide To Secure Cloud Computing, Wiley India Pvt Ltd, 2010
2. Vic Winkler (J.R.), Securing The Cloud: Cloud Computing Security Techniques and Tactics, Syngress/Elsevier ,2011

Reference (s)

1. Thomas Erl, Cloud Computing Design Patterns, Prentice Hall, 2015
2. Barrie Sosinsky, Cloud Computing Bible, Wiley India, 2011

3. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011

21IT010 Social Network Analysis (Elective V)

3 0 0 3

Course Outcomes

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

1. Acquire Knowledge to analyze Social Networks
2. Model, Aggregate and Represent Knowledge for Semantic Web
3. Use Extraction and Mining tools for Social Networks
4. Apply Reality Mining to Predict Human Behaviors for Social Communities
5. Apply various Algorithms for Evolution and Opinion Mining in Social Networks 6. Write algorithms and systems for expert location in social networks

CO – PO Mapping

COs	PO1	PO2	PO4	PO5
1	2	2	2	3
2	2	2	2	3
3	2	2	1	3
4	2	2	1	2
5	2	2	2	3
6	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Syllabus

Unit I **11 Hours**

Introduction

Introduction to Web, Limitations of current Web, Development of Semantic Web, Emergence of the Social Web, Statistical Properties of Social Networks, Network analysis, Development of Social Network Analysis-Key concepts and measures in network analysis, Discussion networks-Blogs and online communities-Web-based networks *Case Studies of Social Network sites like Facebook-Twitter-LinkedIn etc.*

Unit II **13 Hours**

Evolution

Evolution in Social Networks– Framework, Tracing Smoothly Evolving Communities, Models and Algorithms for Social Influence Analysis, Influence Related Statistics, Social Similarity and Influence, Influence Maximization in Viral Marketing, Link Prediction in Social Networks, Feature based Link Prediction *Neurons &Social Actor-Advanced techniques in Link Prediction*

Unit III **12 Hours**

Mining Communities and Opinion Mining

Applications of Community Mining Algorithms, Node Classification in Social Networks

Opinion Extraction–Sentiment Classification and Clustering, Temporal Sentiment Analysis-Irony Detection in Opinion Mining-Wish Analysis–Product Review Mining–Review Classification *Tracking Sentiments towards Topics over Time*

Unit IV **12 Hours**

Modelling and Visualization

Visualizing Online Social Networks, A Taxonomy of Visualizations, Graph Representation-Centrality-ClusteringNode-Edge Diagrams-Visualizing Social Networks with Matrix-Based Representations-Node-Link

Diagrams, Hybrid Representations, Modelling and Aggregating Social Network Data, Random Walks and their Applications, Ontological representation of Social Individuals and Relationships.

Use of Hadoop and MapReduce-Gephi **12 Hours**

Total: 48 Hours

Textbook (s)

1. Charu C. Aggarwal, Social Network Data Analytics, Springer, 2011
2. Peter Mika, Social Networks and the Semantic Web, Springer, 1st Edition, 2007. **Reference (s)**

1. BorkoFurht, Handbook of Social Network Technologies and Applications, Springer, 1st Edition, 2010.
2. GuandongXu, Yanchun Zhang and Lin Li, Web Mining and Social Networking, Techniques and applications, Springer, 1st Edition, 2011.
3. Giles, Mark Smith, John Yen, Advances in Social Network Mining and Analysis, Springer,2010.
4. Ajith Abraham, Aboul Ella Hassanien, VáclavSnášel, Computational Social Network Analysis: Trends, Tools and Research Advances, Springer, 2009.

21ML001 Human Computer Interaction (Elective V)

3 0 0 3

Course Outcomes

1. Demonstrate the capabilities of both humans and computers from the viewpoint of human information processing.
2. Interpret typical human–computer interaction (HCI) models, styles, and various historic HCI paradigms. 3. Choose an interactive design process and universal design principles to designing HCI systems
4. Make use of HCI design principles, standards and guidelines.
5. Examine and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems
6. Categorize and discuss HCI issues in groupware, ubiquitous computing, virtual reality, multimedia, and Word Wide Web-related environments.

COs–POs Mapping

COs	PO1	PO2	PO3	PO12
1	1	1	1	1
2	1	1	1	1
3	2	2	2	1
4	2	1	2	1
5	2	2	1	1
6	2	2	2	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction

Importance of user Interface-definition, importance of good design, Benefits of good design, A brief history of Screen design, The graphical user interface-popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user-Interface popularity, characteristics-Principles of user interface.

Unit II **12 Hours**

Design process

Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions, Screen Designing:-Design goals-Screen planning and purpose, organizing screen elements, ordering of screen data and content-screen navigation and flow-Visually pleasing composition amount of information-focus and emphasis-presentation information simply and meaningfully-information retrieval on web-statistical graphics-Technological consideration in interface design.

Unit III **12 Hours**

Windows

New and Navigation schemes selection of window, selection of devices based and screen based controls, Components-text and messages, Icons and increases-Multimedia, uses problems, choosing colors.

Unit IV **12 Hours**

Software tools &Interaction Devices

Specification methods, interface-Building Tools, Keyboard and function keys, pointing devices-speech recognition digitization and generation-image and video displays-drivers.

Total: 48 Hours

Textbook (s)

1. Wilbert O Galitz, Wiley Dream Tech, The essential guide to user interface design, 3rd Edition, Wiley Computer Publishing, 2007
2. Ben Shneidermann, Designing the user interface, 3rd Edition, Pearson Education Asia, 2008

Reference (s)

1. Alan Dix, Janet Fincay, GreGoryd, Abowd, Russell Bealg, Human Computer Interaction., 4th Edition, Pearson Education, 2018

21CS012 Wireless Adhoc Network (Elective-V)

3 0 0 3

Course Outcomes

1. Understand the various challenging issues of wireless Adhoc networks.
2. Apply the MAC protocols using reservation and scheduling mechanisms.
3. Apply the routing protocols and transport layer protocol using various TCS mechanisms in wireless Adhoc networks.
4. Understand the issues of MAC and network layer solution to provide quality of service.
5. Understand the significances of various battery management schemes for wireless Adhoc networks.
6. Analyze the security requirements and secure routing in wireless Adhoc networks.

COs–POs Mapping

COs	PO1	PO2	PO3	PO12
1	1	2	3	1
2	1	3	2	1
3	2	3	3	1
4	2	2	3	1
5	2	1	3	1
6	2	2	3	1

3–Stronglylinked|2–Moderatelylinked|1–Weaklylinked

UNIT I **12 Hours**

Wireless AdHoc Networks: Introduction, Properties, applications, limitations, Issues in AdHoc Wireless Networks, Ad Hoc Wireless Internet. **MAC Protocols:** Introduction, Issues in Designing a MAC protocol for AdHoc Wireless Networks, Design goals of a MAC Protocol for Ad Hoc Wireless Networks, Classifications of MACProtocols, Contention-Based Protocols, Contention-Based Protocols with reservation Mechanisms.

Contention–Based MAC Protocols with Scheduling Mechanisms

UNIT II **12 Hours**

Routing Protocols: Introduction, Issues in Designing a Routing Protocol for AdHoc Wireless Networks, Classification of Routing Protocols, Proactive/ Table–Driven Routing Protocols, Reactive/ On–Demand Routing Protocols, Hybrid Routing Protocols, Hierarchical Routing Protocols, Power – Aware Routing Protocols. **Transport Layer:** Introduction, Issues in Designing a Transport Layer Protocol for Ad Hoc Wireless Networks, Design Goals of a Transport Layer Protocol for Ad Hoc Wireless Networks, Classification of Transport Layer Solutions, TCP Over AdHoc Wireless Networks.

Other Transport Layer Protocol for AdHoc Wireless Networks.

UNIT III **12 Hours**

Quality of Service: Introduction, Issues and Challenges in Providing QoS in AdHoc Wireless Networks, Classification of QoS Solutions, MAC Layer Solutions, Network Layer Solutions, QoS Frameworks for Ad HocWirelessNetworks.**EnergyManagement:**Introduction,NeedforEnergyManagementinAdHocWirelessNetworks,ClassificationofAdHocWirelessNetworks,BatteryManagementSchemes,TransmissionPowerManagementSchemes.

System Power Management Schemes.

UNIT IV **12 Hours**

Security Protocols: Network Security Requirements, Issues and Challenges in Security Provisioning, Network Security Attacks, Key Management, Secure Routing in AdHoc Wireless Networks.

Wireless Sensor Networks: Introduction, Sensor Network Architecture, Data Dissemination, Data Gathering, Location Discovery.

Quality of a Sensor Network, Evolving Standards, Other Issues

Total: 48 Hours

Textbook(s)

1. AdHoc Wireless Networks: Architectures and Protocols-C.Siva Ram Murthy and B.S.Manoj,2004,PHI.
2. Wireless Ad-hoc and Sensor Networks: Protocols, Performance and Control Jagannathan Sarangapani, CRC Press, Taylor & Francis, 1 January 2007.

Reference(s)

1. Wireless Sensor Networks-C.S.Raghavendra, Krishna.M. Sivalingam, 2004, Springer.
2. Ad-HocMobileWirelessNetworks:Protocols&Systems,C.K.Toh,Firsted.PearsonEducation, 1st edition, December 2001.

21ML002 Large Language Models (Elective VI)

3 0 0 3

Course Outcomes

1. Understand the fundamentals & building blocks of LLMs
2. Exemplify basic data processing techniques.
3. Understand prompt engineering mechanism.
4. Understand the properties of LLMs.
5. Evaluate the performance f various LLMs.
6. Elaborate various case studies of LLMs.

CO–PO Mapping

COs	PO1	PO2	PO4	PO12	PSO1
1	3	2	1	1	1
2	2	3	1	3	1
3	3	2	2	3	2
4	2	3	1	3	3
5	1	3	2	3	3
6	1	2	3	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Introduction to Language Models, History of Language models, Inspiration to LLM, Understanding LLMs : Human Thought Versus LLM Processing, Transformer architecture: Understanding the Architecture , Key Components of the Transformer Model and Practical Applications of Transformers.

Unit II **12 Hours**

Dataset preparation - probabilistic tokenization, Training - RLHF, instruction tuning, MoE, context window, Multimodality of LLMs. Prompt Engineering : Definition of prompt and prompt engineering, Need of Prompt Engineering, Elements of a Prompt, Designing Prompts for Different Tasks, Prompt Engineering Techniques, Risks.

Unit III **12 Hours**

Properties of LLM - Scaling laws, Emergent abilities, Interpretation of LLMs, Evaluation of performance - Perplexity, BPW, BCT & BPT, Task specific benchmarks

Unit IV **12 Hours**

Case studies of popular LLMs - GPT-1/2/3/3.5, BERT/PaLM/Gemini, OPT/LlaMA

Total : 48 Hours

Articles / Reading Materials

1. "Better Language Models and Their Implications". OpenAI. 2019-02-14. Archived from the original on 2020-12-19. Retrieved 2019-08-25.
2. Vaswani, Ashish; Shazeer, Noam; Parmar, Niki; Uszkoreit, Jakob; Jones, Llion; Gomez, Aidan N; Kaiser, Łukasz; Polosukhin, Illia (2017). "Attention is All you Need" (PDF). Advances in Neural Information Processing Systems. Curran Associates, Inc. 30.
3. Merritt, Rick (2022-03-25). "What Is a Transformer Model?". NVIDIA Blog. Retrieved 2023-07-25.
4. Rogers, Anna; Kovaleva, Olga; Rumshisky, Anna (2020). "A Primer in BERTology: What We Know About How BERT Works". Transactions of the Association for Computational Linguistics. 8: 842–866.

arXiv:2002.12327. doi:10.1162/tacl_a_00349. S2CID 211532403.

5. Huyen, Chip (18 October 2019). "Evaluation Metrics for Language Modeling". The Gradient.
6. Wiggers, Kyle (28 April 2022). "The emerging types of language models and why they matter". TechCrunch.

21CS015 SOFTWARE PROJECT MANAGEMENT (Elective VI)

3 0 0 3

Course Outcomes

1. Explain the concepts and issues of Software Project Management
2. Illustrate various Software Architectures
3. Classify Software Risks and Risk Management Strategies
4. Design effective software development model to meet organizational needs
5. Experiment with appropriate methodologies to develop a project schedule
6. Experiment with appropriate techniques to assess ongoing project performance

CO–PO Mapping

COs	PO1	PO2	PO3	PO11	PO12	PSO2
1	3	2	2	2	2	2
2	3	2	2	2	2	3
3	3	2	2	2	2	3
4	3	2	3	2	2	3
5	3	2	3	3	2	3
6	3	2	3	3	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 Hours**

Conventional Software Management: The Waterfall Model, Conventional Software Management Performance. **Transition:** The principles of conventional software Engineering, Principles of Modern software management, transitioning to an iterative process.

Evolution of Software Economics: Software Economics, Software Cost Estimation. **Improving Software Economics:** Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation, Achieving Required Quality, Peer Inspections.

Software Development Phases, Procedural vs Object Oriented Paradigm.

Unit II **12 Hours** **Life**

Cycle Phases: Engineering and Production Stages, Inception, Elaboration, Construction, Transition Phases.

Artifacts of Software Process: The Artifact Sets, Types of Artifacts.

Software Architectures: A Management Perspective and Technical Perspective, **Workflows of the process**: Software Process Workflows, Iteration Workflows.

Off-the-Shelf Components, Requirement Specification Document, Methods of Specification.

Unit III **12 Hours**

Checkpoints of the Process: Major Milestones, Minor Milestones, Periodic Status Assessments.

Iterative Process Planning: Work breakdown Structures, Planning Guidelines, Cost and Schedule Estimating process, Iteration Planning Process.

Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, Evolution of Organizations.

Process Automation: Automation Building Blocks, The Project Environment.

CASE Tools for Requirements Specification, Planning and Estimation, Pragmatic Planning.

Unit IV **12 Hours**

Project control and Process Instrumentation: The Seven Core Metrics, Management Indicators, Quality Indicators, Life Cycle Expectations, Pragmatic Software Metrics, Metrics Automation, Tailoring the Process.

Future Software Project Management: Modern Project Profiles, Next Generation Software Economics,The COCOMO Cost Estimation Model: Basic COCOMO

Bugs of testing, Bug tracking tools

Total: 48 Hours

Textbook (s)

1. Walker Royce, “Software Project Management – A Unified Framework”, 1stEdition, Pearson Education, 2005.
2. PankajJalote, “Software Project Management in Practice”, 1stEdition, Pearson Education, 2002.

Reference (s)

1. Bob Hughes, “Mike Cotterell, Rajib Mall, Software Project Management”, 5thEdition, McGraw-Hill Higher Education, 2011.
2. Joel Henry, “Software Project Management”, 1st Edition, Pearson Education, 2004.
3. Norman E. Fenton, Shari Lawrence Pfleeger, “Software Metrics: A Rigorous and Practical Approach “, 1st Edition, PWS Publishing Company, 1997.

21ML002 Reinforcement Learning (Elective VI)

Course Outcomes

The student will be able to

1. Understand the key principles and terminology in reinforcement learning
2. Understand how real-world problems can be structured as Markov Decision Processes
3. Describe the use of dynamic programming in reinforcement learning
4. Apply model-free reinforcement learning methods in reinforcement learning tasks.
5. Explain the concepts behind policy gradient methods and deep learning-based approaches in reinforcement learning.
6. Identify various advanced reinforcement learning methods and summarize their applications in different fields.

CO–PO Mapping

COs	PO6	PO7	PSO1	PSO2
1	2	1	1	1
2	2	2	1	1
3	1	1	1	1
4	1	3	2	1
5	1	3	2	1
6	1	3	2	1

Unit 1: Introduction to Reinforcement Learning and Markov Decision Processes **12 Hours**

Introduction to RL terminology: Agent, Environment, Action, State, Reward, Episode -Taxonomy of RL methods - Reinforcement Learning Framework: Agent-environment interface, Goals and rewards. Markov Decision Processes (MDPs): Understanding states, actions, and rewards, Episodes in RL, Markov property - Dynamics and Returns: Transition probabilities and dynamics of MDPs, Calculation of returns.

Real world applications of reinforcement learning.

Unit 2: Dynamic Programming and Monte Carlo Methods **12 Hours**

Policy and Value Functions - Bellman equations for value functions -Dynamic Programming in RL: Policy iteration and value iteration, Limitations of dynamic programming - Monte Carlo Methods in RL: Monte Carlo Policy Evaluation, Monte Carlo Control, Control without Exploring Starts, Importance Sampling in Policy Prediction, Incremental Implementation, Policy Monte Carlo Control.

Eligibility Traces

Unit 3: Model-Free Control Methods **12 Hours**

Temporal Difference Learning: TD learning Algorithms: TD(0) and TD(λ) algorithms. Model-Free Control Algorithms: SARSA, Q-learning, Double Q-learning, n-step Bootstrapping methods - Exploration vs. exploitation strategies: Exploration Strategies in RL, Exploration-exploitation trade-off.

Function approximation, hierarchical reinforcement learning

Unit 4: Advanced Algorithms and Strategies **12 Hours**

Deep Reinforcement Learning: Introduction to Deep Q-Networks (DQN), Experience Replay, Target Networks - Policy Gradient Methods: Basics of policy gradients, REINFORCE algorithm - Actor-Critic Methods: Combining policy gradients with value functions, Actor-critic algorithms, Function Approximation in Actor-Critic - Proximal Policy Optimization (PPO).

Policy Approximation, Multi-Agent Reinforcement Learning

Total: 48 Hours

Textbooks:

1. Richard S. Sutton and Andrew G. Barto, "Reinforcement Learning: An Introduction," Second Edition, The MIT Press, 2019.

2. Marco Wiering and Martijn Van Otterlo, "Reinforcement Learning: State-of-the-Art," Adaptation, Learning, and Optimization, Vol. 12, Springer, 2012.
3. Maxim Lapan, "Deep Reinforcement Learning Hands-On," Second Edition, Packt Publishing, 2018.

Reference Books:

1. Csaba Szepesvári, "Algorithms for Reinforcement Learning," Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool Publishers, 2009.
2. Ian Goodfellow, Yoshua Bengio, and Aaron Courville, "Deep Learning," MIT Press, 2016.
3. Yuxi Li, "Reinforcement Learning Applications", arXiv preprint arXiv:1908.06973v1, 2019.

21SIX02 Summer Internship II

0 0 0 1.5

Course Outcomes

At the end of the summer internship students will be able to

1. Demonstrate communication skills to meet the requirement of industry
2. Develop logical thinking and analytical skills to thrive in competitive examinations
3. Use mathematical concepts to solve technical quizzes
4. Develop technical skills to work out real time problems
5. Develop algorithms for different applications
6. Solve industry defined problems using appropriate programming skills

CO–PO Mapping

COs	PO1	PO2	PO5	PO6	PO10	PO12
CO1	-	-	-	-	3	-
CO2	3	1	-	-	-	-
CO3	3	-	3	-	-	2
CO4	3	1	-	-	-	-
CO5	3	1	3	3	-	1
CO6	3	1	3	-	-	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

21PWX01Project Work

0 0 16 8

Course Outcomes

At the end of the project work the students will be able to

7. Identify a contemporary engineering application to serve the society at large 8. Use engineering concepts and computational tools to get the desired solution
9. Justify the assembled/fabricated/developed products intended.
10. Organize documents and present the project report articulating the applications of the concepts and ideas coherently
11. Demonstrate ethical and professional attributes during the project implementation.
12. Execute the project in a collaborative environment.

CO–PO Mapping

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	-	-	-	3	2	-	-	-	-	-
CO2	3	3	-	-	3	-	-	-	-	-	-	-
CO3	3	3	3	2	-	-	-	-	-	-	2	-
CO4	-	-	-	-	-	-	-	-	-	3		2
CO5	-	-	-	-	-	-	-	3	-	-	-	-
CO6	-	-	-	-	-	-	-		3	-	-	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

8th Semester

21CS018 Information Retrieval Systems (Elective VIII)

0 0 0 3

Course Outcomes

1. Explain the Objectives of Information Retrieval Systems and its relation to Digital Libraries and Data warehouse
2. Illustrate various System Search Browse and Miscellaneous Capabilities
3. Explain Automatic Indexing, Information Extraction methods
4. Demonstrate Inverted file structures, N-gram data structure, and PAT data structure
5. Interpret Classical techniques of Information Retrieval, and additional techniques employed by Web search engines
6. Learn to develop a small information retrieval system by using user search techniques and text search Algorithms

CO–PO Mapping

COs	PO1	PO2	PO3	PO4
1	3	2	2	2
2	3	2	2	2
3	2	3	1	1
4	2	3	1	1
5	2	3	1	1
6	3	3	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **11 Hours**

Introduction: Definition, Objectives, Functional Overview, Relationship to DBMS, Digital libraries and Data Warehouses.

Information Retrieval System Capabilities: Search, Browse, Miscellaneous. *Miscellaneous capabilities.*

Unit II **13 Hours**

Cataloging and Indexing: Objectives, Indexing Process, Automatic Indexing, Information Extraction.

Data Structures: Introduction, Stemming Algorithms, Inverted file structures, N-gram data structure, PAT data structure, Signature file structure, Hypertext data structure. *Signature file structure, Hypertext data structure.*

Unit III **12 Hours**

Automatic Indexing: Classes of automatic indexing, Statistical indexing, Natural language, Concept indexing, Hypertext linkages

Document and Term Clustering: Introduction, Thesaurus generation, Item clustering, Hierarchy of clusters. *Measurement Example-TREC Results*.

Unit IV **12 Hours**

User Search Techniques: Search statements and binding, Similarity measures and ranking, Relevance feedback, Selective dissemination of information search, Searching the Internet and hypertext, Information Visualization **Text Search Algorithms:** Introduction, Software text search algorithms, Hardware text search systems.

Hardware text search systems

Total: 48 Hours

Textbook (s)

1. Kowalski, Gerald, Mark T Maybury: Information Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 1997.
2. http://nlp.stanford.edu/IR-book/html/htmledition/evaluation-in-information-retrieval-1.html

Reference (s)

1. Frakes, W.B. and RicardoBaeza Yates, Information Retrieval Data Structures and Algorithms, 1st Edition Prentice Hall, 1992.
2. Robert Korfhage, Information Storage & Retrieval, 1st Edition, John Wiley & Sons, 2005

21CS019 Fundamentals of DevOps (Elective VIII)

0 0 0 3

Course Outcomes:

1. Illustrate the need of Improvement and value of DevOps
2. Outline the Value Stream using DevOps
3. Describe the Organizational Change and transformation
4. Illustrate the Concept and Goal of Accelerate Flow
5. Outline Feedback Loops and its usage
6. Demonstrate the Concept and Goal of Learning

CO-PO Mapping:

COs	PO1	PO3	PO5	PO8	PO10
1	3	3	3	3	3
2	2	2	2	2	2
3	1	2	1	1	2
4	3	2	2	1	2
5	3	3	3	3	2
6	2	3	2	1	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **12 Hours**

Introduction to DevOps: Understanding Improvement, The Convergence, History, and Value of DevOps; **Understanding the Value Stream using DevOps:** Analyzing the technology Value Stream; The ways of DevOps: Flow, Feedback Loops, Culture of Continual Experimentation and Learning;

UNIT II **12 Hours**

Value Stream in DevOps: Concept and Goals of Streams, Picking a Value Stream, Understanding Organizational Change, Enabling Transformation;

UNIT III **13 Hours**

Accelerate Flow: Concept and Goal of Accelerate Flow, Continuous Delivery Patterns and Practices, the Deployment Pipeline, Creating Consistency in the Pipeline, Automated Testing, Deployment Strategies; **Amplifying the Feedback Loops:** Concept and Goals of Feedback Loops, Creating a Service Reliability Culture, Fast Feedback, Understanding Monitoring, Understanding Complexity

UNIT IV **11 Hours**

Learning Acceleration with DevOps: Concept and Goal of Learning, Learning Organizations, Communication, Blameless Culture;

Total: 48 Hours

Text Books:

1. Kim, G., Behr, K., and Spafford, G. (2013). The Phoenix Project: A Novel About IT, DevOps, and Helping Your Business Win. IT Revolution Press
2. Kim, G., Humble, J., Debois, P., and Willis, J. (2016). The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations. IT Revolution Press

Reference Books:

1. Julian Fish ,The Practical Guide to Enterprise DevOps and Continuous Delivery, Addison-Wesley Professional, February 2017.

21DS003 Cyber Security (Elective VIII)

0 0 0 3

Course Outcomes

1. Explain the fundamental concepts of Cyber security
2. Demonstrate the web security and different attacks 3. Identify different network scanning and security measures
4. List out different types of Intrusion detection.
5. Model different types of Intrusion prevention systems
6. Outline different cyber-crimes, IT laws and acts.

CO–PO Mapping

COs	PO1	PO3	PO4	PO5	PSO1
1	3	3	1	1	1
2	3	3	1	3	1
3	1	2	3	3	2
4	3	1	1	3	3
5	1	3	3	3	3
6	1	3	2	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I

Introduction to Cyber Security **12 Hours**

What is Cyber Security, its need, cyber-threats, Cyber Warfare-Cyber Crime-Cyber terrorism-Cyber Espionage), Career Growth, Statistics, Inferences, Need for a Comprehensive Cyber Security Policy, Classification of Cyber Crimes, kinds of cybercrimes, Reasons for Cyber Crimes, Cyber Security Tools : Nmap, Metasploit, wireshark, tcpdump, snort.

Cyber security awareness, social engineering, cyber stalking

Unit II

Web security **12 Hours**

Same origin Policy, Cross Origin Resource Sharing, DDOS, SQL Injection, XSS, Homograph, Generating and storing session tokens.

Networking Scanning & Security Measures:

Packet Sniffing and spoofing, Network scanning types, port scanning & its tools, and Network Architecture

Security Measures :IPtables(firewalls) , Webservers (Nmap & Metasploit for securing webservers),

Cyber Threats and Attacks (Malware, DOS, MITM, Social engineering attacks, Spoofing, Phishing) *Cross-Site Request Forgery (XSRF/CSRF), spear phishing.*

Unit III

Intrusion Detection System **12 Hours**

Intruders, Intrusion Detection, Analysis Approaches, Network-Based IDS, Host-Based IDS, signature based IDS, anomaly based IDS, advantages and disadvantages of NIDS and HIDS

Intrusion Detection Tools, snort architecture, snort rules, case studies of intrusion detection systems,Intrusion detection exchange format.

Honeypots, different types of honeypots, benefits and dangers of honeypots *firewall vs IDS, Physical IDS, honeynet*

Unit IV

Cyber Laws and Digital Forensics **12 Hours**

Digital Forensics: Introduction to Digital Forensics, historical background of digital forensics, Forensic Software, and Hardware, need for computer forensics science, special tools and techniques digital forensic life cycle, challenges in digital forensic.

Law Perspective: Introduction to the Legal Perspectives of Cybercrimes and Cyber security, Cybercrime and the Legal Landscape around the World, Why Do We Need Cyber laws, The Indian IT Act, Cybercrime Scenario in India, Digital Signatures and the Indian IT Act. *Cybercrime and Punishment*

Total: 48 Hours

Textbook (s)

1. Wenliang Du, Computer & Internet Security: A Hands-on Approach, (2020)
2. William Stallings, Lawrie Brown, Computer Security Principle sand Practice Third Edition,2015
3. Sunit Belapure and Nina Godbole, Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley India Pvt. Ltd, 2011.
4. Nelson Phillips and Enfinger Steuart, "Computer Forensics and Investigations", Cengage Learning, New Delhi, 6th edition 2018.

Reference (s)

1. Pande, Jeetendra. "Introduction to Cyber Security.", (2017)
2. Pavan Duggal, Cyber frauds, cybercrimes & law in India. 21 July 2013.
3. Ali A. Ghorbani, Network intrusion detection and prevention concepts and techniques, Springer, 2010
4. Roberto Di Pietro, Luigi V. Mancini (2008), Intrusion Detection System, Springer
5. DafyddStuttard and Marcus Pinto. The Web Application Hacker's Handbook: Finding and Exploiting Security Flaws, Wiley Publication, 2nd edition, 31 Aug 2011.

21FIX01 Full Semester Internship (FSI)

0 0 0 9

Course Outcomes

At the end of the Full Semester Internship the students are able to

1) Use the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems
2) Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
3) Select appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations
4) Use ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice
5) Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings
6) Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions

COs-POs Mapping:

COs	PO1	PO2	PO5	PO8	PO9	PO10	PSO1	PSO2
CO1	3	-	-	-	-	-	3	3
CO2	-	3	-	-	-	-	3	3
CO3	-	-	3	-	-	-	3	3
CO4	-	-	-	3	-	-	-	-
CO5	-	-	-	-	3	-	-	-
CO6	-	-	-	-	-	3	-	-

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Department of CSE-AI&ML

Minimum Credits to be earned: 160 (for Regular Students)
120 for Lateral entry students

S.No	Course Code	Course Name	POs	L	T	P	C
First Semester							
1	23PYX01 23CYX01	Engineering Physics/Chemistry	3/3	3	0	0	3/3
2	23MAX01 23MAX02	Linear Algebra& Calculus/Differential Equations and Vector calculus	3/3	3	0	0	3/3
3	23BEX01/ 23BEX02	Basic Electrical and Electronics Engineering/ Basic Civil & Mechanical Engineering	3/3	3	0	0	3/3
4	23BEX03	Introduction to Programming	3	3	0	0	3
5	23BEX04/ 23HSX01	Engineering Graphics/Communicative English	2/2	2	0	2/0	3/2
6	23PYX02/ 23CYX03/	Engineering Physics Lab/Chemistry Lab	4	2	0	2/2	1/1
7	23BEX05/ 23BEX06	Electrical & Electronics Engineering workshop/Engineering Workshop	1,9,10/1,5,10	2	0	3/3	1.5/1.5
8	23BEX07	Computer Programming Lab	4	2	0	3	1.5
9	23HSX11	-/ECA (Yoga / Sports)	-	-	0	-/1	-/0.5
10	23HSX12	-/CCA (NSS/NCC/Community Service)	-	-	0	-/1	-/0.5
11	23BEX08	IT Workshop/-		0	0	2/-	1/-
12	23HSX02	- /Communicative English Lab		0	0	-/2	-/1
			Total	14/14	00	12/12	20/20
Second Semester							
1	23HSX01/ 23BEX04	Communicative English/ Engineering Graphics	10,12	2/2	0	0/2	2/3
2	23MAX02/ 23MAX01	Differential Equations and Vector calculus/Linear Algebra& Calculus	1	3/3	0	0	3/3
3	23CYX01/ 23PYX01	Chemistry /Engineering Physics	1/1	3/3	0	0	3/3
4	23BEX02/ 23BEX01	Basic Civil & Mechanical Engineering/ Basic Electrical and Electronics Engineering	1,12/1,12	3/3	0	0	3/3
5	23CS201	Data Structures(CSE,CSE-AI&DS,CSE-AI&ML, IT)	1,12	3/3	0	0	3/3
6	23CYX03/ 23PYX02	Chemistry Lab / Engineering Physics Lab	4	0	0	2/2	1/1
7	23BEX06/ 23BEX05	Engineering Workshop/Electrical & Electronics Engineering workshop	1,9,10/1,5,10	0	0	3/3	1.5/1.5
8	23BEX08	IT Workshop/-	4/4	0	0	2/-	1/-
9	23HSX02	Communicative English Lab/-	-/10,12	0	0	2/-	1/-

10	23CS202	Data Structures Lab (CSE, CSE-AI&DS, CSE-AI&ML, IT)	2,3,4,5	0	0	3/3	1.5/1.5
11	23HSX11	-/ECA (Yoga/ Sports)		-	-	-/1	-/0.5
12	23HSX12	-/CCA (NSS/NCC/Community Service)		-	-	-/1	-/0.5
			Total	14/14	0	12/12	20/20

Third Semester

1	23CS301	Problem Solving using Python	2,3	3	-	2	4
2	23ML302	Artificial Intelligence	1,2,3	3	-	-	3
3	23CS303	Design and Analysis of Algorithms	1,2,12	3	-	-	3
4	23CS304	Digital Logic Design	1, 4	3	-	2	4
5	23DS305	Mathematical Foundation for Data Science	1,12,PSO1	3	-	-	3
6	23CS306	Object Oriented Programming with JAVA	1,2,3	3	-	-	3
7	23CS307	Design and Analysis of Algorithms Lab	2,3,4,5	-	-	3	1.5
8	23CS308	JAVA Lab	2,3,4,5	-	-	3	1.5
9	23BEA01	Environmental Studies	1,7	-	-	-	-
10	23ESX01	Employability Skills I	1,2,5,8,10,12	0	-	2	-
			Total	**18**	**-**	**12**	**23**

i

Fourth Semester

1	23IT304	Database Management Systems	1,4,12	3	-	-	3
2	23IT403	Operating Systems	1,12	3	-	-	3
3	23CS403	Computer Organization and Architecture	1,12	3	-	-	3
4	23MA404	Probability and Statistics usingpython	1,4, 12	3	-	2	4
5	23ML405	Foundations of Machine learning	2, 3, PSO1,PSO2	3	-	-	3
6	23IT308	Database Management Lab Systems	4	-	-	3	1.5
7	23ML407	Foundations of Machine learning Lab	4,5,8	-	-	3	1.5
8	23ESX01	Employability Skills I	1,2,5,8,10,12	0	-	2	2
			Total	**15**	·	**10**	**21**

Fifth Semester

1	23IT405	Web Technologies	3,5,PSO1	3	-	2	3
2	23ML502	Neural Networks	1,2,4,5,12	3	-	-	3
3	23DS503	Data Analytics &Visualization Techniques	1,4,PSO1	3	-	2	3
4	23ML504	Computer Networks	1, 2,3	3	-	-	3
5		Elective I (Professional Elective)		3	-	-	3
6		Elective II (Open Elective I)		3	-	-	3
7	23ML507	Neural Networks Lab	4,5,8	-	-	3	1.5
8	23TPX01	Term Paper	1,4,10,12	-	-	3	1.5
9	23ESX02	Employability Skills II	1,2,5,8,10,12	0	-	2	2
10	23SIX01	Summer Internship I	1,2,8,10,12	-	-	-	1
			Total	**18**	**-**	**12**	**24**
Sixth Semester							
1	23ML601	Deep Learning Techniques	1, 2,4,12,PSO1	3	-	-	3
2	23ML602	Automata Theory and Language Processors	1,2,3	3	-	-	3
3	23CS603	Software Engineering	4,5,8,11, PSO1	3	-	-	3
4		Elective III (Professional Elective)		3	-	2	4
5		Elective IV (Open Elective II)		3	-	-	3
6	23ML606	Deep Learning Techniques Lab	4, 5,08	-	-	3	1.5
7	23MPX01	Mini Project	All POs & PSOs	-	-	3	1.5
8	23ESX02	Employability Skills II	1,2,3,5,6,8,10,12	0	-	2	2
9	23ATX01	Environmental Studies	1,7	-	-	-	-
10	23ATX02	Professional Ethics and Human Values	**----**	-	-	-	-
11	23ATX---	Audit Course	**12**	-	-	-	-
			Total	**15**	**-**	**10**	**21**
Seventh Semester– AI & ML							
1		Elective V (Professional Elective)		3	-	-	3
2		Elective VI (Professional Elective)		3	-	-	3
3		Elective VII (Open Elective III)		3	-	-	3
4	23SIX02	Summer Internship II	1,2,5,6,10,12	-	-	-	1
5	23PWX01	Project	All POs& PSOs	-	-	16	8
			Total	**9**	**-**	**16**	**18**
Eighth Semester-AI & ML							
1		Elective VIII (Professional Elective)		-	-	-	3
2		Elective IX (Open Elective IV)		-	-	-	2
3	23FIX01	Full Semester Internship (FSI)	1,2,5,8,9,10, PSO1,PSO2	-	-	-	8
			Total	**6**	**-**	**-**	**13**

List of Electives

Language Electives							
No.	Course Code	Course	POs	Contact Hours			
				L	T	P	C
1	23HSX03	Advanced Communicative English	10,12	2	-	-	2
2	23HSX04	Communicative German		2	-	-	2
3	23HSX05	Communicative French		2	-	-	2
4	23HSX06	Communicative Japanese		2	-	-	2
5	23HSX07	Communicative Spanish		2	-	-	2
6	23HSX08	Communicative Korean		2	-	-	2
7	23HSX09	Communicative Hindi		2	-	-	2

Elective I							
Career Path I							
1	23MLC11	Computer Vision & Pattern Recognition	1,3,PSO1,PSO2	3	-	-	3
2	23CSC21	Web Programming Languages (Full Stack Development)	1,2,7,12	3	-	-	3
3	23MLC31	Fundamentals of Cloud Computing	2,6,7,8	3	-	-	3
4	23CS004	Principles of Programming Languages	1, 2, 3, 4	3	-	-	3
5	23CS005	Mobile Computing	3, 8	3	-	-	3
6	23CS006	Distributed Operating Systems	1,2	3	-	-	3
Elective III							
Career Path II							
1	23MLC12	Machine Learning for Business Intelligence	2,3,PSO1,PSO2	3	-	2	4
2	23CSC22	Web Application Developments Framework (Full Stack Development)	1, 3,4	3	-	2	4
3	23MLC32	Cloud Services using AWS		3	-	2	4
4	23CS007	Cloud Computing Essentials	2,5,6,7,8	3	-	2	4
5	23CS008	Cryptography and Network Security	3, 6,8	3	-	2	4
Elective V							
Career Path III							
1	23MLC13	Conversational AI	1,2,4,12,PSO1	3	-	-	3
2	23CSC23	Web Application Databases (Full Stack Development)	2,3	3	-	-	3
3	23MLC33	Cloud Security Essentials	2,3	3	-	-	3
4	23IT010	Social Network Analysis	2, 4, 5	3	-	-	3
5	23ML001	Human Computer Interaction	2, 3	3	-	-	3
6	23CS012	Wireless Adhoc Networks	2, 3	3	-	-	3
Elective VI							
1	23ML002	Large Language Models	2,3,5,6	3	-	-	3
2	23CS015	Software Project Management	3,6	3	-	-	3
3	23ML003	Reinforcement Learning	6,7	3	-	-	3

Elective VIII: Professional Elective							
1	23CS018	Information Retrieval Systems	1,2,3,4	-	-	-	3
2	23CS019	Fundamentals of Devops	1,3, 5,8,10	-	-	-	3

3	23DS003	Cyber Security	1,3,4,5,PSO1				3

Audit Course							
1	23AT001	Communication Etiquette in Workplaces	-	-	-	-	-
2	23AT002	Contemporary India: Economy, Policy and Society	-	-	-	-	-
3	23AT003	Design The Thinking	-	-	-	-	-
4	23AT004	Ethics and Integrity	-	-	-	-	-
5	23AT005	Indian Heritage and Culture	-	-	-	-	-
6	23AT007	Intellectual Property Rights and Patents	-	-	-	-	-
7	23AT008	Introduction to Journalism	-	-	-	-	-
8	23AT009	Mass Media Communication	-	-	-	-	-
9	23AT010	Science, Technology and Development	-	-	-	-	-
10	23AT011	Social Responsibility	-	-	-	-	-
11	23AT012	The Art of Photography and Film Making	-	-	-	-	-
12	23AT013	Gender Equality for Sustainability	-	-	-	-	-
13	23AT014	Women in Leadership	-	-	-	-	-
14	23AT015	Introduction to Research Methodology	-	-	-	-	-
15	23AT016	Climate Change and Circular Economy	-	-	-	-	-

iii

B. Tech. (Honors)							
Domain I (Data Engineering)							
1	23CSH11	Advanced Data Structures	2,3,4	4	-	-	4
2	23CSH12	Advanced Databases	2,3,4	4	-	-	4
3	23CSH13	Programming, Data Structures and Algorithms Using Python	2,3,4,5	4	-	-	4
4	23CSH14	Bioinformatics	2,3	4	-	-	4
Domain II (Modern Software Engineering)							
1	23CSH21	DevOps	1,3,5,8,10	4	-	-	4
2	23CSH22	Design Patterns	2,3	4	-	-	4
3	23CSH23	Advanced Software Engineering	1,3,4, PSO1	4	-	-	4
4	23CSH24	Robotic Process Automation	3,5, 8, PSO2	4	-	-	4
Domain III(Security)							
1	23CSH31	Fundamentals of Systems Security	1,2	4	-	-	4
2	23CSH32	Python Programming for Security	2,3,4	4	-	-	4
3	23CSH33	Management of Information Security	3,6,7	4	-	-	4
4	23CSH34	Computer Forensics	2,3	4	-	-	4
Domain IV (User Interface Design)							
1	23CSH41	Computer Graphics	1,2,3,4	4	-	-	4
2	23CSH42	Multimedia Systems	3,4	4	-	-	4
3	23CSH43	Human Computer Interaction	2,3	4	-	-	4
4	23CSH44	Mobile Programming	3,4	4	-	-	4

3rd Semester

23CS301 Problem Solving using Python

3 0 2 4

Course Outcomes

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

1. Explain the fundamentals of Python with syntax and semantics.
2. Apply conditional and control flow statements, and analyze strings and regular expressions for pattern matching and text processing.
3. Understand and apply the built-in data structures such as dictionaries, sets, lists, and tuples for data manipulation.
4. Demonstrate and apply the concepts of functions and functional programming techniques.
5. Understand and apply the file Handling and exception handling techniques.
6. Understand the principles of object oriented programing in Python and demonstrate the use of Modules and Packages.

CO – PO Mapping

CO	PO1	PO2	PO3	PO12
1	3	2	2	1
2	3	2	2	1
3	3	3	3	1
4	3	3	3	1
5	3	3	3	1
6	2	3	3	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Functions- Defining Functions, Calling Functions, Types of Arguments, Recursion, Anonymous Functions: Lambda

Expressions, Global and Local Variables. Functional Programming: map, filter and reduce, decorators. **Modules and**

Unit I **12+8 Hours**

Introduction: History and need for python programming, Identifiers, Statements, Variables, Keywords, Input-Output statements, Indentation, command line arguments.

Data Types - Integers, Strings, Booleans. **Operators -** Arithmetic, Comparison/Relational, Assignment, Logical, Bitwise, Membership, Identity. **Expressions and order of evaluations -** Precedence and Associativity. **Built-in Functions** – Importing from Packages. **Conditional Statements**: If, Elif, Else statements, iteration statements - While, For - Break, Continue and pass statements.

Applications of Python, REPL (Read, Evaluation, Print, Loop)

Practical Components

1. Basic Operations: Validating and Converting Date Formats and Numeric Systems in Python
2. Control Statements: Implementing Fibonacci Series and Advanced List Operations
3. Array Manipulations : Matrix Operations: Transpose, Symmetry Check, and Multiplication

Unit II **12+6 Hours**

Collections: Lists: Create, Access, Slicing, Negative indices, List methods, List comprehensions – Tuples: Create, Indexing and slicing, Operations on tuples – Dictionary: Create, add, and replace values, Operations on dictionaries – Sets: Creation and operations.

Strings: Strings: Comparison, Formatting, Slicing, Splitting, Stripping . **Regular Expressions**: Matching, Search and replace, Patterns.

collections.deque, grouping with Dictionaries

Practical Components

1. Array Manipulations: Intersection, Kth Largest Element, and Maximum Subarray in Lists
2. Working with Dictionaries: 6. Word Frequency Counting, Dictionary Merging, and Filtering
3. String Manipulation: Acronyms, Valid Parentheses, Anagram Check, and Common Prefix
4. Regular Expressions for Email Validation, Phone Number Extraction, and Password Strength Checking

Unit III **12 +8 Hours**

Packages- Built-in modules, User-Defined modules, itertools module. **Numpy and Pandas:** Introduction to Numpy and Pandas packages of python, Basic operations on Numpy and Pandas.

Higher order function, keyworded variable length arguments. **Practical Components**

1. Recursive Functions: Prime Factorization, Subsets Generation, and Fibonacci Sequence
2. Exploring Default and Keyword Arguments, Variable Scopes, and Lambda Functions
3. Functional Programming: Map, Filter, and Reduce Operations
4. Itertools Applications: Counting, Cycling, Permutations, Combinations, and Sliding Windows
5. Working with Numpy Arrays and Pandas DataFrames

Unit IV **12+10 Hours**

Classes and Objects: Defining and instantiating Class, Class Attributes and Methods, Constructor, self-keyword inheritance – overriding methods from parent class – Instances as Attributes, inner class. **File Handling-**Creating files, operations on files - Open, Close, Read and Write - Working with File Paths - File and Directory Operations. **Exception Handling-** Exception, try except block, Raising Exceptions, User Defined Exception.

polymorphism, Abstract class and abstract methods. **Practical Components**

1. Object-Oriented Programming: Creating and Using Classes and Objects

2. Inheritance and Method Overloading/Overriding in Python
3. File Handling and Simple File Explorer Implementation
4. Exception Handling: Custom Exceptions

Total: 48+32 Hours

Textbook (s)

1. Richard L. Halterman, "Fundamentals of Python Programming" , 3rd Edition, Southern Adventist University, 2019.
2. Matthes, E. ,"Python Crash Course: A Hands-On, Project-Based Introduction to Programming", 2nd Edition, No Starch Press, 2019.
3. Willaim Mckenny, "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython" 2nd Edition, O'Reily 2017. **Reference(s)**

1. Wesley J Chun, "Core Python Applications Programming", 3rd Edition, Pearson Education India, 2015.
2. Mertz, D., "Functional Programming in Python", 1st Edition, O'Reilly Media,2015.
3. Kenneth A. Lambert. "Fundamentals of Python: First Programs", 2nd Edition, Publisher: Cengage Learning 2018
4. Python Programming: A Modern Approach, VamsiKurama, Pearson 2017 Learning Python, Mark Lutz, Orielly, 5th Edition.

23ML302 Artificial Intelligence

3 0 0 3

Course Outcomes

1. Summarize the fundamental concepts of Artificial Intelligence.
2. Understand the control strategies and production systems.
3. Illustrate the Concepts of Heuristic Search Techniques.
4. Demonstrate working knowledge of reasoning in the presence of incomplete and/or uncertain information
5. Make use of concept of Game Playing Algorithms.
6. Outline the concept of Planning System

CO–PO Mapping

CO	PO1	PO2	PO3	PSO1	PSO2
1	2	1	2	2	2
2	2	2	3	3	3
3	3	1	1	3	3
4	2	2	2	3	3
5	1	3	2	3	3
6	1	2	2	3	3

3. Strongly linked | 2-Moderately linked | 1-Weaklylinked

Unit I 12 Hours

Introduction

Definition, AI problems, AI techniques, Defining problem as a state space search, Production systemsControl Strategies, Problem characteristics, Production system characteristics. Problems – tic-tac-toe, 8puzzle problem, Chess problem, Water Jug Problem, Missionaries and cannibals problem, Monkey and banana problem.

Application of AI-Tower of Hanoi problem- *Travelling Salesman Problem*

Unit II 12 Hours

Heuristic Search Techniques

Generate-and-test, Hill climbing, Best-first-search – OR Graphs – A* Algorithm, Problem reduction – AND-OR Graphs – AO* Algorithm, constraint satisfaction - cryptarithmetic problem. *Depth-first search – Breadth-first search- Data abstraction*

Unit III 12 Hours

Knowledge representation

Knowledge Representation Issues, Representation and mapping, Approaches to Knowledge Representation, Frame Problem, Propositional logic, Predicate logic, Resolution, weak slot and filler structure-semantic nets.

Filler structures-conceptual dependency-scripts

Unit IV 12 Hours

Game Playing and Planning

Mini-max search, Alpha-beta cutoffs, planning system, Block world problem, goal stack planning, hierarchical planning, Natural language processing, syntactic processing. Decision trees, Perception, Vision, Speech recognition.

Learning - Navigation – Manipulation - Robot Architecture

Textbook (s)

1. E. Rich K .Knight, and B. Nair, Artificial Intelligence, 3rdEdition, TMH, 1 July 2017
2. Russel Norvig, Artificial Intelligence A modern Approach, 3rd Edition, Pearson Education, 2010

Reference (s)

1. Patrick henry Winston, Artificial Intelligence, third edition, Pearson Education Asia, 2005
2. Dan W. Patterson, Introduction to Artificial intelligence and Expert Systems, 2nd Edition, PHI, 2009

23CS303 Design and Analysis of Algorithms

3 0 0 3

Course Outcomes

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

1. Analyse the performance of algorithms using space and time complexity, and asymptotic notations.
2. Implement algorithms using brute force and divide and conquer approaches.
3. Apply advanced data structures and graph algorithms to solve complex computational problems effectively.
4. Demonstrate the ability to solve problems efficiently using greedy approach.
5. Develop problem-solving skills by tackling complex computational problems using dynamic programming and backtracking.
6. Apply branch and bound techniques to solve complex problems, and understand the concepts of NPcompleteness to identify NP-complete problems.

COs–POs Mapping

COs	PO1	PO2	PO3	PO4	PSO1
1	3	3	-	1	2
2	2	2	2	1	2
3	3	3	3	2	2
4	3	2	3	2	2
5	3	2	2	2	2
6	3	3	2	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **12 hours**

Algorithm Analysis and Divide and Conquer

Fundamentals of algorithmic problem solving - Analysis framework - Performance Analysis: Space and Time complexity - Growth of Functions: Asymptotic Notations - Mathematical Analysis of Non-recursive and Recursive algorithms - Master's theorem - Brute Force String Matching Algorithm- Divide and Conquer: Merge sort, Quick sort, Fibonacci search.

Interpolation search and Radix sort

Unit II **12 hours**

Advanced Data Structures and Graph Algorithms:

Balanced Search Trees: AVL trees, B-trees - Heaps: Heapsort - Disjoint set: Find and Union Operations - Graphs: Introduction, Representation - Graph Traversals: BFS, DFS, Topological sort.

Tries and Strongly Connected Components

Unit III **12 hours**

Greedy and Dynamic Programming

Greedy Method: General method, Job sequencing with deadlines, Optimal Merge Patterns - Minimum cost spanning trees: Prim’s and Kruskal’s Algorithms - Single source shortest path problem: Dijkstra’s Algorithm - Dynamic Programming: General method, Matrix chain multiplication, 0/1 knapsack problem, All pairs shortest path problem: Floyd Warshall Algorithm.

Graph coloring problem, Huffman Trees and Codes

Unit IV **12 hours**

Backtracking and Brach and Bound

Backtracking: General method, n-queen problem, sum of subsets problem, Hamiltonian Circuit problem. Branch and Bound: General method, Knapsack Problem, Traveling sales person problem, Assignment Problem.

Complexity Classes : P, NP and NP-Complete Problems.

Exhaustive Search, Space and Time Trade-Offs

Total: 48hours Textbook (s)

1. Introduction to The Design and Analysis of Algorithms, 3rdEdition, AnanyLevitin, Pearson Education, 2017.
2. Introduction to Algorithms, second edition, T.H.Cormen, C.E.Leiserson, R.L. Rivest, and C.Stein, PHI Pvt. Ltd./ PearsonEducation
3. Fundamentals of Computer Algorithms, Ellis Horowitz, SatrajSahni and Rajasekaran, University press.

Reference (s)

1. Design and Analysis of algorithms,Aho,Ullman and Hopcroft,Pearsoneducation.
2. Algorithms–Richard Johnson BaughandM arcus Schaefer,Pearson Education.
3. www.geeksforgeeks.org
4. www.hackerearth.com
5. www.tutorialspoint.com

23DS305 Mathematical Foundation for Data Science

3 0 0 3

Course Outcomes

After undergoing this course, the learners are enabled to 1.Demonstrate Vector Spaces and their applications.

2.Solve Problems involving, Linear Transformations and Inner product spaces 3.Apply generating functions and Characteristic roots to solve recurrence relations 4.Derive statements using the rules of inference for Mathematical logic.

5.Understand the basic knowledge of Graph theory and applications and the rules of inference for Predicate logic.

6.Construct the minimum spanning tree using different algorithms.

CO-PO Mapping

COs	PO1	PO2	PO12	PSO1
1	3	2	1	1
2	3	2	1	3
3	3	2	1	3
4	3	2	1	3
5	3	2	1	3
6	3	2	1	3

3-Strongly linked | 2-Moderately linked | 1-Weaklylinked

Unit I

Vector Spaces and Linear Transformations

Vector Spaces: Definition, Properties of Vector Spaces, Subspaces, Algebra of Subspaces, Basis and Dimension, Linear Span, Linear Independence and Dependence of vectors.

Linear Transformations: Definition, Properties, Range and Null Spaces, Rank-Nullity theorem, Matrices of Transformations. (All theorems without Proof)

Singular Value Decomposition

Unit II

Inner Product Space and Recurrence Relations

Inner Product Spaces: Definition, Properties of Inner Product Spaces, Norm of a vector, CauchySchwarz's Inequality (without Proof), Complete orthonormal set, Gram-Schmidt Orthogonalization process.

Recurrence Relations: Generating Functions, Calculating Coefficient of Generating Functions, Solving Recurrence Relations by Substitution method, Solving Recurrence Relations by Method of Characteristic roots, Solving Recurrence Relations by generating functions

Bessel's inequality

UnitIII Mathematical Logic

Propositional Logic: Statements and Notations, Connectives, Well-formed Formulas, Truth Tables, Tautology, Equivalence Implication, Normal Forms, Rules of Inference for Statement Logic, Proof Techniques, Proof by Contradiction, Method of Induction.

Predicative Logic: Quantifiers, Universal Quantifiers, Free & Bound Variables, Rules of Inference for Predicate Logic.

Relations in Databases

Unit IV

Graph Theory

Basics of Graph Theory: Representation of Graphs, Degree of a Graph, Complete Graphs, Regular

Graphs,Bipartite Graphs, Walk – Path – Circuit – Cycle, Planar Graphs, Euler's Formula, Isomorphism, Euler Circuit and Hamilton Cycle, Chromatic Number.

Trees: Definition and properties of tree, Spanning Tree, Breadth First Search (BFS) algorithm, Depth First Search (DFS) algorithm, Minimum Spanning Tree, Kruskal algorithm, Prim's algorithm.
Applications of Graphs and Trees, Ternary Tree

Textbook(s)

1. Gilbert Strang, "Linear Algebra and Learning from Data", Wellesley, Cambridge Press, 2019.
2. Kenneth H. Rosen, Discrete Mathematics and Applications, 7th Edition, Tata McGraw Hill, 2015.
3. J. L. Mott, A. Kandel & T. P. Baker, Discrete Mathematics for Computer Scientists & Mathematicians, Prentice Hall India, 2nd Edition, 2010.

Reference (s)

1. Tremblay J.P. and P. Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill,2007.
2. Erwin Kreyszig, Advanced Engineering Mathematics, Wiley India, 2015.
3. K Hoffman and R Kunze, Linear Algebra, Pearson Education, 2017.

23CS306 Object Oriented Programming with JAVA

3 0 0 3

Course Outcomes

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

1. Understand the object oriented programming concepts
2. Develop applications using different types of inheritances
3. Understand the creation of user defined packages
4. Analyze and recover runtime exceptions arise in the applications
5. Develop multithreaded applications with synchronization
6. Develop interactive applications for standalone and Internet

COs–POs Mapping

COs	PO1	PO2	PO3	PSO2
1	3	2	1	1
2	1	1	1	1
3	1	1	1	1
4	2	2	2	2
5	3	2	2	2
6	1	1	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I **13 Hours**

Introduction to Java

Overview of Object Oriented Programming principles, Importance of Java to the Internet, Byte code, Methods, classes and instances, Data types, arrays, control statements, simple java program, Classes and Objects–constructors, methods, access control, this keyword, overloading methods and constructors, garbage collection, java.lang – String, StringBuffer, StringBuilder, Type wrappers – Integer, Character, Float, Double, Boolean

Features of object oriented programming–Java History–Computer Programming Hierarchy–Role of Java Programmer in Industry

Unit II **11 Hours**

Inheritance, Packages & Interface

Inheritance: Basics, Using super, Multilevel Hierarchy, Method overriding, Dynamic Method Despatch, Using Abstract classes, Using final with inheritance, Sealed Classes

Packages: Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, Member access rules, Java Collections – Arrays, ArrayList, Hashtable, StringTokenizer, Vector

Interface: Defining an interface, differences between classes and interfaces, implementing interface, variables in interface and extending interfaces, Default Methods in Interfaces, Static Methods in Interface

Nested–Inner Class & Anonymous Classes–Generic Class Types

Unit III **12 Hours**

Exception Handling & Multithreading

Exception handling: Concepts and benefits of exception handling, exception hierarchy, usage of try, catch, throw, throws and finally, built-in and User Defined Exceptions

Multithreading: Definition thread, thread life cycle, creating threads, synchronizing threads, daemon threads

Control Flow In Exceptions– JVM reaction to Exceptions– Inter Communication of Threads– Critical Factor in Thread–Deadlock

Unit IV **12 Hours**

Applets & Event Handling

Applets: Concepts of Applets, life cycle of an applet, types of applets, creating applets, passing parameters to applets, The AWT class hierarchy, user interface components labels, button, Text components

Event Handling: Events, Delegation event model, handling mouse and keyboard events, Adapter classes, inner classes, Compare basic AWT components with swing components, More user interface components-canvas, scrollbars, check box, choices, lists panels–scroll pane, dialogs, menu bar, layout managers.

Applet Security Policies–Anonymous Inner classes a Short–cut to Event Handling–Java Utilities (java.util Package)

Total: 48 Hours

Textbook (s)

1. H. Schildt, Java: The complete reference, 7th Edition, TMH, 2006
2. T. A. Budd, An Introduction to Object–Oriented Programming, 3rd Edition, Addison Wesley Longman, 2002

Reference (s)

1. Dietal&Dietal, Java: How to Program, 8th Edition, PHI, 2010
2. C. S. Horstmann and G. Cornell, Core Java, Vol 1. Fundamentals, 7th Edition, Pearson Education, 2004
3. C. Horstmann, BIG JAVA Compatible with Java 5 & 6, 3rd Edition, Wiley Publishers, 2008

23CS307 Design and Analysis of Algorithms Lab

0 0 3 1.5

Course Outcomes

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

1. Analyze the performance of algorithms by implementing and measuring their space and time complexity using asymptotic notations.
2. Implement algorithms using brute force and divide and conquer approaches, and assess their efficiency.
3. Implement advanced data structures and graph algorithms to solve complex computational problems, and evaluate their performance.
4. Solve optimization problems efficiently using a greedy approach, and analyze the results.
5. Develop dynamic programming solutions to tackle complex computational problems, and measure their effectiveness.
6. Apply backtracking and branch and bound techniques to solve complex problems.

CO-PO Mapping

COs	PO1	PO2	PO3	PO4	PSO1
1	3	3	-	1	2
2	2	2	2	1	2
3	3	3	3	2	2
4	3	2	3	2	2
5	3	2	2	2	2
6	3	3	2	1	1

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Practical Components

1. Analysis of recursive algorithms
2. Analysis of Non-recursive algorithms
3. Solve the String matching problem using Brute Force approach 4. Solve sorting problem using divide and conquer approach
5. Implement heapsort algorithm using a max-heap.
6. Implement Simple Union and Find operation on disjoint-sets 7. Implement the graph traversal algorithm.
8. Implement topological sorting of a directed acyclic graph (DAG).
9. Find the minimum cost spanning tree for a given graph using greedy approach
10. Implement dijkstra's algorithm using greedy approach
11. Solve matrix chain multiplication problem using dynamic programming
12. Solve the 0/1 knapsack using dynamic programming
13. Implement Floyd algorithm using dynamic programming
14. Find the sum of subsets using backtracking technique 15. Solve N queen problem using the backtracking technique.
16. Find the solution for Assignment Problem using branch and bound method

23CS308 JAVA Lab

0 0 3 1.5

Course Outcomes

1. Make use of JAVA SDK environment to create - debug and run java programs
2. Create applications based on code reusability
3. Develop programs using threads
4. Develop and debug real time problems using exception handling
5. Create interactive applications using event handling mechanisms 6. Design Graphical User Interface using AWT components

CO-PO Mapping

COs	PO2	PO3	PO4	PO5
1	3	2	3	1
2	3	2	3	1
3	2	2	2	1
4	2	2	3	1
5	2	2	3	2
6	2	3	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

List of Experiments

1. Case Study: Permutations and Combinations Calculator **User Input**:
   - The program should prompt the user to enter the total number of elements (n) and the number of elements to be selected (r).
   - Validate the input to ensure that both n and r are positive integers, and r is less than or equal to n. **Functional Description:**
   - Create a method **calculate Permutations** that takes two integers n and r as input and returns the number of permutations.

   Formula *nPr = n! / (n - r)!, where "!" denotes factorial.*
   - Create a method **calculate Combinations** that takes two integers n and r as input and returns the number of combinations.

   Formula *nCr = n! / (r! * (n - r)!).*
   - Implement a separate method to calculate the factorial of a given integer.

   Sample Output:

   ```
   Enter the total number of elements (n): 5
   Enter the number of elements to be selected (r): 3
   Permutations (nPr): 60
   Combinations (nCr): 10
   ```
2. Conversion Utility that allows users to convert measurements between different units. **User Input**:
   - The program should prompt the user to enter positive integer value for conversion type.
   - The program should prompt the user to enter the positive input (feet/ meters/ miles/ kilometers) based on user option.

   Functional Description:
   - utilize method overloading to handle conversions between various units, such as feet to meters, miles to kilometers, and vice versa. **formula**:feet to meters - feet * 0.3048 miles to kilometers - miles * 1.60934 **Sample Output:**

     ```
     Measurement Conversion Utility
     1. Feet to Meters
     2. Meters to Feet
     3. Miles to Kilometers
     4. Kilometers to Miles
     5. Exit
     Enter your choice: 3

     Enter the distance in miles: 5
     5 miles is approximately 8.046 kilometers.
     ```
3. Case Study: **BankAccount** Class with Constructor Overloading **Functional Description:**
   - Create a **BankAccount** class to represent a bank account. The class should have private attributes such as **accountNumber** (a unique identifier for the account), **accountHolderName**, and **balance** (representing the current account balance).
   - The class should have the following constructors:
     - **BankAccount(String accountNumber, String accountHolderName)**: Create an account with a zero balance.

- ○ **BankAccount(String accountNumber, String accountHolderName, double initialBalance)**: Create an account with the specified initial balance.
- Implement methods to: ○ **deposit(double amount):** Add the given amount to the account balance.
  - ○ **withdraw(double amount):** Subtract the given amount from the account balance if sufficient funds are available.

Sample Output:

```
Welcome to the Bank Account System Creating
a new account...
Enter the account holder's name: John
Doe Enter the initial balance: 500.0
Account created successfully.
Account Number: ACCT123456
Account Holder: John Doe
Balance: $500.00
What would you like to do?
1. Deposit
2. Withdraw
3. Exit
Enter your choice: 1
Enter the amount to deposit: 250.0
$250.00 deposited successfully. Updated
Balance: $750.00
```

4. Implement inheritance concept for below case study:
   Note: class **FictionBook** extends class **LibraryBook**, class **ClassicBook** and class **ScienceFictionBook** extends from **FictionBook**
   - **LibraryBook**: Represents a general library book with attributes like title, author, and year, along with a method **displayBookInfo()** to display book details.
   - **FictionBook**: Extends **LibraryBook** and adds an additional attribute genre.
   - **ClassicBook**: Extends **FictionBook** and adds an additional attribute publisher.
   - **ScienceFictionBook**: Extends **FictionBook** and adds an additional attribute edition
5. Case Study: Abstract Class for Finding Areas of Different Shapes **User Input:**
   - The program should prompt the user to enter positive integer value to calculate type of area.
   - The program should prompt the user to enter the input based on user option.
     - ○ **Circle**: Requires the radius as an input parameter ○ **Rectangle**: Requires the length and width as input parameters. ○ **Triangle**: Requires the base and height as input parameters.
     - ○ **Square**: Requires the side length as an input parameter.

   Functional Description:
   - Create an abstract class **Shape** that will serve as the base class for different shapes. Include an abstract method **calculateArea()**
   - Create concrete subclasses that extend the **Shape** class for different shapes like **Circle**, **Rectangle**, **Triangle** and **Square**. **Sample Output:**

```
Select a shape:
1. Circle
2. Rectangle
3. Triangle
4. Square
5. Exit
Enter your choice: 1

Enter the radius of the circle: 5
The area of the selected shape is: 78.53981633974483
```

6. Implement Dynamic Method Dispatch in Java.
7. i. Create an **ArrayList** of integers to store a collection of numbers. Implement methods to add elements to the list, remove elements, update elements, and search for elements in the list.
   ii. Create a Java program that accepts a string input from the user. Use the **StringTokenizer** class to tokenize the input string into separate tokens based on specified delimiters (e.g., space, comma, etc.).
8. Demonstrating Mutability Differences between **String**, **StringBuilder**, and **StringBuffer** in Java
   - Explain the Concept in 100 words.

- Perform string manipulation operation concatenation on each class
- Print the results of each class after modification operation

9. Demonstrate multiple inheritance-like behavior achieved through interface implementation.
10. Case Study: Java Program Illustrating Multiple Catch Clauses with Finally Block **Functional Description:**
    - Implement exception handling for possible exceptions that may occur during the division, such as **ArithmeticException** and **NumberFormatException**.
    - Use multiple catch clauses to handle **ArithmeticException** and **NumberFormatException** separately.
    - Use a **finally** block to execute code that should be performed regardless of whether an exception occurred or not **Sample Output:**

    ```
    Enter the dividend: 10
    Enter the divisor: 0
    Division by zero is not allowed. Execution
    completed.
    ```
11. Create your own custom exception classes to handle specific situations or errors that are not covered by the built-in exception classes.

    Functional Description:
    - Create a custom exception class that extends the **Exception** class (or any other appropriate superclass, depending on your requirements).
    - Define the custom exception class to represent a specific error situation, such as **InvalidAgeException**, which is thrown when an invalid age is provided.
    - Use the custom exception class in a method to throw the exception when a certain condition is met, such as when the age is less than zero or exceeds a specific limit. **Constraints:**

    The program should prompt the user to enter age integer value. Here, age<0 or age>120 **Sample Output:**

    ```
    Enter age:-5
    Exception: Invalid age. Age must be between 0 and 120.
    ```
12. Creates three threads by extending the **Thread** class. Each thread will display a specific message at a regular interval. The messages to be displayed are as follows:
    First thread: "**Good Morning**" every 1 second
    Second thread: "**Hello**" every 2 seconds
    Third thread: "**Welcome**" every 3 seconds
    Note: implement the same by using **Runnable** interface.
13. Case Study: Java Program for Producer-Consumer Problem **User Input:**
    - The program should prompt the user to enter positive integer valuesfor **bufferSize**
    - The program should prompt the user to enter positive integer values to **produceCount** and **consumeCount Functional Description:** • Buffer Implementation: ○ Create a shared buffer with a limited size (e.g., an array or an ArrayList) to hold the produced items.
    - Producer & Consumer Implementation:
      - ○ Create a Producer class & Consumer class that implements the Runnable interface.
      - ○ The Producer class should produce items and put them into the buffer. ○ The Consumer class should consume items from the buffer.
      - ○ Implement synchronization to avoid race conditions when accessing the buffer.

    Sample Output:

    ```
    Enter Buffer Size:5

    Enter Procedure Count:5
    Enter Consumer Count:5
    Produced: 0
    Consumed: 0
    Produced: 1
    Consumed: 1
    Produced: 2
    Consumed: 2
    Produced: 3
    Consumed: 3
    Produced: 4
    Consumed: 4
    ```
14. Case Study: Java Program for a Basic Calculator using AWT

In this case study, we'll implement a Java program to create a basic calculator using AWT. The calculator will have a simple graphical user interface (GUI) with buttons for numbers, arithmetic operations, and a display to show the result of calculations.

AWT Implementation:

- Create a Java AWT to implement the calculator GUI.
- The AWT should have buttons for digits (0 to 9), arithmetic operations (addition, subtraction, multiplication, division), and an equal (=) button to perform calculations.

Display and Input:

- Implement a display area where the user can see the entered numbers and the result of calculations.
- Provide a way for users to input numbers and arithmetic operations using the GUI buttons. **Basic Calculations:**
- Implement methods to handle basic arithmetic calculations (addition, subtraction, multiplication, division).
- Perform calculations based on the user's input and display the result in the GUI. **User Interface:**
- Design a user-friendly GUI with appropriate labels, buttons, and display areas.
- Use layout managers to organize the components effectively.

15. Case Study: Java Program for Temperature Converter using AWT

In this case study, we'll implement a Java AWT to create a Temperature Converter. The AWT will allow users to convert temperatures between Celsius and Fahrenheit. The user will input a temperature value in one unit, and the AWT will display the converted temperature in the other unit.

AWT Implementation:

- Create a Java AWTpplet to implement the temperature converter GUI.
- The AWT should have input fields for the temperature value and buttons to select the source and target temperature units (Celsius or Fahrenheit).
- Use appropriate layout managers to organize the components effectively.

Conversion Logic:

- Implement methods to handle temperature conversions between Celsius and Fahrenheit.
- The temperature conversion formulas are: ○ Fahrenheit to Celsius: C = (F - 32) * 5 / 9 ○ Celsius to Fahrenheit: F = (C * 9 / 5) + 32 **User Interface:**
- Design a user-friendly GUI with appropriate labels, input fields, and buttons.
- Display the converted temperature in the GUI after the conversion is performed.

16. Case Study: Java Swing Application for Job Application Form

In this case study, we'll implement a Java Swing application with a JFrame that contains fields for name, age, email, and qualifications. The application will validate the form data when the user clicks the "Submit" button. If all fields are filled, the data will be written to a file named "application_form.txt." **Swing Application:**

- Create a Java Swing application with a JFrame to display the job application form.
- Design the form with labels for each field (name, age, email, qualifications) and corresponding input fields (text fields or other appropriate components).
- Add a "Submit" button to the form to allow users to submit their job application. **Data Validation:**
- Implement validation for the form data to ensure that all fields are filled before submission.
- Display error messages for missing fields or invalid data.

Data Writing:

- Create a method to write the validated form data to a file named "application_form.txt." • The file should be created in the current working directory.
- Use appropriate file handling techniques to write the data.

User Interaction:

- Allow users to input their name, age, email, and qualifications using the form fields.
- Trigger the form data validation when the user clicks the "Submit" button.
- Display appropriate messages to inform the user about the success or failure of form submission.

List of Augmented Experiments

1. New Patient Registry Management System
2. Restaurant Billing Management System
3. Library Management System
4. ATM Management System
5. Bus Ticket Booking Management System
6. Movie Ticket Booking Management System
7. Queuing Management System

8. Attendance Management System
9. Medical Store Billing Management System
10. Text Editor Projects in Java
11. Google Search Engine Filter
12. Electronic voting System
13. Day Planner
14. Library management System
15. Personal Finance Management System

Reading Material (s)

1. JAVA Lab manual, Department of CSE and IT, GMRIT, Rajam

23ESX01 Employability Skills I

0 0 2 0

Course Outcomes

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

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs –POs Mapping

COs	PO1	PO2	PO5	PO 8	PO10	PO12
CO1					3	2
CO2				1	2	2
CO3	2	1		2		
CO4	2		2			
CO5	2		2			
CO6	2		2			

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Unit-I Communication Skills 16 hours

Communication Skills & Confidence: How Communication Skills affect Confidence? How to communicate effectively.(with Examples)

Listening: Listening? , Listening Vs Hearing, Possible reasons for why people do not Listen at times, Active Listening Vs Passive Listening, How Listening can affect our relationships? How Listening helps in Campus Placements also? (with Examples)

Goal Setting: SMART Technique to Goal Setting, Putting First things First, SWOT Analysis and Time Management

Attitude & Gratitude: Attitude Vs Skills Vs Knowledge, Attitude Vs Behaviour, How to develop Positive Attitude? Developing the attitude of Gratitude.

Public Speaking: JAM, J2M, Presentations by Students on General Topics.

Quantitative Aptitude 16 hours

Number system, L.C.M and H.C.F, Problems on Ages, Averages, Time and work, Pipes and cisterns

Unit-II

Module-1:

No of Hours: 16 Hours

Introduction: Importance of Competitive Programming, Conditional Statements, Loops, Nested Loops. Strings: Declaration of strings, Accessing strings elements, operations

Practice Problems:

Week	Topic	Practice Problem Links
1	Introduction to Competitive Programming	**Codechef, Hackerrank,Codeforces, Leetcode, interviewbit…etc**

2	Conditional Statements-I	https://www.codechef.com/practice/tags/basic-programmingconcepts
		https://www.codechef.com/practice/tags/conditional-statements
3	Conditional Statements-II	https://www.codechef.com/practice/tags/basic-programmingconcepts
		https://www.codechef.com/practice/tags/conditional-statements
4	Loops-I	https://www.codechef.com/practice/tags/loops
		https://leetcode.com/tag/math/
		https://www.interviewbit.com/practice/#topics[]=math
		https://www.codechef.com/practice/tags/basic-maths
5	Loops-II	https://www.codechef.com/practice/tags/loops
		https://leetcode.com/tag/math/
		https://www.interviewbit.com/practice/#topics[]=math
		https://www.codechef.com/practice/tags/basic-maths
6	Nested Loops-I	https://www.codechef.com/practice/topics/math
		https://www.interviewbit.com/practice/#topics[]=math
		https://leetcode.com/tag/math/
7	Nested Loops-II	https://www.codechef.com/practice/topics/math
		https://www.interviewbit.com/practice/#topics[]=math
		https://leetcode.com/tag/math/
8	Strings-I	https://www.codechef.com/practice/topics/strings
		https://leetcode.com/tag/string/
		https://www.interviewbit.com/practice/#topics[]=strings
9	Strings-II	https://www.codechef.com/practice/topics/strings
		https://leetcode.com/tag/string/
		https://www.interviewbit.com/practice/#topics[]=strings
10	String Matching Algorithms	https://www.codechef.com/practice/topics/strings
		https://leetcode.com/tag/string/
		https://leetcode.com/tag/string-matching/
		https://www.interviewbit.com/practice/#topics[]=strings

Module-2:

No of Hours: 16 Hours

Arrays: Basics on Arrays: Creating elements, converting a string into array of characters, converting a number into array of character, accessing array elements **Recursion:** Definition, examples and how recursion differs from iteration.

Practice Problems:

Week	Topic	Practice Problem Links
11	Arrays - I	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=array
		https://leetcode.com/tag/array/
		https://www.codechef.com/practice/topics/arrays
		https://www.interviewbit.com/practice/#topics[]=arrays
12	Arrays - II	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=array
		https://leetcode.com/tag/array/
		https://www.codechef.com/practice/topics/arrays

		https://www.interviewbit.com/practice/#topics[]=arrays
13	2D Arrays & Matrix	https://www.interviewbit.com/practice/#topics[]=two-pointers
		https://leetcode.com/tag/two-pointers/
		https://leetcode.com/tag/matrix/
		https://www.codechef.com/practice/tags/matrices
14	Recursion - I	https://leetcode.com/tag/recursion/
		https://www.codechef.com/practice/tags/recursion
15	Recursion - II & Assessment	https://leetcode.com/tag/recursion/
		https://www.codechef.com/practice/tags/recursion
		https://www.hackerearth.com/practice/basic-programming/recursion/recursion-and-backtracking/practiceproblems/
16	Recursion - III & Assessment - II	https://leetcode.com/tag/recursion/
		https://www.codechef.com/practice/tags/recursion

Total: 32 Hours

References:

1. https://leetcode.com
2. https://www.codechef.com
3. https://www.hackerearth.com
4. https://www.interviewbit.com
5. https://www.hackerrank.com

4th Semester

23IT304 Database Management Systems

3 0 0 3

Course Outcomes

Understand the fundamental concepts of data base and data models

1. Explain the use of Relational Algebra and integrity constraints in databases
2. Use SQL's Commands to handle the Database
3. Apply Normalization for schema refinement
4. Make use of the concept of transaction management and recovery system in databases
5. Outline Indexing concepts, different types of data

CO–PO Mapping

COs	PO1	PO2	PO3	PSO2
1	3	1	1	1
2	3	2	3	1
3	3	3	3	2
4	3	3	3	2
5	3	2	3	3
6	2	1	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I Introduction to DBMS and ER Model 12 Hours

DBMS Vs. File System, instance and schema, Data abstraction, Data independence, database users and database administrator, Database system structure, Introduction to Data Models (E-R Model, Relational Model, Hierarchical Model, Network Model, Object Oriented Data Model), Database Design Process, Entities, Attributes, Entity Sets, Relationships, Relationship Sets, Additional features of ER Model.

Applications of DBMS, Object Relational Data Model

Unit II Introduction to Relational Model and Basic SQL Queries 12 Hours

Relational Algebra Operations: Selection, Projection, Rename, Set Operators, Joins, Division, Examples of Relational Algebra Queries, Relational Calculus: Tuple Relational Calculus.

Integrity Constraints over Relations, Introduction to Views.

SQL Queries: Basic Structure, Set Operations, Aggregate Functions, Null values, Sub Queries, Group By And Having Clauses, Outer Joins.

Domain Relational Calculus, Query Optimization

Unit III 12 Hours

Normalization and Transaction Management

Introduction To Schema Refinement - Problems Caused By Redundancy - Decomposition - Problems Related To Decomposition - Functional Dependency - Closure of a Set of Fds - Attribute Closure - First - Second - Third Normal Forms – BCNF - Multi Valued Dependencies – Fourth Normal Form, Join Dependency, Fifth Normal Form

Transactions: Acid Properties of Transaction - Transaction States - Schedule: Serial Schedule - Concurrent Schedules - Anomalies Associated with Concurrent Schedules (RW - WR - and WW Conflicts) - Serializability - Conflict Serializability - and View Serializability.

EF Codd Rules, Domain Dependency

Unit IV 12 Hours

Locking, Recovery Systems, Indexing, Different Types of Data

Introduction to Lock Management-Lock Based Concurrency Control: 2pl-Strict 2pl-Concurrency without Locking: Timestamp–Based Concurrency Control, Optimistic Concurrency Control. Introduction to Aries - the Log - the Write-Ahead Log Protocol-Check Pointing Indexing: Types of Single-Level Ordered Indexes, Multilevel Indexes Different Types of Data: Structured, Semi-Structured and Unstructured Data *Heap File, Hash File Organizations*

Total: 48 Hours

Textbook (s)

1. Elmasri & Navatha, Fundamentals of Database Systems, Pearson Education, 7thEdition,2016
2. SilberschatzKorth, Database System Concepts, McGraw hill, 7thEdition,2020

Reference (s)

1. SorayaSedkaoui, Data Analytics and Big Data, Wiley, 1st Edition,2018.
2. PeterRob&CarlosCoronel,DatabaseSystemsdesign,ImplementationandManagement,9th Edition, 2010.
3. Raghurama Krishnan & Johannes Gehrke, Database Management Systems, TATA McGraw-Hill, 3rd Edition,2003
4. C.J.Date, An Introduction to Database Systems, Pearson Education, 8th Edition,2006

23IT403 Operating Systems Course Outcomes 3 0 0 3

1. Understand computer resources and operating system management.
2. Analyze various CPU Scheduling Algorithms for Process Management.
3. Examine process synchronization and coordination of operating system.
4. Analyze the Main Memory Management and allocation strategies.
5. Identify the use of Virtual Memory management policies with respect to storage management.
6. Identify the need of File-System Interface and I/O Systems.

CO–PO Mapping

COs	PO1	PO2	PO12	PSO2
1	2	1	2	2
2	3	2	2	3
3	3	3	2	3
4	3	3	2	2
5	2	2	2	2
6	2	2	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I Operating-Systems Overview and Process Management 12 Hours

Operating-System Overview: Computer-System Organization and Architecture, Operating-System Structure, Operating-System Operations & Services, System Calls & its types.

Threads: Multi Core Programming, Multithreading Models, Thread Scheduling algorithms.

Process Management: Process Concepts, Process Scheduling Criteria, Scheduling Algorithms and evaluation.

Thread issues, Multilevel Queue, Multilevel feedback Queue Scheduling.

Unit II Inter Process Communication Mechanism 12 Hours

Process Synchronization: Cooperative process, the Critical Section Problem, Peterson’s Solution, Synchronization Hardware, Semaphores, Classical Synchronization problems.

Deadlocks: Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance & Detection, Recovery from Deadlock. *Monitors, Synchronization Examples.*

Unit III Memory Management 12 Hours

Main Memory: Contiguous Memory allocation, Swapping, Segmentation, Paging, Segmented paging, Multilevel paging.

Virtual Memory Management: Demand Paging, Page Replacement algorithms, Allocation of Frames. *Structure of page table, Thrashing, Memory-Mapped Files.*

Unit IV File System Interface &I/O Systems 12 Hours

Mass-Storage Structure: Disk structure, Disk Scheduling, Disk management, Raid Structure.

File System: Access Methods, Directory Structures, Allocation Methods, Free-Space Management.

I/O Systems: I/O hardware Application of I/O Interface, Kernel I/O Sub-System.

File Sharing, File System Recovery, Transforming I/O Requests to Hardware Operation.

Total: 48 Hours

Textbook (s)

1. Operating System Concepts, Abraham Silberschatz, Greg Gagne, Peter B. Galvin, 9th Edition, Wiley, 2016.
2. Operating Systems, Harvey M. Deitel,Paul J. Deitel, David R. Choffnes, 3rd Edition, Pearson Prentice Hall,2004.

Reference (s)

1. Operating Systems: Internals and Design Principles, William Stallings, 7th Edition, Pearson Prentice Hall,2013.
2. Operating systems: A Concept based Approach, D. M. Dhamdhere, 2nd Edition, TMH,2006.
3. Operating System: A Design Approach, Crowley, 1st Edition, TMH,2001.
4. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI,2009.

23CS403 Computer Organization and Architecture

3 0 0 3

Course Outcomes

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

1. Interpret the functional architecture of computing systems.(Understand).
2. Summarizethetypesofinstructionanditsmicrooperationwithaddressingmodes(Understand)
3. Identify various arithmetic operations on fixed, floating point numbers and its representation (Apply)
4. Illustrate the concepts of control unit design and I/O processor(Understand)
5. Understand the memory hierarchy concepts(Understand)
6. Describe concept of parallelism and types of hazard(Understand)

CO-PO Mapping

COs	PO1	PO2	PO12	PSO1
1	3	2	1	2
2	3	2	1	2
3	3	3	1	2
4	3	2	1	2
5	3	2	1	2
6	3	2	1	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Unit I Overview & Microoperation 13 Hours

Components of a computer system – Performance measures - Classifying Instruction Set Architecture-Representing instructions -Micro operation – Logical operations – Shift operations - instruction codes - Computer Registers instruction –memory Reference instruction –Input-Output Reference instruction - Instruction cycle -Addressing and addressing modes.

Trends in Technology-Arithmetic micro-operations

Unit II Arithmetic Operations 11 Hours

ALU - Addition and subtraction with Signed Magnitude Data - Hardware Implementation – Multiplication – Hardware Implementation for Signed Magnitude Data – Division - Hardware Implementation for Signed Magnitude Data – Divide Overflow - Floating Point operations – Parallelism and Computer Arithmetic: Sub wordParallelism.

BCD Adder-BCD Subtraction

UNIT III Control Unit and Memory Systems 11 Hours

Basic MIPS implementation – Building data path – Control Implementation scheme – Memory hierarchy – Cache basics – Measuring and improving cache performance - Virtual memory- Input/output system-programmed I/O-DMA and Interrupts-I/O processors

Stack organization-RISC Vs CISC Architecture

Unit IV Parallelism 13 Hours

Instruction-level-parallelism – Parallel processing challenges – Flynn's classification – Multicore processors- Pipelining – Arithmetic pipeline –Instruction pipeline -Pipelined data path and control – Handling Data hazards & Control hazards – Exceptions.

Vector processing –single processor Vs parallel processor

Total: 48 Hours

Text Book(s):

1. DavidA.PattersonandJohnL.Hennessey,“Computerorganizationanddesign:Thehardware /software interface‟, Morgan Kauffman / Elsevier, Fifth edition, 2014.
2. M.Morris Mano,” Computer System Architecture”, 3rdedition, Pearson/PHI,2092.

Reference(s):

1. V.CarlHamacher, Zvonko G. Varanesic and Safat G. Zaky, “Computer Organization “, 6thedition, McGraw-Hill Inc,2012.
2. William Stallings “Computer Organization and Architecture, Seventh Edition, Pearson Education, 2007.
3. Andrew S Tanenbaum “Structured Computer Organization “, 5thedition, Pearson/PHI,2007

23MA404 - Probability and Statistics using Python

3 0 2 4

Course Outcomes:

1. List the basic data structures of Python and various libraries of Python.
2. Illustrate basic probability axioms and apply Bayes' theorem related to engineering problems.
3. Differentiate between various distributions of random variables on given data.
4. Examine given dataset based on descriptive statistics.
5. Compare the null or alternative hypotheses using the suitable test statistic.
6. Choose appropriate curve fitting and correlation methods for a given data

CO-PO Mapping

CO	PO1	PO2	PO4	PO12
1	2	3	1	2
2	3	2	3	2
3	3	2	3	1
4	3	1	3	3
5	3	1	2	1
6	3	1	3	3

3- Strongly linked | 2-Moderately linked | 1-Weaklylinked

Unit I **12+8 Hours**

Fundamental of Python Programming

Strings, Files, Lists, Dictionaries, Tuples. **Introduction to NumPy:** Creating Numpy Arrays, Numpy Data Objects, dtype, Numerical Operations. Numpy Arrays: Concatenating, Flatten, reshape, adding Dimensions, Vector Stacking,Pandas DataFrame, Accessing and Changing values of DataFrames, Pandas Pivot, groupby **Introduction to Pandas:** Reading and Writing Data in Pandas, Multi-level Indexing, Data Visualization in Pandas *Objects and Classes*

Practical Components

1. Experiments on Basic experiments on fundamentals on python
2. Exploratory Data Analysis using NumPy
3. Exploratory Data Analysis using Pandas
4. Exploratory Data Analysis using Matplotlib

Unit II **12+6 Hours**

Random Variables and Probability Distributions

Introduction to Probability: Events, Axioms of Probability and Probabilities of events; Conditional Probability, Bayes Theorem. Discrete and Continuous random variables; Expectation and Variance, Distribution of a Random Variables: **Discrete and Continuous Distributions-** Bernoulli, Binomial, Poison and Normal Distributions (without proofs); Central Limit Theorem (without proof). *Gaussian distribution*

Practical Components

1. Simulation of random experiments and plot the empirical probability of each event, alongside the theoretical probability.
2. Python script to find conditional probability of a given event.
3. Simulation of various probability distributions

Unit III **12+10 Hours**

Sampling Theory and Testing of Hypothesis

Sampling: Population and sample, parameter and statistic, standard error. Parameter estimations, Confidence Intervals, Unknown Standard Deviation. **Hypothesis Testing**: Null and alternative hypothesis, level of significance, Type I and Type II errors, one tail and two-tail tests. Hypothesis testing concerning one mean and proportions, two means –Proportions and their differences using Z-test, Student's t-test, F-test, Chi-square test.

One-way ANOVA

Practical Components

1. Python Modules for Descriptive Statistics

2. Write a python program to perform z-test, t- test for testing the Null hypothesis for single mean
3. Write a python program to perform z-test, t- test for testing the Null hypothesis for difference of means
4. Write a python program to explore problems in F-test.
5. Write a python program to perform chi square for determine whether there is a significant association between the two variables

Unit IV **12+8 Hours**

Curve Fitting, Correlation & Regression

Curve Fitting: Introduction-fitting a straight line-second degree curve-exponential curve by method of least squares. **Correlation:** Simple correlation, Karl Pearson's Correlation coefficient and Spearman's rank correlation. Simple regression, multiple regression.

Covariance

Practical Components

1. Develop a straight line curve to fit some real world observations
2. Develop a exponential curve to fit some real world observations
3. Explore problems in Correlation with sample data
4. Explore problems in Regression with sample data

Total: 48+32 Hours Text

Books:

1. Probability & Statistics by T.K.V.Iyengar&B.Krishna Gandhi & Others, S.Chand (2016) 6th edition.
2. E. Walpole, R. H. Mayers, S. L. Mayers and K. Ye, (2012), Probability and Statistics for Engineers and Scientists,9th Edition, Pearson Education, ISBN: 978-8-131-71552-9.
3. Wes McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, 2nd Edition, O'Reilly Media, Inc.

Reference Books:

1. Michael Baron, Probability and Statistics For Computer Scientists, 2nd Edition, CRC Press 2. Sheldon M. Ross, (2011), Introduction to Probability and Statistics for Engineers and Scientists, 4th Edition, Academic Foundation, ISBN: 978-8-190-93568-5.

23ML405 Foundations of Machine Learning

3 0 1 3

Course Outcomes

1. Understand the pipeline of machine learning to apply on real world problems
2. Understand the need of Similarity and Dissimilarity between features in data
3. Importance of Pre-processing techniques in real-time
4. Identify the characteristics of various Regression models in real-world
5. Identify the characteristics of various supervised learning and ensemble models
6. Identify the characteristics of various algorithms of unsupervised learning

COs–POs Mapping

CO	PO1	PO2	PSO1	PSO2
1	3	2	3	3
2	3	2	3	3
3	2	2	3	3
4	2	3	3	3
5	2	3	3	3
6	3	3	3	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

UNIT I **12 hours**

Introduction: Relation among AI, ML and Data Science, Importance of Machine learning, Types of machine learning ,ML Use cases; Phases of ML: Training, Testing and Validation Machine learning Pipeline; Evaluation Machine Learning Models: Error Analysis, Performance Indices

Data Exploration: Objectives of Data Exploration, Forms of Data (Structured, Semi Structured, Unstructured), Datasets (data objects and types of attributes/fields), Characteristics of Datasets and corresponding Statistical Measures.

Discriminant analysis, LDA

UNIT II **12 hours**

Data Similarity/Dissimilarity: Data Matrix, Dissimilarity Matrix, Proximity/Similarity Measures for various types of data (nominal, ordinal, binary, numerical, etc.), Dissimilarity for Attributes of Mixed Types, Cosine Similarity; **Data Preprocessing:** Preprocessing Pipeline, Forms of Preprocessing, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Discretization.

Distance based similarity (Euclidean distance, Jaccard Similarity)

UNIT III **12 hours**

Regression (forecast a given numerical quantity): Linear Regression, Logistic Regression (Regression for Classification), Regularization;

Classification: Classification Principles, Classification Model Evaluation Metrics (Confusion Matrix, ROC and AUC, Lift Covers), Classification using Decision Trees, Bayesian Classification, Distance based Classifier(k-NN), Support Vector Machines (SVM), ANN-based Classification

LASSO and Ridge Regression

UNIT IV **12 hours**

Ensemble Learning: Conditions for Ensemble Modeling, Ensemble Techniques (Voting, Bagging, Boosting and Random Forest);

Clustering: Clustering Principles, Applications/Purpose of Clustering- Clustering for description/preprocessing/classification, Types of Clustering, Clustering Evaluation Parameters, Clustering Algorithms (k-Means, Agglomerative/Divisive, DBSCAN and Self Organizing Maps) and Evaluation Metrics;

Data Science Tools: Examples.

ADABoost, XGBoost

Textbooks:

1. Stephen Marsland, "Machine Learning **-**An Algorithmic Perspective ", CRC Press, 2009
2. Jlawei Han & Micheline Kamber, "**Data Mining: Concepts and Techniques**", 3rd Edition, Morgan Kaufmann Publishers, 2012
3. Pang-Ning Tan et al., "**Introduction to Data Mining**", 2nd Edition, Pearson Publications, 2019.

References:

1. Tom M. Mitchell, "Machine Learning ",Tata McGraw Hill, 1997
2. Machine Learning" First Edition 2016,Packt Publishing Ltd.

23IT308 Database Management Systems Lab

0 0 3 1.5

Course Outcomes

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

1. Demonstrate ER Modeling concepts to design the Database
2. Apply integrity constraints on a database
3. Make use of DDL, DML, DCL, TCL commands in creation and manipulation of Database
4. Utilize sub queries to make the complex queries more readable
5. Implementation of database queries using PL/SQL
6. Experiment with triggers to maintain the referential integrity of data

CO-PO Mapping

COs	PO4	PSO2
1	3	3
2	3	2
3	3	3
4	3	3
5	3	3
6	2	2

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Experiment 1 Application: 1

"Established in 1977, "Roadway Travels" has been actively operating for decades, providing connectivity across various destinations in India through a fleet of several buses. Headquartered in Hyderabad, the company now aims to modernize its operations in the following key areas:

1. Reservation System:

Managing reservations is the responsibility of the booking office. Customers can reserve their seats up to 60 days in advance, utilizing either cash or credit payment methods. In instances where tickets are unavailable, customers receive wait-listed tickets, which can be confirmed once cancellations occur.

2. Ticketing:

The booking office handles ticketing processes, including sales and issuance. Both cash and credit payments are accepted for this purpose. **3. Cancellation and Modification:**

Cancellation services are also facilitated at the booking office. Applicable cancellation charges will be levied as per company policy. Furthermore, modifications to reservations can be processed within this domain. Notably, wait-listed tickets that remain unconfirmed are eligible for full refunds." The Following are the entities:

Bus (Busno, Source, Destination, coachtype)
Reservation (PNRNo,Busno, NoofSeats, JourneyDt, ContactNo, Address)
Ticket (TicketNo, BusNo, JourneyDt, Source, Destination, Dept_Time, Age, Sex
Passenger (TicketNo, PNRNo, Name, ContactNo, Age, Sex)
Cancellation (PNRNo, JourneyDt, SeatNo, ContactNo)

Application: 2

Online Learning Platform: An online learning platform requires an ER model to handle courses, students, instructors, and enrollments. Courses have titles, descriptions, and durations. Instructors have names, expertise, and contact details. Students have profiles with names, email addresses, and student IDs.
Enrollments link students to courses, including enrollment dates.

Entities and Attributes:

Course (Course ID (Primary Key), Title, Description, Duration (in weeks or hours)Level (Beginner, Intermediate, Advanced), Category (e.g., Programming, Mathematics, Language), Price, Language, Thumbnail Image URL,)
Instructor (Instructor_ID (Primary Key), Name, Bio, ContactInfo (Email, Phone), Expertise (Areas of specialization), ProfilePictureURL)
Student (Student_ID (Primary Key), FirstName. LastName, Email, DOB, Address, Contact)
Enrollment(Enrollment_ID (Primary Key), Course_ID (Foreign Key referencing Course), Student_ID (Foreign Key referencing Student), EnrollmentDate, Status (Enrolled, Completed, Dropped, etc.))
Lesson(Lesson_ID (Primary Key), Course_ID (Foreign Key referencing Course), Title
Order (Sequence within the course), Duration (in minutes or hours), Content (Text, Video URL, PDF URL, etc.))

Quiz(Quiz_ID (Primary Key), Course_ID (Foreign Key referencing Course), Title, Number of Questions, Passing Score)
Question(Question_ID (Primary Key), Quiz_ID (Foreign Key referencing Quiz), Question Text, Question Type (Multiple Choice, True/False, Short Answer, etc.))
Answer(Answer_ID (Primary Key), Question_ID (Foreign Key referencing Question), AnswerText, Correct_Answer_Indicator)

Application: 3
Music Streaming Service: A music streaming service needs an ER model to organize its content and user interactions. The system should handle songs, albums, artists, playlists, and users. Songs have titles, durations, and release years. Albums include album names, release dates, and cover art. Artists have names, genres, and biographies. Users have profiles with names, email addresses, and subscription details. Playlists contain songs and are created by users.

Application: 4
Company Database: The company is organized into departments. Each department has a unique name, a unique number, and a particular employee who manages the department. We keep track of the start date when that employee began managing the department. A department may have several locations. A department controls several projects, each of which has a unique name, a unique number, and a single location. We store each employee's name, social security number, address, salary, sex, and birthdate. An employee is assigned to one department but may work on several projects, which are not necessarily controlled by the same department. We keep track of the number of hours per week that an employee works on each project. We also keep track of the direct supervisor of each employee. We want to keep track of the dependents of each employee for insurance purposes. We keep each dependent's name, sex, birthdate, and relationship to the employee.
Entities are:
Company Database:
EMPLOYEE (SSN, EName, Address, Sex, Bdate, Salary, SuperSSN, DNo)
DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate)
DLOCATION (DNo,DLoc)
PROJECT (PNo, PName, PLocation, DNo)
WORKS_ON (SSN, PNo, Hours)
DEPENDENT (Essn, Dependent_name, Sex, Bdate, Relationship)

Apply ER Model for the above applications and identify the relations present in each application represent them in ER diagram.& Convert ER model to Relational Model.
Note: Students need to form a group minimum of 3 students and maximum of 5, and should design at least two applications into ER model by using EdrawMax software and convert them into relational model

Experiment 2
Creating users, roles and grant privileges to users and relations.

Experiment 3
Create and alter tables for Company Database in SQL using Integrity Constraints.
Integrity Constraints are CHECK, NOT_NULL, DEFAULT, UNIQUE, PRIMARY KEY, FOREIGN KEY.
Use atleast one constraint in one relation to understand the usage of constraints on database.
Schema:

```
CREATE TABLE employee ( Fname varchar (15) NOT NULL, Minit varchar(1), Lname varchar(15) NOT NULL, SSN varchar(9) NOT NULL, Bdate date, Address varchar(30), Sex varchar(1), Salary float, Super_ssn varchar(9), Dno int NOT NULL, PRIMARY KEY (ssn), FOREIGN KEY (Super_ssn) REFERENCES employee(ssn));

CREATE TABLE department (Dnamevarchar(15) NOT NULL, Dnumber int NOT NULL, Mgr_ssn varchar(9) NOT NULL, Mgr_start_date date, PRIMARY KEY (Dnumber), UNIQUE(Dname), FOREIGN KEY (Mgr_ssn) REFERENCES employee(ssn));

CREATE TABLE dept_locations (Dnumber int NOT NULL, Dlocationvarchar(15), PRIMARY KEY (Dnumber, Dlocation), FOREIGN KEY (Dnumber) REFERENCES department (Dnumber));
```

CREATE TABLE project (Pnamevarchar(15) NOT NULL, Pnumber int NOT NULL, Plocation varchar(15), Dnum int NOT NULL, PRIMARY KEY (Pnumber), UNIQUE (Pname), FOREIGN KEY (Dnum) REFERENCES department (Dnumber));

CREATE TABLE works_on (Essnvarchar(9) NOT NULL, Pno int NOT NULL, Hours float NOT NULL, PRIMARY KEY (Essn, Pno), FOREIGN KEY (Essn) REFERENCES employee(ssn), FOREIGN KEY (Pno) REFERENCES project(Pnumber));

CREATE TABLE dependent (Essnvarchar(9) NOT NULL, Dependent_name varchar(15) NOT NULL, Sex varchar(1), Bdate date, Relationship varchar(8), PRIMARY KEY (Essn, Dependent_name), FOREIGN KEY (Essn) REFERENCES employee(ssn));

Experiment 4

Execute basic SQL queries on Employee Database
To Create a database called Employee which consists of EMP, Dept, Salgrade, Project, Works_on and Dependent tables as relations in the database use Data Definition Language command (Create, Alter). Run sql queries on database using DML commands like Insert, Update, Delete and for retrieval of data from database use Select statement.
Employee Database:
EMP (Empno, Ename, Job, Mgr, Hiredate, sal, comm, deptno)
DEPT (Deptno, Dname, Location)
SALGRADE(Grade, Losal, Hisal)
PROJECT (Pno, Pname, Plocation, Deptno)
WORKS_ON(Empno, Pno, Hours)
DEPENDENTS(EMPNO, DependentName, Sex, Bdate, Relation)

4.1 Insertion:
Insert rows into table by using datatypes like varchar, int, date, float etc.,
Syn: INSERT INTO employee (Fname, Minit, Lname, SSN, Bdate, Address, Sex, Salary, Super_ssn, Dno) values (‘James’, ‘E’, ‘Borg’, 888665555, ’10-NOV-1937’, ‘430 Stone, Houston, TX’, ‘M’, 55000, NULL, 1); 4.2 Retrieval (Set operations)
Syn: Select Name from EMP Union Select Salary from EMP;
4.3 Updating
Update rows by changing the values in a row using update command syntax.
4.4 Deletion
Delete an entire row or multiple rows from the table using this command

Experiment 5

Implement the SQL operations on Employee Database created in Experiment 4. Operations are: In, Any, All, Between, Like, Exists, Group by, Order by, having.

Employee Database:
EMP (Empno, Ename, Job, Mgr, Hiredate, sal, comm, deptno)
DEPT (Deptno, Dname, Location)
SALGRADE (Grade, Losal, Hisal)
PROJECT (Pno, Pname, Plocation, Deptno)
WORKS_ON (Empno, Pno, Hours)
DEPENDENTS (EMPNO, DependentName, Sex, Bdate, Relation)

Experiment 6

Implementing the concepts of Rollback – commit, checkpoints, and Views.
Rollback – commit, checkpoints:
Create Teacher table with the following fields(Name, DeptNo, Date of joining, DeptName, Location, Salary)

1. Insert five records
2. Give Increment of 25% salary for Mathematics Department .
3. Perform Rollback command

4. Give Increment of 15% salary for Commerce Department
5. Perform commit command.

Views:

To create different views, use the Emplpyee Database

1. Create a view to display Hiredate of all employees.
2. Create a view to display no of working hours of all employees with Plocation.

Experiment 7

Apply different types of joins availablein SQL on company database created in Experiment 4. Types of Joins:

- Inner Join
- Outer Join
- Letf Join
- Right Join

Experiment 8

Apply subqueries and nested queries on company database (refer to experiment 4).

Experiment 9

Implement built in functions in oracle (Numeric, Aggregate functions) on Dual table(provided by Oracle) and Employee database.

Numeric Functions: ABS, CEIL, FLOOR, SQRT etc.,

Aggregate Functions: MIN, MAX, SUM, COUNT, AVG

Experiment 10

Implement String handling and Conversion functions on Dual table (provided by Oracle) and Employee database.

String Functions: Lowercase, Uppercase, Initcap, Rtrim, Ltrim, Lpad, Rpad etc., Conversion Functions: TO_CHAR(), TO_DATE(), TO_NUMBER().

Experiment 11

Sql Practice links from online platforms:

Employee Bonus	https://leetcode.com/problems/employee-bonus/
Big Countries	https://leetcode.com/problems/big-countries/
Employees Earning more than their managers	https://leetcode.com/problems/employees-earning-more-thantheir-managers/
Managers with at least 5 direct reports	https://leetcode.com/problems/managers-with-at-least-5-directreports/
Second Highest Salary	https://leetcode.com/problems/second-highest-salary/
Count Salary Categories	https://leetcode.com/problems/count-salary-categories/

Triangle	https://www.hackerrank.com/challenges/what-type-of triangle/problem?isFullScreen=true
Average each Population of country in a continent	https://www.hackerrank.com/challenges/average-population-of-eachcontinent/problem?isFullScreen=true
TopEarners	https://www.hackerrank.com/challenges/earningsofemployees/problem?isFullScreen=true
Occupations	https://www.hackerrank.com/challenges/occupations/problem?isFull
New Company	https://www.hackerrank.com/challenges/theompany/problem?isFullScreen=true
15Days of Learning SQL	https://www.hackerrank.com/challenges/15-days-of-learningsql/problem?isFullScreen=true

Experiment 12

S

Experiment 13

Implement PL/SQL operations on Employee Database to retrieve one row as output.

Experiment 14

Implementing functions, stored procedures on company database using PL/SQL

Experiment 15

Implement cursors on Employee database using PL/SQL to retrieve multiple rows as output.

Experiment 16

Creating triggers using PL/SQL

E-RESOURCES AND OTHER DIGITAL MATERIAL

1. Prof Arnab Bhattacharya IIT Kanpur, SQL Introduction https://nptel.ac.in/courses/106104135/10
2. Prof Arnab Bhattacharya IIT Kanpur SQL: Updates, Joins, Views, and Triggers
3. https://nptel.ac.in/courses/106104135/11

List of Augmented Experiments

1. Inventory control management System
2. College Management System
3. Hospital management System
4. Library management System
5. Payroll management System
6. Health care organization Management System
7. Restaurant Management System
8. Blood Donation Management System
9. Art Gallery Management System

10. Hotel Management System
11. School Management System
12. Salary Management System
13. Wholesale Management System
14. Time Table Management System
15. Website Management **Reading Material (s)**

1. Database Management Systems Lab Manual, Department of CSE, GMRIT,Rajam

23ML407 Foundations of Machine Learning Lab 0 0 3 1.5

Course Outcomes

1. Demonstrate data formats and linear algebra matrices operation
2. Analyzing similarities/dissimilarities and visualizing data importance through plots
3. Demonstrate regression techniques on various datasets
4. Demonstrate Classification techniques on various datasets
5. Demonstrate Clustering techniques on various datasets
6. Build and evaluate the performance of various machine learning models **CO–PO Mapping**

CO	PO3	PO4	PO5	PSO1
1	2	3	2	1
2	2	3	3	2
3	1	3	1	3
4	2	2	2	3
5	2	2	2	3
6	1	1	2	3

3–Strongly linked | 2–Moderately linked | 1–Weakly linked

Experiments:

1. Linear Algebra Operations
2. Data Formats, Processing& Transformation
3. Obtaining Statistical Properties of data
4. Basic Experiments on Data Visualization, Interpretation, and Identification of outliers
5. Single Valued Decomposition and PCA
6. Optimization Techniques
7. Regression and variants (Prediction)
8. Regression with regularization (Prediction/classification)
9. Decision Tree Induction, Rule Generation and Classification
10. Implementation of K-Nearest Neighbor Classifier
11. Implementation of Bayesian Classifier
12. Implementation of Ensemble models
13. Implementation of K-Means Clustering
14. Implementation of Hierarchical clustering
15. Implementation of DBSCAN Clustering
16. Build, Train, Test and Deployment on cloud environment (e.g.AWS/Google Cloud)

Augmented Experiments:

Classification:

1. Design and Deploy a machine learning model for health care System Sample Datasets:

a. https://www.kaggle.com/datasets/rashikrahmanpritom/heart-attack-analysis-predictiondataset

b. https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000 2. Design and Develop a machine learning model for Business Analytics Sample Datasets:

a. https://www.kaggle.com/datasets/arashnic/hr-analytics-job-change-of-data-scientists

3. Design and Develop a machine learning model for Education institutions Sample Datasets:

a. https://www.kaggle.com/datasets/spscientist/students-performance-in-exams
4. Design and Develop a machine learning model for TV Shows Sample Datasets:
a. https://www.kaggle.com/datasets/shivamb/netflix-shows
5. Design and Develop a machine learning model for E-Commerce applications Sample
Datasets:
a. https://www.kaggle.com/datasets/shivamb/netflix-shows
6. Design and Develop a machine learning model Supermarket store applications Sample
Datasets:
a. https://www.kaggle.com/datasets/surajjha101/stores-area-and-sales-data Classification/Clustering:
7. Design and Develop a machine learning model for Drugs Review Sample
Datasets:
a. https://archive.ics.uci.edu/ml/datasets/Drug+Review+Dataset+%28Druglib.com%29
8. Design and Develop a machine learning model for Plants Sample
Datasets:
a. https://archive.ics.uci.edu/ml/datasets/Plants
9. Design and Develop a machine learning model for TV News Channel Commercial Detection Sample
Datasets:
a. https://archive.ics.uci.edu/ml/datasets/TV+News+Channel+Commercial+Detection+Dataset
10. Design and Develop a machine learning model for SMS Spam Collection Sample
Datasets:
a. https://archive.ics.uci.edu/ml/datasets/SMS+Spam+Collection

23ESX01 Employability Skills I

0 0 2 2

Course Outcomes

1. Demonstrate oral communication and writing skills as an individual to present ideas coherently
2. Develop life skills with behavioral etiquettes and personal grooming.
3. Assess analytical and aptitude skills.
4. Develop algorithms for engineering applications.
5. Solve engineering problems using software.
6. Utilize simulation tools for testing

COs – POs Mapping

COs	PO1	PO2	PO5	PO 8	PO10	PO12
CO1					3	2
CO2				1	2	2
CO3	2	1		2		
CO4	2		2			
CO5	2		2			
CO6	2		2			

3–Strongly linked | 2–Moderately linked| 1–Weakly linked

Unit-I Communication Skills 16 hours

Building Confidence: Fear? Steps to Overcoming the Fear of Public Speaking? Self Esteem: Definition? Types of Self Esteem, Causes of Low Self Esteem, Merits of Positive Self Esteem and Steps to build a positive Self Esteem. Group Discussions (Practice): GD? GD Vs Debate, Overview of a GD , Skills assessed in a GD, Dos & Don'ts, & Conducting practice sessions (SimpleTopics).

Motivational Talk: Team Work: Team Vs Group? Stages in Team Building, Mistakes to avoid and Lessons to Learn (Through Stories or Can be a Case Specific)

Quantitative Aptitude 16 hours

Percentages, Profit and loss, Mixtures and Allegations, Simple Interest, Compound Interest

Unit-II Competitive Programming Phase – II 16 Hours

Module-1:

Fundamental mathematical concepts: Basic Mathematics, Number Theory, Geometry - Bit manipulation - Twopointer technique. **Practice Problems:**

Week	Topic	Practice Problem Links
1	Math-I	https://www.codechef.com/practice-old/topics/math
		https://leetcode.com/tag/math/
		https://www.codechef.com/practice-old/tags/mathematics
		https://www.interviewbit.com/courses/programming/math
2	Math-II	https://www.hackerrank.com/domains/mathematics
		https://www.codechef.com/practice-old/tags/mathematics
		https://www.hackerearth.com/practice/math/number-theory/basicnumber-theory-1/practice-problems/
3	Number Theory	https://www.codechef.com/practice-old/tags/number-theory
		https://leetcode.com/tag/number-theory/

		https://www.hackerrank.com/domains/mathematics?filters%5Bsubdomains%5D%5B%5D=number-theory
4	Geometry	https://www.codechef.com/practice-old/tags/geometry
		https://www.hackerrank.com/domains/mathematics?filters%5Bsubdomains%5D%5B%5D=geometry
		https://leetcode.com/tag/geometry/
5	Bit Manipulation-I	https://www.codechef.com/practice-old/tags/bit-manipulation
		https://www.hackerearth.com/practice/basic-programming/bitmanipulation/basics-of-bit-manipulation/practice-problems/
6	Bit Manipulation-II	https://leetcode.com/tag/bit-manipulation/
		https://www.interviewbit.com/courses/programming/bit-manipulation
		https://www.hackerrank.com/domains/algorithms?filters%5Bsubdomains%5D%5B%5D=bit-manipulation
7	Two Pointer	https://www.interviewbit.com/practice/#topics[]=two-pointers
		https://leetcode.com/tag/two-pointers/
		https://www.codechef.com/practice-old/tags/two-pointers

Module-2:

Data Structures: Searching - Linked List - Stack – Queue – Graph Traversal Methods - Hashing Techniques.

8	Searching - I	https://www.codechef.com/practice/topics/binary-search
		https://leetcode.com/tag/binary-search/
		https://www.interviewbit.com/practice/#topics[]=binary-search
		https://www.codechef.com/practice/tags/searching
9	Searching - II	https://www.codechef.com/practice/topics/binary-search
		https://leetcode.com/tag/binary-search/
		https://www.interviewbit.com/practice/#topics[]=binary-search
		https://www.codechef.com/practice/tags/searching
10	Stacks	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=stacks
		https://leetcode.com/tag/stack/
		https://www.interviewbit.com/practice/#topics[]=stacks-and-queues
11	Queues	https://www.hackerrank.com/domains/datastructures?filters%5Bsubdomains%5D%5B%5D=queues
		https://leetcode.com/tag/queue/
		https://www.interviewbit.com/practice/#topics[]=stacks-and-queues
12	Linked List-I	https://www.hackerrank.com/domains/data-structures?filters%5Bsubdomains%5D%5B%5D=linked-lists
		https://leetcode.com/tag/linked-list/
13	Linked List-II	https://www.interviewbit.com/practice/#topics[]=linked-lists
		https://leetcode.com/tag/doubly-linked-list/
	BFS	https://www.codechef.com/practice-old/tags/bfs&two-pointers

14		https://leetcode.com/tag/breadth-first-search/
		https://www.hackerearth.com/practice/algorithms/graphs/breadth-firstsearch/practice-problems/
15	DFS	https://leetcode.com/tag/depth-first-search/
		https://www.hackerearth.com/practice/algorithms/graphs/depth-firstsearch/practice-problems/
16	Hashing	https://www.codechef.com/practice-old/tags/hashing
		https://leetcode.com/problemset/?page=1&topicSlugs=hash-function

References:

1. https://leetcode.com
2. https://www.codechef.com
3. https://www.hackerearth.com
4. https://www.interviewbit.com
5. https://www.hackerrank.com