BONAM VENKATA CHALAMAYYA ENGINERING COLLEGE
(AUTONOMOUS)
ODALAREVU – 533 210, Andhra Pradesh, India

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

I Year – I SEMESTER

1	20ES1T01	Basic Electrical and Electronics Engineering	3	0	0	3
2	20BS1T01	Linear Algebra & Differential Equations	3	0	0	3
3	20BS1T07	Applied Chemistry	3	0	0	3
4	20ES1T05	Programming for Problem Solving 1	3	0	0	3
5	20ES1L07	Computer Engineering Workshop Lab	2	0	2	3
6	20ES1L01	Basic Electrical and Electronics Engineering Lab	0	1	2	1.5
7	20BS1L03	Applied Chemistry Lab	0	0	3	1.5
8	20ES1L08	Programming for Problem Solving 1 Lab	0	0	3	1.5
	20HS1M01	*Induction Training	3	0	0	0
Total Credits			**16**	**1**		**19.5**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

I Year – II SEMESTER

S. No	Course Code	Subjects	L	T	P	Credits
1	20HS2T01	English	3	0	0	3
2	20BS2T02	Integral Transforms and Calculus	3	0	0	3
3	20ES2T04	Engineering Drawing	2	0	2	3
4	20BS2T05	Applied Physics	3	0	0	3
5	20ES2T 06	Programming for Problem Solving 2	3	0	0	3
6	20BS2L01	Applied Physics Lab	0	0	3	1.5
7	20ES2L09	Programming for Problem Solving 2 Lab	0	0	3	1.5
8	20HS2L01	English Communication Skills Lab	0	0	3	1.5
	20HS2M02	* Indian Constitution	2	0	0	0
Total Credits			**17**	**1**	**8**	**19.5**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

II Year – I SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20BS3T03	Probability and Statistics	3	0	0	3
2	20HS3T02	Managerial Economics and Financial Analysis	3	0	0	3
3	20ML3T01	Computer Organization	3	0	0	3
4	20ES3T04	Python Programming	3	0	0	3
5	20CS3T05	Digital Logic Design	3	0	0	3
6	20ML3L01	Unix and Shell Programming Lab	3	0	0	1.5
7	20ES3L07	Python Programming Lab	0	0	3	1.5
8	20CS3L08	R Programming Lab	0	0	3	1.5
9	20CS3S01	Skill Oriented Course - Java Script	1	0	2	2
	20BS3M03	Environmental Science	2	0	0	0
		Total	21	0	8	21.5

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

II Year – II SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20BS4T04	Numerical Methods and Complex Variables	3	0	0	3
2	20CS4T06	Java Programming	3	0	0	3
3	20AD4T02	Data Science	3	0	0	3
4	20CS4T07	Operating Systems	3	0	0	3
5	20CS4T08	Mathematical Foundations of Computer Science	3	0	0	3
6	20CS4L07	Java Programming Lab	0	0	3	1.5
7	20CS4L09	Operating Systems Lab	0	0	3	1.5
8	20AD4L02	Data Science Lab	0	0	3	1.5
9	20CS4S02	Skill Oriented Course - MongoDB	1	0	2	2
		Total	16	0	11	21.5

BONAM VENKATA CHALAMAYYA ENGINERING COLLEGE
(AUTONOMOUS)
ODALAREVU – 533 210, Andhra Pradesh, India

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

III Year – I SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20CS5T09	Computer Networks	3	0	0	3
2	20AD5T03	Machine Learning	3	0	0	3
3	20CS5T11	Data Base Management System	3	0	0	3
4	 20ML5E01 20CS5E02 20CE5E01 20ME5E01	**Open Elective- I** 1. Artificial Intelligence 2. Object Oriented Programming 3. Interior Space Design 4. Industrial Automation and Robotics	3	0	0	3
5	 20CS5D01 20CS5D03 20AD5D01 20AD5D02	**Professional Elective- I** 1.Computer Graphics 2.Software Engineering 3.Object Oriented Analysis and Design 4.DevOps/**MOOCs -1**	3	0	0	3
6	20CS5L12	DBMS Lab	0	0	3	1.5
7	20AD5L03	Machine Learning Lab	0	0	3	1.5
8	20CS5P01	Summer Internship	0	0	3	1.5
9	20AD5S01	Skill Advanced Course: Continuous Integration and Continuous Delivery using DevOps	1	0	2	2
10	20HS5M04	Foreign Language	2	0	0	0
		Total	18	0	11	21.5

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

III Year – II SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20CS6T12	Compiler Design	3	0	0	3
2	20CS6T13	Data Mining and Data Warehousing	3	0	0	3
3	20AD6T04	Deep Learning	3	0	0	3
4	20AD6D04 20AD6D05 20CS6D07 20AD6D06	**Professional Elective -II** 1. Design and Analysis of Algorithms 2. Software Project Management 3.Distributed Systems 4. Data Wrangling in Data Science/**MOOCs-2**	3	0	0	3
5	20CS6E03 20ME6E02 20CE6E02 20EC6E02	**Open Elective- II** **1.** Mean Stack Development **2.** Operations Research **3.** Green Building **4.** Internet of Things	2	0	2	3
6	20CS6L13	Compiler Design Lab	0	0	3	1.5
7	20CS6L14	Data Mining Lab	0	0	3	1.5
8	20AD6L04	Deep Learning with Tensorflow Lab	0	0	3	1.5
9	20CS6S04	Skill Advanced course: Game Development	1	0	2	2
10	20HS6M05	Universal Human Values	2	0	0	0
		Total	17	0	13	21.5

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
COURSE STRUCTURE – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

IV Year – I SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20CS7D09 20AD7D07 20AD7D08 20CS7D18	**Professional Elective-III** 1. Cryptography and Network Security 2 Reinforcement Learning 3. Social Media Analytics 4. Block Chain Technologies/**MOOCs-3**	3	0	0	3
2	20CS7D13 20CS7D16 20CS7D11 20AD7D09	**Professional Elective-IV** 1. Software Testing Methodologies 2. Computer Vision 3. Adhoc and Sensor Networks 4. NOSQL Databases/**MOOCs-4**	3	0	0	3
3	20AD7D10 20AD7D11 20AD7D12 20AD7D13	**Professional Elective-V** 1. Social Network Analysis 2. Recommender Systems 3. AI Chatbots 4. Data Visualization/**MOOCs-5**	3	0	0	3
4	20CS7E04 20CE7E03 20EC7E03 20EE7E03	**Open Elective -III** 1. Nature Inspired Computing Techniques 2. Disaster Management 3. Embedded Systems 4. Electrical Distribution Systems	2	0	2	3
5	20CS7E05 20EC7E04 20ME7E04 20CE7E04	**Open Elective- IV** 1. Secure Coding Techniques 2. 5G Communications 3. 3D Printing Technologies 4. Project Management	2	0	2	3
6	20HS7E01	**Humanity and Social Science Elective** 1. Resource Management 2. Industrial Management 3. Management Science 4. IPR and Patents	3	0	0	3
7	20HS7S01	English for Employability	1	0	2	2
8	20CS7P02	Industrial/Research Internship 2 months after III_I Evaluated during IV_I	0	0	6	3
		Total	17	0	12	23

IV Year II Semester

S.No	Course Code	Courses	L	T	P	Credits
1	20CS8P03	Major Project Work	0	0	24	12
		Total	0	0	24	12

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DETAILED SYLLABUS FOR
UG: B.Tech - ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
I Year – I SEMESTER

1	20ES1T01	Basic Electrical and Electronics Engineering	3	0	0	3
2	20BS1T01	Linear Algebra & Differential Equations	3	0	0	3
3	20BS1T07	Applied Chemistry	3	0	0	3
4	20ES1T05	Programming for Problem Solving 1	3	0	0	3
5	20ES1L07	Computer Engineering Workshop Lab	2	0	2	3
6	20ES1L01	Basic Electrical and Electronics Engineering Lab	0	1	2	1.5
7	20BS1L03	Applied Chemistry Lab	0	0	3	1.5
8	20ES1L08	Programming for Problem Solving 1 Lab	0	0	3	1.5
	20HS1M01	*Induction Training	3	0	0	0
Total Credits			**16**	**1**		**19.5**

I Year – II SEMESTER

S. No	Course Code	Subjects	L	T	P	Credits
1	20HS2T01	English	3	0	0	3
2	20BS2T02	Integral Transforms and Calculus	3	0	0	3
3	20ES2T04	Engineering Drawing	2	0	2	3
4	20BS2T05	Applied Physics	3	0	0	3
5	20ES2T 06	Programming for Problem Solving 2	3	0	0	3
6	20BS2L01	Applied Physics Lab	0	0	3	1.5
7	20ES2L09	Programming for Problem Solving 2 Lab	0	0	3	1.5
8	20HS2L01	English Communication Skills Lab	0	0	3	1.5
	20HS2M02	* Indian Constitution	2	0	0	0
Total Credits			**17**	**1**	**8**	**19.5**

BASIC ELECTRICAL AND ELECTRONICS ENGINEERING (20ES1T01)

UNIT – I Electrical Circuits:
Basic definitions - Types of network elements - Ohm's Law - Kirchhoff's Laws - Inductive networks - Capacitive networks – Series - Parallel circuits - Star-delta and delta-star transformations.

UNIT - II
Dc Machines:
Principle of operation of DC generator – EMF equation - Types of DC machine – Applications – Three point starter - Speed control methods of DC motor – Swinburne's Test.

UNIT – III
Transformers & **Machines**
Principle of operation and construction of single phase transformers – EMF equation – Losses – OC & SC tests – Efficiency- Applications.

Principle of operation and construction of alternators– Types of alternators – Principle of operation of synchronous motor – Applications.

UNIT IV
Rectifiers & Linear ICs:
PN junction diodes - Diode applications (Half wave and bridge rectifiers). Characteristics of operation amplifiers (OP-AMP) - applications of OP-AMPs (inverting, non-inverting).

UNIT V
Transistors:
PNP and NPN junction transistor, transistor as an amplifier- Transistor amplifier - Frequency response of CE amplifier

Text Books:
1. **Electrical Technology** by Surinder Pal Bali, Pearson Publications.
2. **Electronic Devices and Circuits**, R.L. Boylestad and Louis Nashelsky, 9th edition, PEI/PHI 2006.

Reference Books:
1.Electrical Circuit Theory and Technology by John Bird, Routledge Taylor &Francis Group
2. Basic Electrical Engineering by M.S.Naidu and S.Kamakshiah,TMH Publications
3.Fundamentals of Electrical Engineering by Rajendra Prasad, PHI Publications,2nd edition
4.Basic Electrical Engineering by Nagsarkar,Sukhija, Oxford Publications,2nd edition
5.Industrial Electronics by G.K. Mittal, PHI

Course Code: 20BS1T01

L	T	P	C
3	0	0	3

Course Objectives:

COB 1: To develop the basic mathematical skills those are imperative for effective Understanding of engineering subjects

COB 2: To equip the students with the methods of solving system of linear equations and the Concepts of Eigen values and Eigen vectors.

COB 3: To familiarize students with analytical methods to solve Ordinary differential equations.

Course Outcomes:

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

CO 1: Determine rank of the given matrix and is able to find the solution of given system of linear equations.(L3)

CO 2: Illustrate Eigen values, Eigen vectors and is able to understand their applications. (L4)

CO 3: Apply differential calculus to model rate of change in time of any physical phenomena and solve it.(L3)

CO 4: Apply appropriate analytical technique to model and solving higher order differential equations.(L3)

CO 5: Understand partial differentiation and able to apply in finding maxima and minima. (L2)

CO6: Find the solution of one-dimensional wave equation and heat equation(L3)

Unit I:- Matrices and Linear systems of equations

Rank of a Matrix-Echelon form-Normal form – Solution of linear system of equations homogenous and non-homogenous equations – Gauss elimination -Gauss Jordon- Gauss Seidal methods.

Applications: Finding the current in electrical circuits.

Unit II:- Eigen values - Eigen vectors

Eigen values - Eigen vectors– Properties – Cayley-Hamilton theorem (statement only) - Inverse and powers of a matrix by using Cayley-Hamilton theorem- Reduction to Diagonal form(symmetric) – Quadratic forms and nature of the quadratic forms – Reduction of quadratic form to canonical forms by orthogonal transformation

Unit III:- Differential equations of first order and first degree

Formation of D.E-Linear-Bernoulli-Exact-Reducible to exact differential equations

Applications: - Newton's Law of cooling-Law of natural growth and decay-Orthogonal trajectories

Unit IV:-Linear differential equations of higher order

Non-homogeneous equations of higher order with constant coefficients with Non homogeneous terms of the type e^{ax}, $\sin ax, \cos ax, polynomials\ in\ x, e^{ax}V(x), xV(x)$- Method of Variation of parameters.

APPLIED CHEMESTRY

Course Code: 20BS1T07 **L T P C**

3 0 0 3

Course objectives

COB:1 Recognize the relative energies of bonding and anti bonding molecular orbitals and crystal field stabilization energy.
COB:2 Understand the basic principles of spectroscopy where electromagnetic radiation interacts with Chemical substances.
COB:3 To impart the knowledge of water treatment, setting of cement and repair of concrete structures, Nano materials.

course outcomes

CO:1 Analyze microscopic chemistry in terms of atomic and molecular orbitals.
CO:2 Distinguish the range of the electromagnetic spectrum used for exciting different molecular energy levels in various spectroscopic techniques.
CO:3 Solve the numerical problems on hardness of water and explain water treatment methods and Principles.
CO:4 Outline galvanic cells, reference electrodes, electrochemical series, analyze the corrosion of metals and apply a suitable method of corrosion prevention.
CO:5 Summarize the methods of synthesis, properties and applications of nano materials.
CO:6 Understand the preparation, properties and applications of some polymers and elastomers.

UNIT-I **8 - HOURS**

Molecular Orbital Theory and Coordination Compounds:
Linear Combination Of Atomic Orbitals – Molecular orbital theory – homo nuclear diatomic molecules (H_2,He_2,He_2^+,Li_2,Be_2,B_2,C_2,N_2,O_2,O_2^+,O_2^-,F_2 and Ne_2). Application of Molecular orbital theory for bonding in metals.
Introduction to coordination compounds- crystal field theory and splitting of d-orbitals in octahedral, tetrahedral and square planar complexes, factors effecting crystal field splitting**.**

UNIT-II **10 - HOURS**

Spectroscopic techniques and applications:
UV: Introduction - electronic energy levels-electronic transitions-absorption laws-chromophores and auxochromes - bathochromic shift - hypsochromic shift- hyper chromic shift – hypochromic shift-Applications.
IR: Introduction - basic theory - selection rules - absorption of infrared radiation and molecular vibrations-Applications.
NMR: Introduction-spin active nuclei behaving as nuclear magnets- energy absorption and resonance-relaxation phenomenon-commonly used NMR solvents – chemical shift-factors influencing the chemical shift shielding and deshielding-Applications.

UNIT-III **9 - HOURS**

Water Technology:
Introduction - hardness of water - calcium carbonate equivalents - units of hardness - causes of hardness - types of hardness, disadvantages of hard water. Determination of hardness by EDTA.
Boiler troubles - priming and foaming, scale and sludge formation, boiler corrosion, caustic embrittlement - softening of hard water: lime - soda process, zeolite process and ion exchange process. Water purification by Reverse Osmosis for drinking and Industry.

UNIT-V **9 - HOURS**

Electrochemistry and Corrosion:
Introduction to electrochemistry galvanic cells, single electrode potential, standard electrode potential, electro chemical series, E.M.F of cell and its measurements, Nernst equation, electrolyte concentration cells, reference electrodes- Standard Hydrogen Electrode & Standard Calomel Electrode. Reversible and irreversible cells. Lithium Batteries for Electric Vehicles and electronic devices.
Corrosion:-
Chemical and electro chemical corrosion – mechanism - Galvanic corrosion, pitting corrosion –passivity - Factors influencing corrosion- Electrochemical Principles to control corrosion.

UNIT-V **19 - HOURS**

Chemistry of Engineering Materials:
Components of Portland cement, their chemical Composition and Properties, setting and hardening of cement, Cements for special purpose- Repair of concrete structures using special additives.
Nano Materials:- Introduction – sol-gel method - characterization by Brunauer –Emmett- Teller,Transmission Electron Microscopic methods-properties and applications of Nano materials, Fullerenes and Nanotubes**.**Preparation of Nanotubes.
High Polymers:
Introduction - addition and condensation polymerization - effect of polymer structure on properties. Thermoplastics and Thermosetting plastics - Preparation, Properties and Applications of PVC, Polycarbonates and Bakelite. Biodegradable Polymers. Advantages of plastics and how to reuse, recycle and manage/dispose of plastic waste.
Elastomers:-
Natural rubber - vulcanization of rubber - preparation, properties and applications of Buna-S, Buna-N, Thiokol and Polyurethanes**.**

Text Books:
i) Engineering chemistry by jain and jain: Dhanpat Rai publishing company
ii) Engineering chemistry –foundamentals and applications by SHIKHA AGARWAL-Cambridge University press
iii) Engineering chemistry by R.P.Mani,K.N.Mishra,B.Ramadevi,V.R.reddy Cengage Learning
Reference books:
i) University chemistry by B.H. Mahan
ii) Engineering chemistry by K.Sesha Maheswaramma mridula chugh pearson publications

****

I Year - I Semester	Code: 20ES1T05	L	T	P	C
		3	0	0	3

PROGRAMMING FOR PROBLEM SOLVING 1

Course Objectives:
The objectives of Programming for Problem Solving Using C are
● To learn about the computer systems, computing environments, developing of a computer program and Structure of a C Program
● To gain knowledge of the operators, selection, control statements and repetition in C

● To learn about the design concepts of arrays, strings, enumerated structure and union types. To learn about their usage.
● To assimilate about pointers, dynamic memory allocation and know the significance of Preprocessor.
● To assimilate about File I/O and significance of functions

Course Outcomes:

Upon the completion of the course the student will learn
● To write algorithms and to draw flowcharts for solving problems
● To convert flowcharts/algorithms to C Programs, compile and debug programs
● To use different operators, data types and write programs that use two-way/ multi-way selection
● To select the best loop construct for a given problem
● To design and implement programs to analyze the different pointer applications
● To decompose a problem into

UNIT I

Introduction to C Programming- History, evolution of C**,** The Software Development Process, Development of C Algorithms. C Tokens, The main() Function, The printf() Function, Data Types**,** Operators, evaluation of Expression, Variable , Declarations and Initializations. Operator Precedence and Associativity. **Assignment -** Implicit Type Conversions, Explicit Type Conversions (Casts), Assignment Variations, Mathematical Library Functions, Interactive Input, Formatted Output, Format Modifiers.

UNIT II

Flow of Control:

Selection: if-else Statement, nested if, examples, Multi-way selection: switch, else-if, examples.
Repetition: Basic Loop Structures, Pretest and Posttest Loops, Counter-Controlled and Condition-Controlled Loops, The while Statement, The for Statement, Nested Loops, The do-while Statement.

UNIT III

Arrays & Pointers:

Arrays: One-Dimensional Arrays, Input and Output of Array Values, Array Initialization, Arrays as Function Arguments, Two-Dimensional Arrays, Matrices, Larger Dimensional Arrays, basics of searching and sorting. **Pointers**: Concept of a Pointer, Initialization of pointer variables, pointers as function arguments, passing by address, Dangling memory, address arithmetic, character pointers and functions, pointers to pointers, Dynamic memory management functions, command line arguments

UNIT IV

Functions: Parameter Declarations, Returning a Value, Functions with Empty Parameter Lists, Variable Scope, Variable Storage Class, Local Variable Storage Classes, Global Variable Storage Classes, Pass by Reference, Passing Addresses to a Function, Storing Addresses, Using Addresses, Declaring and Using Pointers, Passing Addresses to a Function. Case Study: Swapping Values, Recursion - Mathematical Recursion, Recursion versus Iteration. **Strings:** String Fundamentals, String Input and Output, String Processing, Library Functions

UNIT V

Structures: Derived types, Structures declaration, Initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, **Pointers to structures, self referential structures. Unions**, declaration, Initialization of Unions, difference between Structures and Unions-memory allocations, type def, bit-fields. **Data Files**: Declaring, Opening, and Closing File Streams, Reading from and Writing to Text Files, Random File Access

Text Books:

1) Programming for Problem Solving, Behrouz A. Forouzan, Richard F.Gilberg, CENGAGE.
2) The C Programming Language, Brian W.Kernighan, Dennis M. Ritchie, 2e, Pearson.

Reference Books:

1) Computer Fundamentals and Programming, Sumithabha Das, McGraw Hill.
2) Programming in C, Ashok N. Kamthane, AmitKamthane, Pearson.
3) Computer Fundamentals and Programming in C, PradipDey, ManasGhosh, OXFORD.

I Year – I SEMESTER

L	T	P	C
4	1	0	3

COMPUTER ENGINEERING WORKSHOP LAB

Course Code: 20ES1L07

Common for CSE &CAD

Theory
Unit 1

Identification of peripherals of a PC, Laptop, Server and Smart Phones: Prepare a report containing the block diagram along with the configuration of each component and its functionality, Input/ Output devices, I/O ports and interfaces, main memory, cache memory and secondary storage technologies, digital storage basics, networking components and speeds.

Unit 2

Linux Operating System commands:

- General command syntax
- Basic *help* commands
- Basic File system commands
- Date and Time
- Basic Filters and Text processing
- Basic File compression commands
- Miscellaneous: apt-get, vi editor

Unit 3

Networking and internet:

Networking Commands:

- ping, ssh, ifconfig, scp, netstat, ipstat, nslookup, traceroute, telnet, host, ftp, arp, wget,route

Unit 4

Audio players, recording using Mic, editing, podcast preparation

- video players, recording using webcam/camcorder, editing
- podcast, screencast, vodcast, webcasting

Practicals

1. Installation of Operating System Windows 10
2. Installation of Ubuntu 18.0.4 LTS
3. Dual OS installation Ubuntu on Windows 10
4. Installation of Adv Server Windows
5. Demonstration and practice on Microsoft Word, Power Point presentation, scanning, word to pdf. Creating email id
6. Demonstration and practice on Microsoft Excel.
7. Demonstration and practice on LaTeX and produce professional pdf documents.
8. Preparing paper, textbook certificate template using Latex
9. Cloud based productivity enhancement and collaboration tools: AWS
   - Store, sync, and share files with ease in the cloud using Google Drive
   - Document creation and editing text documents in your web browser using Google docs
   - Handle task lists, create project plans, analyze data with charts and filters using Google Sheets
   - Create pitch decks, project presentations, training modules using Google Slides
   - Manage event registrations, create quizzes, analyze responses using Google Forms o Build public sites, internal project hubs using Google Sites
   - Online collaboration through cross-platform support using Jamboard
   - Keep track of important events, sharing one's schedule, and create multiple calendars using Google Calenda
10. Virtual Machine setup:
    - Setting up and configuring a new Virtual Machine
    - Setting up and configuring an existing Virtual Machine
    - Exporting and packaging an existing Virtual Machine into a portable format

11. Network connectivity, RJ 45 clipping and color code.
12. Hardware repair Fault detection,
13. Formatting, software installation
14. VM Ware, antivirus installation updation and system maintenance.
15. Creating and maintaining Git hub account, Google form creation

ADDITIONAL EXPERIMENTS

1. Creating Blogs
2. Becoming Youtuber

Text Books:

1) Computer Fundamentals, Anita Goel, Pearson Education, 2017
2) PC Hardware Trouble Shooting Made Easy, TMH

References Books:

1) Essential Computer and IT Fundamentals for Engineering and Science Students,

Dr.N.B.Vekateswarlu, S.Chand

e-Resources:

1) https://explorersposts.grc.nasa.gov/post631/2006-2007/computer_basics/ComputerPorts.doc
2) https://explorersposts.grc.nasa.gov/post631/2006-2007/bitsnbyte/Digital_Storage_Basics.doc
3) https://www.thegeekstuff.com/2009/07/linux-ls-command-examples

BASIC ELECTRICAL AND ELECTRONICS ENGINEERING LAB
(20ES1L01)

Section A: Electrical Engineering:

The following experiments are required to be conducted as compulsory experiments:

1. Verification of Kirchhoff's laws
2. Swinburne's test on D.C. Shunt machine (Predetermination of efficiency of a given D.C.Shunt machine working as motor and generator).
3. OC and SC tests on single phase transformer (Predetermination of efficiency for different loads).
4. Brake test on 3-phase Induction motor (Determination of performance characteristics)
5. Speed control of D.C. Shunt motor by a) Armature Voltage control b) Field flux control method
6. Brake test on D.C. Shunt Motor.

Section B: Electronics Engineering.

The following experiments are required to be conducted as compulsory experiments:

1. Identification of Resistor values using color coding
2. PN junction diode characteristics a) Forward bias b) Reverse bias (Cut in voltage and resistance calculations)
3. Half wave rectifier with and without filters.
4. Full wave rectifier with and without filters.
5. CE amplifiers.
6. OP- Amp applications (Inverting, Non inverting)

APPLIED CHEMISTRY LAB

Course code: 20BS1L03 (CSE&CAD)

L	T	P	C
-	-	3	1.5

COURSE OBJECTIVE: The purpose of the titration is the detection of the equivalence point at which chemically equivalent amount of the reactants have been mixed.

Outcomes: The students entering into the professional course have practically very little exposure to lab classes. The experiments introduce volumetric analysis; redox titrations with different indicators. EDTA titrations; then they are exposed to a few instrumental methods of chemical analysis. Thus at the end of lab course, the student is exposed to different methods of chemical analysis and use of some commonly employed instruments. They thus acquire some experimental skills.

Experiments

1. Introduction to chemistry laboratory-Molarity,Normality,primary,secondary standard solutions, Volumatric titrations,Qualitative analysis,Quantitative analysis etc.
2. Determination of alkalinity of a sample containing Na_2CO_3 and $NaHCO_3$.
3. Determination of $KMnO_4$ using standard oxalic acid solution.
4. Determination of Ferrous iron using standard $K_2Cr_2O_7$ solution.
5. Determination of Copper Using standard $K_2Cr_2O_7$ solution.
6. Determination of temporary and permanent hardness of water using standard EDTA solution.
7. Determination of copper using standard EDTA solution.
8. Determination of pH of the given sample solution using pH meter.
9. Conductomeric titration between strong acid and strong base.
10. Conductomeric titration between strong acid and weak base.
11. Determination of zinc using standard EDTA solution.
12. Potentiometric titration between strong acid and strong base.
13. To determine available chlorine in bleaching powder.
14. Preparing a Printed Circuit Board using Ferric Chloride Solution.

I Year - I Semester	Code: 20ES1L08	L	T	P	C
		0	0	3	1.5

PROGRAMMING FOR PROBLEM SOLVING 1 LAB

Exercise - 1

a) Demonstrate basic Linux command to write C program , compile and Run
b) Write a program to print your details
c) Write a program for addition, subtraction, multiplication and division

Exercise - 2

a) Write a C Program to convert Celsius to Fahrenheit and vice versa
b) Calculate simple and compound Interest

Exercise - 3 Control Flow - I

a) Write a C Program to Find Whether the Given Year is a Leap Year or not.
b) Write a C Program to decide a number is even or odd

Exercise – 4 Control Flow - II

a) Write a C Program to Find Whether the Given Number is Prime Number Armstrong Number
b) Write a C program to print Floyd Triangle
c) Write a C Program to print Pascal Triangle

Exercise – 5 Control Flow - III

a) Write a C Program to make a simple Calculator to Add, Subtract, Multiply or Divide Using switch…case
b) Write a C Program to grade a student based on his percentage.

Exercise – 6 Functions

a) Write a C Program demonstrating of parameter passing in Functions and returning values.
b) Write a C Program illustrating Fibonacci, Factorial with Recursion without Recursion
c) Write a C Program FOR Towers of Hanoi

Exercise – 7 Arrays

Demonstration of arrays
a) Search-Linear.
b) Sorting-Bubble, Selection.
c) Operations on Matrix.

Exercise – 8 Strings

a) Implementation of string manipulation operations **with** library function.
i) Copy ,
ii) concatenate
iii) length
iv) compare

b) Implementation of string manipulation operations **without** library function.
i) copy
ii) concatenate
iii) length
iv) compare

Exercise – 9 Arrays and Pointers

a) Write a C Program to Access Elements of an Array Using Pointer
b) Write a C Program to find the sum of numbers with arrays and pointers

Exercises - 10 Structures

a) Write a C Program to Store Information of a Book Using Structure
b) Write a C Program to Store Information Using Structures with Dynamically Memory Allocation
c) Write a C Program to Add Two Complex Numbers by Passing Structure to a Function

Exercise – 11 Unions

a) Write a C Program to print the student details by using Unions

b) Write a C Program to demonstrate bit fields

Exercise – 12 Dynamic Memory Allocations

a) Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using malloc () function.

b) Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc () function. Understand the difference between the above two programs

Exercise -13 Files

a) Write a C programming code to open a file and to print it contents on screen.

b) Write a C program to copy files

Exercise - 14 Files Continued

a) Write a C program merges two files and stores their contents in another file.

b) Write a C program to delete a file.

BONAM VENKATA CHALAMAYYA ENGINERING COLLEGE
ODALAREVU – 533 210, Andhra Pradesh, India

I Year II Semester

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BR20 Syllabus for AIML, BVCEC w.e.f 2020-21

DETAILED SYLLABUS FOR
UG: B.Tech - ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

I Year – II SEMESTER

S. No	Course Code	Subjects	L	T	P	Credits
1	20HS2T01	English	3	0	0	3
2	20BS2T02	Integral Transforms and Calculus	3	0	0	3
3	20ES2T04	Engineering Drawing	2	0	2	3
4	20BS2T05	Applied Physics	3	0	0	3
5	20ES2T 06	Programming for Problem Solving 2	3	0	0	3
6	20BS2L01	Applied Physics Lab	0	0	3	1.5
7	20ES2L09	Programming for Problem Solving 2 Lab	0	0	3	1.5
8	20HS2L01	English Communication Skills Lab	0	0	3	1.5
	20HS2M02	Indian Constitution	2	0	0	0
Total Credits			**17**	**1**	**8**	**19.5**

ENGLISH – I (20HS2T01)
(Common to all branches)

L	T	P	C
2	-	-	2

Introduction

In view of the growing importance of English as a tool for global communication and the consequent emphasis on training the students to acquire communicative competence, the syllabus has been designed to develop linguistic and communicative competence of the students of Engineering.

As far as the detailed textbooks are concerned, the focus should be on the skills of listening, speaking, reading and writing. The non-detailed textbook is meant for extensive reading for pleasure and profit. Thus the stress in the syllabus is primarily on the development of communicative skills and fostering of ideas.

Course Objectives:

COB 1: To improve the language proficiency of the students in English with emphasis on LSRW skills.
COB 2: To enable the students to study and comprehend the prescribed lessons and subjects more effectively relating to their theoretical and practical components.
COB 3: To develop the communication skills of the students in both formal and informal situations.
COB 4: To develop the ways to overcome fear and use of words for irony.
COB 5: To make the learners understand the development conditions and the core competences of the State of prioritize education system.
COB 6: To discuss that water is the world's most precious natural resources.
COB 7: To discuss how human sensitivity changes in accordance to times and situations in life.
COB 8: To inform the learner that all men can come together to abolish the war.

LISTENING SKILLS:

Objectives:

1. To enable the students to appreciate the role of listening skill and improve their pronunciation.
2. To enable the students to comprehend the speech of people belonging to different backgrounds and regions.
3. To enable the students to listen for general content, to fill up information and for specific information.

SPEAKING SKILLS:

Objectives:

1. To make the students aware of the importance of speaking for their personal and professional communication.
2. To enable the students to express themselves fluently and accurately in social and professional success.
3. To help the students describe objects, situations and people.
4. To make the students participate in group activities like role-plays, discussions and debates.
5. To make the students participate in just a minute talks.

READING SKILLS:

Objectives:

1. To enable the students to comprehend a text through silent reading.
2. To enable the students to guess the meanings of words, messages and inferences of texts in given contexts.
3. To enable the students to skim and scan a text.
4. To enable the students to identify the topic sentence.
5. To enable the students to identify discourse features.
6. To enable the students to make intensive and extensive reading.

WRITING SKILLS:

Objectives:

1. To make the students understand that writing is an exact formal skills.
2. To enable the students to write sentences, paragraphs, e-mails and essays.
3. To make the students identify and use appropriate vocabulary.
4. To enable the students to narrate and describe.
5. To enable the students to write coherently and cohesively.

Course Outcomes:

At the end of the Course, Student will be able to:

CO 1: Identify the ways to overcome fear and use of words for irony.
CO 2: Interpret the development conditions and the core competences of the state to prioritize education system.
CO 3: Analyze water as the world's most precious natural resources.
CO 4: Illustrate human sensitivity to the changing times and situations in life.
CO 5: Predict that war causes destruction and calling for peace establishment.
CO 6: Illustrate biographies for motivating readers and strengthening communicative grammar skills

Methodology:

1. The class is to be learner-centred where the learners are to read the texts to get a comprehensive idea of those texts on their own with the help of the peer group and the teacher.
2. Integrated skill development methodology has to be adopted with focus on individual language skills as per the tasks/exercise.
3. The tasks/exercises at the end of each unit should be completed by the learners only and the teacher intervention is permitted as per the complexity of the task/exercise.
4. The teacher is expected to use supplementary material wherever necessary and also generate activities/tasks as per the requirement.
5. The teacher is permitted to use lecture method when a completely new concept is introduced in the class.

Recommended Topics:

UNIT-I:
1. AN ASTROLOGERS'S DAY - R.K. NARAYAN (Detailed)
2. G.D. NAIDU (Non-Detailed)
UNIT-II:
1. BUILDING A NEW STATE - A. P. J. ABDUL KALAM (Detailed)
2. G.R. GOPINATH (Non-Detailed)
UNIT-III:
1. WATER: THE ELIXER OF LIFE- C. V. RAMAN (Detailed)
2. SUDHA MURTHY (Non-Detailed)
UNIT-IV:
1. THE WOODROSE-ABBURI CHAYA DEVI (Detailed)
2. JAGADIS CHANDRA BOSE (Non-Detailed)
UNIT-V:
1. PROGRESS- ST. JOHN ERVINE (Detailed)
2. HOMI JEHANGIR BHABHA(Non-Detailed)

Textbooks:
Detailed Text Book: **USING ENGLISH** by Orient Black Swan.
Non-Detailed Text Book: **Trailblazers** by Orient black Swan.

INTEGRAL TRANSFORMS AND CALCULAS
(Common to all Branches)

Subject code: 20BS2T02

L	T	P	C
3	0	0	3

Course Objectives:

COB 1: To equip the students with the necessary mathematical skills and techniques those are Essential for an engineering course.

COB 2: To help the students acquire a necessary base of Multiple Integrals.

COB 3: Making the student capable of creating vector differentiation and integration for real life Situation.

Course Outcomes:

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

CO 1: Apply the Laplace transform for solving differential equations.(L3)
CO 2: Determine Fourier series, Fourier Transforms of the given functions.(L3)
CO 3: Compute the Z- transforms for certain functions and apply the properties of Z- transforms to solve difference equations.()
CO 4: Apply multiple integrals to find area, volume center of mass, center of gravity of given Region/object. (L3)
CO 5: Assess the information with their identities regarding physical perception by Gradient, divergence and curl. (L5)
CO 6: Evaluate integrals of function or vector related quantities over curves, surfaces, and domains in two- and three – dimensional space.(L5)

UNIT I: - Laplace transforms

Laplace transforms of standard functions-Shifting theorems - Transforms of derivatives and integrals – Unit step function –Dirac's delta function- Inverse Laplace transforms– Convolution theorem (with out proof). ROC of Laplace transforms

Applications: Solving ordinary differential equations (initial value problems) using Laplace transforms.

Unit II:- Fourier series and Fourier Transforms

Fourier Series: Introduction – Periodic functions – Fourier series of periodic function – Dirichlet's conditions – Even and odd functions – Change of interval – Half-range sine and cosine series.

Fourier Transforms: Fourier integral theorem (without proof) – Fourier sine and cosine integrals – Fourier Sine and cosine transforms – Properties – inverse transforms – Finite Fourier transforms. Convolution Theorem (with out proof).

UNIT III:- Z-transforms

Z-transform – properties – Damping rule – Shifting rule – Initial and final value theorems -Inverse z- transform - Convolution theorem(with out proof) – Solution of difference equation by z-transforms. ROC of Z-transforms

UNIT IV: Multiple Integrals:

Evaluation of Double Integrals (Cartesian and polar coordinates)-change of order of integration (only Cartesian form)-Evaluation of Triple Integrals-Change of variables (Cartesian to polar) for double integrals only.

Applications: Finding Areas and Volumes.

Unit V:- Vector Differentiation& Vector Integration

Line, Surface and volume integrals, Vector integral theorems: Green's-Gauss divergence-Stokes Theorem (statements only) and related problems.

Gradient- Divergence- Curl - Laplacian and second order operators -Vector identities.

Applications: Potential surfaces

Vector Integration

Line, Surface and volume integrals, Vector integral theorems: Green's-Gauss divergence-Stokes Theorem (statements only) and related problems.

Applications: work done by force.

Text Books:

1. **B. S. Grewal**, Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **Dr T K V Iyengar**, **Dr B.Krishna Gandhi**, **S.Ranhanatham** and Dr. **M.V.S.S.N Prasad**, Engineering Mathematics, S.Chand & Company Ltd.
3. **B. V. Ramana**, Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.

Reference Books:

1. **Erwin Kreyszig**, Advanced Engineering Mathematics, 10th Edition,Wiley-India.
2. **Dean. G. Duffy**, Advanced Engineering Mathematics with MATLAB, 3rd Edition, CRC Press.

3. **Peter O' Neil**, Advanced Engineering Mathematics, Cengage.
4. **N.P.Bali & Manish Goyal**, A Text book of Engineering Mathematics, Lakshmi Publications.

L	T	P	C
1	0	4	3

ENGINEERING DRAWING
(Common to EEE, ECE, CSE & CAD)

Course Objectives:

COB1: To provide basic concepts in engineering drawing
COB2: To impart knowledge about standard principles of orthographic projections of objects
COB3: To visualize and represent the pictorial views with proper dimensioning and scaling

Course Outcomes:

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

CO1: Construct polygons and engineering curves
CO2: Draw orthographic projections of points and lines inclined to one reference plane
CO3: Draw projections lines inclined to both the reference planes and traces
CO4: Draw projections of planes

CO5: Draw projections of solids
CO6: Convert the isometric views to orthographic views and vice-versa

UNIT I

Objective: To introduce the students to use drawing instruments and to draw polygons, Engg. curves and scales.

Lines, Lettering and Dimensioning: Types of lines, Lettering, Dimensioning

Polygons: Constructing regular polygons by general methods, inscribing and describing polygons on circles.

Curves: Parabola, Ellipse and Hyperbola by general methods, cycloids, involutes, tangents & normals for the curves.

UNIT II

Objective: To introduce orthographic projections, projections of points & Line inclined both the planes

Orthographic Projections: Horizontal plane, vertical plane, profile plane, importance of reference lines, projections of points in various quadrants, projections of line inclined to one plane and both the planes, determination of true lengths, angles of inclination

,

UNIT-III

Objective: The objective is to make the students draw the projections of the various types of Planes in different simple positions and inclined to one plane and both the planes.

Projections of planes: Regular planes perpendicular / parallel to one plane and inclined to the other reference plane; inclined to both the reference planes.

UNIT IV

Objective: The objective is to make the students draw the projections of the various types of solids in different simple positions and inclined to one plane.

Projections of Solids – Prisms, Pyramids, Cones and Cylinders with the axis inclined to both the Planes.

UNIT V

Objective: The objective is to represent the object in 3D view through isometric views and convert the isometric view to orthographic view and vice versa.

Principles of Isometric Projections - Isometric Scale – Isometric Views – Conventions – Isometric Views of Lines, Plane Figures, Simple Solids, Conversion of Isometric Views to Orthographic Views and Vice-versa

TEXT BOOKS:

1. Engineering Drawing by N.D. Butt, Charotar Publications

2. Engineering Drawing by Agarwal & Agarwal, Tata McGraw Hill Publishers

REFERENCE BOOKS:

1. Engineering Drawing, K. Venugopal, V. Prabhu Raja and G. Sreekanjana, New Age International Publications
2. Engineering Drawing, K.L.Narayana & P. Kannaiah, SciTech Publishers.
3. Engineering Drawing, Dhananjay V. Jolhe, Tata McGraw Hill Publishers

4. Engineering Graphics for Degree, K.C. John, PHI Publishers.

APPLIED PHYSICS

(common to ECE,EEE, CSE CAD)

Course Code : 20BS2T05 **L T P C**

3 0 0 3

Objective

The objective of this course is to develop scientific temper and analytical capability through learning physical concepts and their applications in science and technology.

Course Outcomes:

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

CO 1- Understand the principles of interference and diffraction to design and enhance the resolving Power of various optical instruments.(BL2)

CO 2 - Understand the basics of modern technologies in lasers and optical fibers and their applications in Various fields.(BL2)

CO 3 - Identify the magnetic and dielectric materials from their basic behavior and Learn their applications.(BL3)

CO 4 - Summarise the basic concepts of Quantum Theory.(BL2)

CO 5 - Interpret the effects of temperature on Fermi Dirac distribution function and classification of Solids based on band theory (BL5)

CO 6 - Analyze the principles of carrier concentration in semiconductors.(BL4)

UNIT I Wave Optics 8 hrs

Interference: Introduction- Principle of superposition –Interference of light -Interference in thin films (Reflection Geometry) -Newton's Rings-Determination of wavelength and refractive index.

Diffraction: Introduction -Fresnel and Fraunhofer diffraction -Fraunhofer diffraction due to single slit, double slit -N-slits (Qualitative) –Rayleigh criterion for resolving power-Diffraction Grating -Resolving power of Grating(Qualitative)-Introduction to polarization.

Unit II Lasers & Fiber Optics 10 hrs

Lasers: Introduction –Characteristics of laser –Spontaneous and Stimulated emissions of radiation –Einstein's coefficients –Population inversion –Lasing action -Pumping Schemes –Ruby laser –He-Ne laser -Applications of lasers.

Fiber optics: Introduction –Principle of optical fiber-Acceptance Angle -Numerical Aperture -Classification of optical fibers based on refractive index profile and modes –Applications

Unit III Dielectric & Magnetic Materials 8 hrs

Dielectric Materials: Introduction -Dielectric polarization -Dielectric polarizability, Susceptibility and Dielectric constant -Types of polarizations-Electronic (Quantitative), Ionic (Quantitative) and Orientation polarizations (Qualitative) -Lorentz internal field-Clausius-Mossotti equation-Piezoelectricity.

Magnetic Materials :Introduction -Magnetic dipole moment -Magnetization-Magnetic susceptibility and permeability -Origin of permanent magnetic moment –Classification of magnetic materials: Dia, para, Ferro, antiferro & Ferrimagnetic materials -Domain concept for Ferromagnetism & Domain walls (Qualitative) -Hysteresis -soft and hard magnetic materials-Eddy currents-Engineering applications.

Unit IV Quantum Mechanics& Free Electron Theory 5 hrs

Quantum Mechanics: Dual nature of matter –Heisenberg's Uncertainty Principle –Physical Significance and properties of wave function –Schrodinger's time independent and dependent wave equations–Particle in a one-dimensional infinite potential well

Free Electron Theory: Classical free electron theory (Qualitative with discussion of merits and demerits) –Quantum free electron theory–Equation for electrical conductivity based on quantum free electron theory-Fermi-Dirac distribution-Density of states (3D) -Fermi energy.

Band theory of Solids: Bloch's Theorem (Qualitative) -Kronig -Penney model (Qualitative)-E vs K diagram -effective mass of electron –Classification of crystalline solids–concept of hole.

UNIT V: Band theory of Solids& Semiconductor Physics **9 hrs**

Band theory of Solids: Bloch's Theorem (Qualitative) -Kronig -Penney model (Qualitative)-E vs K diagram -effective mass of electron –Classification of crystalline solids–concept of hole.

Semiconductors: Introduction-Intrinsic semiconductors –Density of charge carriers –Electrical conductivity –Fermi level –extrinsic semiconductors –density of charge carriers –dependence of Fermi energy on carrier concentration and temperature -Drift and diffusion currents –Einstein's equation-Hall effect –Hall coefficient –Applications of Hall effect.

Textbooks

1. **Pillai, S. O**., Solid State Physics,5th edition, New Age International (P) Ltd., New Delhi,2004.
2. Solid State Physics by **A.J. Dekker** (Mc Millan India Ltd).
3. Principles of electronics by **V.K Mehta** S .Chand Publishers.
4. Applied Physics' by **MN Avadhanulu** & **TVS Arun Murthy** (S.CHAND).
5. **RK Gaur** & **SL Gupta**, Engineering physics , Dhanapati rai publications.
6. **Applied Physics by palanasamy (Scitech publications)**

References

1. Engineering Physics' by **Sanjay D Jain** and **Girish G Sahasrabudhe** (University Press),
2. Engineering Physics' by **Mani Naidu S** (Pearson Publications)).
3. Applied Physics' by **T. Bhimasankaram** (BSP BH Publications)).
4. Physics' by **Palanisamy** (Scitech Publishers)

5. Engineering Physics' by **B.K.Pandey** & **S. Chaturvedi** (Cengage Learning)
6. Introduction to solid state physics' by **Charles Kittle** (Willey India Pvt.Ltd)

I Year - II Semester	Code: 20ES2T06	L	T	P	C
		3	0	0	3

PROGRAMMING FOR PROBLEM SOLVING 2

Course Objectives:

The objective of the course is to

- Introduce the fundamental concept of data structures and abstract data types
- Emphasize the importance of data structures in developing and implementing efficient algorithms
- Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms

Course Outcomes:

After completing this course a student will be able to:

- Summarize the properties, interfaces, and behaviors of basic abstract data types
- Discuss the computational efficiency of the principal algorithms for sorting & searching
- Use arrays, records, linked structures, stacks, queues, trees, and Graphs in writing programs
- Demonstrate different methods for traversing trees

UNIT I

Data structure-Definition, types of data structures

Recursion: Definition, Design Methodology and Implementation of recursive algorithms, Linear and binary recursion, recursive algorithms for factorial function, GAD computation, Fibonacci sequence, Towers of Hanoi, Tail recursion List Searches using Linear Search, Binary Search, Fibonacci Search

Sorting Techniques: Basic concepts, Sorting by:

Insertion sort, selection-heap sort, exchange-bubble sort, quick sort, distribution-radix sort and merging-merge sort Algorithms

UNIT II

Stacks and Queues: Basic Stack Operations, Representation of a Stack using Arrays, Stack Applications: Reversing list, Factorial Calculation, Infix to postfix Transformation, Evaluating

Arithmetic Expressions.

Queues: Basic Queues Operations, Representation of a Queue using array, Implementation of Queue Operations using Stack, Queues.

Applications of Queues-Round robin Algorithm, Circular Queues, Priority Queues

UNIT III

Linked Lists: Introduction, single linked list, representation of a linked list in memory, Operations on a single linked list, Reversing a single linked list, applications of single linked list to represent polynomial expressions and sparse matrix manipulation, Advantages and disadvantages of single linked list, Double linked list, Circular linked list.

UNIT IV

Trees: Basic tree concepts, Binary Trees: Properties, Representation of Binary Trees using arrays and linked lists, operations on a Binary tree, Binary Tree Traversals (recursive), Creation of binary tree from in-order , preorder and post order traversals

Advanced concepts of Trees: Tree Traversal using stack (**non recursive**), Threaded Binary Trees. Binary search tree, Basic concepts, BST operations: insertion, deletion, balanced binary trees–need, basics and applications in computer science (No operations).

UNIT V

Graphs: Basic concepts, Representations of Graphs: using Linked list and adjacency matrix, Graph algorithms Graph Traversals (**BFS & DFS**), applications: Dijkstra's shortest path, Transitive closure, Minimum Spanning Tree using Prim's Algorithm, Warshall's Algorithm (**Algorithmic Concepts Only, No Programs required**).

Text Books:

1.Data Structure with C, Seymour Lipschutz, TMH
2.DataStructures,2/e, Richard F, Gilberg, Forouzan, Cengage

Reference Books:

1. C & Data structures 5th Edn. by Balagurusamy
2.Data Structures using C, Reema Thareja, Oxford
3.Data Structures and Algorithm Analysis in C,2nded, Mark Allenweiss

APPLIED PHYSICS LAB

(Common to ECE,EEE, CSE,CAD)

Course Code: 20BS1L01(ECE,EEE)	L	T	P	C
20BS2L01(CSE,CAD)	0	0	3	1.5

Objective: To train the Engineering students to handle instruments and their design methods to improve the accuracy of measurements.

Outcome: Physics lab curriculum gives fundamental understanding of design of an instrument with targeted accuracy for physical measurements.

APPLIED PHYSICS LAB

(Any 10 of the following listed 15 experiments)

- 8 experiments to be performed in regular mode
- 2 experiments to be performed in online mode (virtual lab)

List of Experiments:

1. Determination of wavelength of a source-Diffraction Grating-Normal incidence
2. Newton's rings –Radius of Curvature of Plano convex Lens.
3. Determination of Acceleration due to Gravity and Radius of Gyration- Compound Pendulum
4. Energy Band gap of a Semiconductor p - n junction diode
5. Study of LED I/V Characteristics
6. I/V characteristics of Photodiode
7. Determination of Planck's constant using photocell
8. Determine the rigidity modulus of a material – Torsional pendulum
9. Thermistor characteristics – Temperature Coefficient
10. Magnetic field along the axis of a current carrying coil – Stewart and Gee's apparatus
11. Study the variation of B versus H by magnetizing the magnetic material(B-H curve)
12. Verification of the laws of vibrations in stretched stings - Sonometer.
13. Determine the time constant for R-C circuit.

14. Determination of Hall voltage and Hall coefficients of a given semiconductor using Hall effect
15. Study of I/V Characteristics of Semiconductor p - n junction diode.

REFERENCES:

1. Engineering Physics Lab Manual by Dr.Y. Aparna & Dr.K.Venkateswarao (V.G.S.Book links)
2. Physics practical manual, Lorven Publications.

Web link:

URL: www.vlab.amrita.edu/index.php

I Year - II Semester	Code: 20ES2L09	L	T	P	C
		0	0	3	1.5

PROGRAMMING FOR PROBLEM SOLVING 2 LAB

Exercise1:

Write recursive program which computes the n^{th} Fibonacci number, for appropriate values of n. Analyze behavior of the program obtain the frequency count of the statement for various values of n.

Exercise2:

Write recursive program for the following

a) Write recursive and non recursive C program for calculation of Factorial of an integer
b) Write recursive and non recursive C program for calculation of GAD(n, m)
c) Write recursive and non recursive C program for Towers of Hanoi: N disks are to be transferred from peg S to peg D with Peg I as the intermediate peg.

Exercise3:

a) Write C program that use both recursive and non recursive functions to perform Linear search for a Key value in a given list.
b) Write C program that use both recursive and non recursive functions to perform Binary search for a Key value in a given list.
c) Write C program that use both recursive and non recursive functions to perform Fibonacci search for a Key value in a given list.

Exercise4:

a) Write C program that implement Bubble sort, to sort a given list of integers in ascending order
b) Write C program that implement Quick sort, to sort a given list of integers in ascending order
c) Write C program that implement Insertion sort, to sort a given list of integers in ascending order

Exercise5:

a) Write C program that implement heap sort ,to sort a given list of integers in ascending order
b) Write C program that implement radix sort, to sort a given list of integers in ascending order
c) Write C program that implement merge sort, to sort a given list of integers in ascending order

Exercise6:

a) Write C program that implement stack(its operations)using arrays
b) Write C program that implement stack(its operations)using Linked list

Exercise7:

a) Write a C program that uses Stack operations to convert infix expression into postfix expression
a) Write C program that implement Queue (its operations) using arrays.
b) Write C program that implement Queue(its operations)using linked lists

Exercise8:

a) Write a C program that uses functions to create a singly linked list
b) Write a C program that uses functions to perform insertion operation on a singly linked list
c) Write a C program that uses functions to perform deletion operation on a singly linked list

Exercise9:

a) Adding two large integers which are represented in linked list fashion.
b) Write a C program to reverse elements of a single linked list.
c) Write a C program to store a polynomial expression in memory using linked list
d) Write a C program to representation the given sparse matrix using arrays.
e) Write a C program to representation the given sparse matrix using linked list

Exercise10:

a) Write a C program to Create a Binary Tree of integers
b) Write a recursive C program for traversing a binary tree in preorder, in order and post order.
c) Write a non recursive C program for traversing a binary tree in preorder, in order and post order.
d) Program to check balance property of a tree.

Exercise11:

a) Write a C program to create a BST
b) Write a C program to insert an ode in to a BST.
c) Write a C program to delete an ode from a BST.

ENGLISH LANGUAGE AND COMMUNICATION SKILLS LAB
(Common to all branches)

Course Code 20HS1L01 (EEE & ECE)
20HS2LO1 (CIV, MECH, CSE &CAD, AIML)

L	T	P	C
-	-	3	1.5

OUT COMES

Co.1. Prepare the learners to improve speech production and confidence levels avoiding stage fright.
Co.2. Build the ability of qualifying IELTS, TOFEL, GRE and the other competitive exams.
Co.3. Develop the technical and professional skills in writing emerged to corporate needs.
Co.4. Apply to use the language devices and connectors while writing and speaking.
Co.5. Generate the interest among the learners to read all texts intensively and extensively.
Co.6. Identify the different speeches in life through active and focused listening.

SYLLABUS

Unit I: Communication Skills

- **a.** Body Language
- **b.** JAM
- **c.** Introduce yourself
- **d.** Role-play

Unit. II: Vocabulary Development

a. Synonyms and Antonyms
b. Cloze Test
c. Interchange of Words
d. Sentence Matching
e. One Word Substitutions

f. Commonly Confused Words
g. Verbal Analogy
h. Sentence Fillers
i. Idioms & Phrases

Unit III: Writing Strategies

a. Paragraph Writing
b. Report Writing
b. Essay Writing and Picture composition
c. Letter Writing: Business Letters,
d. Job Application Letters: Opening, Body and Closing
e. Resume Design and Parts of a Resume
f. E-mail Writing

Unit IV: Communicative Grammar

a. Error Identification/ Error Location
b. Sentence Improvement
c. Identifying Correct Alternative
d. Incomplete Sentence
e. Para Jumbles
f. Phrase/Connectors (Starters)

Unit V: Reading Comprehension

a. Reading Process
b. Reading Strategies
c. Tips to Answering Questions

Unit VI: Listening Skills

a. Listening to audio lectures/stories/biographies
b. Listening to announcements/Radio

Lab Software: Globarena Technologies Ltd.,

Prescribed Text Books:

1.**Communicative English for Engineers and Professionals** by Nitin Bhatnagar & Mamata Bhatnagar, Pearson Publications, 2010, New Delhi

2. **English for Technical Communication for Engineering students**, Aysha Vishwamohan, TataMc Graw Hill, 2009

3. **Effective Technical Communication** by M.A. Sharif Rizvi, Tata MacGrawhill, 2005, New Delhi

4. **Soft Skills – Key to Success in workplace and life** by Meenakshi Raman & Shalini Upadhayay; Cengage Publications

Reference Books:

1. Business Communication by Meenakshi Raman (II edition), Oxford Publications.

2. Technical Communication Principles and Practice. Raman, Meenakshi and Sharma, Sangeeta Third Edition. New Delhi: Oxford University Press. 2015. Print.

3. Technical Communication by Meenakshi Raman & Sangeetha Sarma, OU Press, 3 rd edition, 2014

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DETAILED SYLLABUS FOR – BR20
UG: B.Tech - ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
II Year – I SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20BS3T03	Probability and Statistics	3	0	0	3
2	20HS3T02	Managerial Economics and Financial Analysis	3	0	0	3
3	20ML3T01	Computer Organization	3	0	0	3
4	20ES3T04	Python Programming	3	0	0	3
5	20CS3T05	Digital Logic Design	3	0	0	3
6	20ML3L01	UNIX and Shell Programming Lab	3	0	0	1.5
7	20ES3L07	Python Programming Lab	0	0	3	1.5
8	20CS3L08	R Programming Lab	0	0	3	1.5
9	20CS3S01	Skill Oriented Course - Java Script	1	0	2	2
	20BS3M03	Environmental Science	2	0	0	0
		Total	21	0	8	21.5

L T P C

3 0 0 3

PROBABILITY AND STATISTICS
(Common to all Branches)

Subject code: 20BS3T03

Course Objectives:

COB 1: Know the importance of usage of Probability and Statistics.
COB 2: To help the students acquire a necessary base of Correlation, Quality Checking.
COB 3**:** Use R for statistical programming, computation, graphics, and modeling

Course Outcomes:

At the end of the course, student will be able to:
CO 1: Use the knowledge of Probability and Statistics in various situations. (L3)
CO 2: Analyze the concepts, Random variables and Probability distributions. (L4)
CO 3: Measure Correlation between variables and obtain lines of regression. (L5)
CO 4: Design quality control charts for quality checking. (L6)
CO 5: Use R for statistical programming, computation, graphics, and modeling. (L3)
CO 6: Classify Functions and use R in an efficient way. (L4)

UNIT I:-Introduction to Probability and Statistics

Probability-Addition law of Probability-Independent Events-Conditional Probability-Bayes Theorem (with out proof).

Statistics-Measures of Central Tendency-mean, mode, median, Measures of Dispersion-Variance, standard deviation, Skewness, Kurtosis.

UNIT II:-Random variables and Distributions

Introduction- Random variables- Distribution function- Discrete distributions (Binomial and Poisson distributions only) Continuous distributions: Normal, Normal approximation to Binomial distribution, Gamma and Weibull distributions.

UNIT III: - Curve fitting and Correlation

Introduction - Fitting a straight line –Second degree curve-exponential curve - power curve by method of least squares, Simple Correlation and Regression

UNIT IV: - Statistical Quality Control Methods

Introduction - Methods for preparing control charts – Problems using $\bar{x}$, p, R charts and attribute charts

UNIT V: - Introduction of R

Basic Math, Variables, Data Types, Vectors, Arrays , Lists, Data Frames Matrices, , Classes**.** R Programming Structures, Control Statements, Loops, - Looping Over Non vector Sets,If-Else, Arithmetic and Boolean Operators and values.

Math Function Extended Example Functions Sorting, Linear Algebra Operation on Vectors and Matrices. Graphics, Creating Graphs, Histogram, the plot() Function –Customizing Graphs, Saving Graphs to Files. Regression. Create R Pie, Bar Charts using R, Create Histogram and Line graph, Calculate mean, median and mode using R Analyze Linear Regression using R.

Text Books:

1. **B.S.GREWAL,** Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **N.P.Bali**, Engineering Mathematics, Lakshmi Publications.
3. Probability and Statistics, **Dr T K V Iyengar**, **Dr B.Krishna Gandhi**, **S.Ranhanatham** and Dr. **M.V.S.S.N Prasad**, S.Chand & Company Ltd.
4. The Art of R Programming, **Norman Matloff**, Cengage Learning
5. R for Everyone, **Lander**, Pearson

Reference Books:

1. Fundamentals of Mathematical Statistics, **S.C .Gupta , V.K .Kapoor**, S.Chand & Company Ltd
2. **B.V.Ramana**, Engineering Mathematics: 4th Edition, Tata McGraw Hill, 2009, New Delhi
3. **ERWIN KREYSZIG,** Advanced Engineering Mathematics, 10th Edition, Wiley-India
4. R Cookbook, **PaulTeetor, Oreilly**.
5. R in Action, **Rob Kabacoff**, Manning

II Year – I SEMESTER

L	T	P	C
3	0	0	3

MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS

Course Code:20HS3T02

Unit – I:

Introduction to Managerial Economics and demand Analysis:
Definition of Managerial Economics and Scope-Managerial Economics and its relation with other subjects-Concepts of Demand-Types-Determents-Law of Demand its Exception-Elasticity of Demand-Types and Measurement-Demand forecasting and its Methods.

Unit – II:

Production and Cost Analyses:
Production function-Isoquants and Isocosts-Law of Variable proportions-Cobb-Douglas Production function-Economics of Sale-Cost Concepts-Opportunity Cost-Fixed vs Variable Costs-Explicit Costs vs Implicit Costs-Out of Pocket Costs vs Imputed Costs-Cost Volume Profit analysis-Determination of Break-Even Point (Simple Problem)

Unit – III:

Introduction to Markets, Theories of the Firm & Pricing Policies:
Market Structures: Perfect Competition, Monopoly and Monopolistic and Oligopoly – Features – Price, Output Determination – Managerial Theories of firm: Maris and Williamson's models – Methods of Pricing: Limit Pricing, Market Skimming Pricing, Internet Pricing: Flat Rate Pricing, Usage sensitive, Transaction based pricing, Priority Pricing.

Unit – IV:

Types of Business Organization and Business Cycles:
Features and Evaluation of Sole Trader – Partnership – Joint Stock Company – State/Public Enterprises and their forms – Business Cycles – Meaning and Features – Phases of Business Cycle.

Unit – V:

Introduction to Accounting & Financing Analysis:
Introduction to Double Entry Systems – Preparation of Financial Statements-Analysis and Interpretation of Financial Statements-Ratio Analysis – Preparation of Funds flow cash flow statements (Simple Problems)

Capital and Capital Budgeting: Capital Budgeting: Meaning of Capital-Capitalization-Meaning of Capital Budgeting-Need for Capital Budgeting-Techniques of Capital Budgeting-Traditional and Modern Methods.

TEXT BOOKS

1. Dr. N. Appa Rao, Dr. P. Vijay Kumar: 'Managerial Economics and Financial Analysis', Cengage Publications, New Delhi – 2011
2. Dr. A. R. Aryasri – Managerial Economics and Financial Analysis, TMH 2011
3. Prof. J.V.Prabhakara rao, Prof. P. Venkatarao. 'Managerial Economics and Financial Analysis', Ravindra Publication.

REFERENCES:

1. V. Maheswari: Managerial Economics, Sultan Chand.
2. Suma Damodaran: Managerial Economics, Oxford 2011.
3. Dr. B. Kuberudu and Dr. T. V. Ramana: Managerial Economics & Financial Analysis, Himalaya Publishing House 2011.

II Year - I Semester	Code: 20ML3T01	L	T	P	C
		3	0	0	3

COMPUTER ORGANIZATION

Course Objectives:

The course objectives of Computer Organization are to discuss and make student familiar with the

- Principles and the Implementation of Computer Arithmetic
- Operation of CPUs including RTL, ALU, Instruction Cycle and Busses
- Fundamentals of different Instruction Set Architectures and their relationship to the CPU Design
- Memory System and I/O Organization
- Principles of Operation of Multiprocessor Systems and Pipelining

Course Outcomes:

By the end of the course, the student will

- Develop a detailed understanding of computer systems
- Cite different number systems, binary addition and subtraction, standard, floating-point, and micro operations
- Develop a detailed understanding of architecture and functionality of central processing unit
- Exemplify in a better way the I/O and memory organization
- Illustrate concepts of parallel processing, pipelining and inter processor communication

UNIT I

Basic Structure of Computers: Basic Organization of Computers, Historical Perspective, Bus Structures, Data Representation: Data types, Complements, Fixed Point Representation. Floating, Point Representation. Other Binary Codes, Error Detection Codes. Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Division Algorithms.

UNIT II

Register Transfer Language and Micro operations: Register Transfer language. Register Transfer Bus and Memory Transfers, Arithmetic Micro operations, Logic Micro Operations, Shift Micro Operations, Arithmetic Logic Shift Unit. Basic Computer Organization and Design: Instruction Codes, Computer Register, Computer Instructions, Instruction Cycle, Memory – Reference Instructions. Input –Output and Interrupt, Complete Computer Description.

UNIT III

Central Processing Unit: General Register Organization, STACK Organization. Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer. Microprogrammed Control: Control Memory, Address Sequencing, Micro Program example, Design of Control Unit.

UNIT IV

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory. Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupts, Direct Memory Access.

UNIT V

Multi Processors: Introduction, Characteristics of Multiprocessors, Interconnection Structures, Inter Processor Arbitration. Pipeline: Parallel Processing, Pipelining, Instruction Pipeline, RISC Pipeline, Array Processor.

Text Books:

1) Computer System Architecture, M. Morris Mano, Third Edition, Pearson, 2008.
2) Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 5/e, McGraw Hill, 2002.

Reference Books:

1) Computer Organization and Architecture, William Stallings, 6/e, Pearson, 2006.
2) Structured Computer Organization, Andrew S. Tanenbaum, 4/e, Pearson, 2005.
3) Fundamentals of Computer Organization and Design, Sivarama P. Dandamudi, Springer, 2006.

II Year - I Semester	Code: 20ES3T04	L	T	P	C
		3	0	0	3

PYTHON PROGRAMMING

Course Objectives:

The Objectives of Python Programming are

- To learn about Python programming language syntax, semantics, and the runtime environment
- To be familiarized with universal computer programming concepts like data types, containers
- To be familiarized with general computer programming concepts like conditional execution, loops & functions
- To be familiarized with general coding techniques and object-oriented programming

Course Outcomes:

- Develop essential programming skills in computer programming concepts like data types, containers
- Apply the basics of programming in the Python language
- Solve coding tasks related conditional execution, loops
- Solve coding tasks related to the fundamental notions and techniques used in object-oriented programming

UNIT I

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

UNIT II

Control Statement: Definite iteration for Loop Formatting Text for output, Selection if and if else Statement Conditional Iteration The While Loop. Strings and Text Files: Accessing Character and Substring in Strings, Data Encryption, Strings and Number Systems, String Methods Text Files.

UNIT III

List and Dictionaries: Lists, Defining Simple Functions, Dictionaries, Design with Function: Functions as Abstraction Mechanisms, Problem Solving with Top Down Design, Design with Recursive Functions, Case Study Gathering Information from a File System, Managing a Program's Namespace, Higher Order Function.
Modules: Modules, Standard Modules, Packages.

UNIT IV

File Operations: Reading config files in python, Writing log files in python, Understanding read functions, read(), readline() and readlines(), Understanding write functions, write() and writelines(), Manipulating file pointer using seek, Programming using file operations. Object Oriented Programming: Concept of class, object and instances, Constructor, class attributes and destructors, Real time use of class in live projects, Inheritance , overlapping and overloading operators, Adding and retrieving dynamic attributes of classes, Programming using Oops support. Design with Classes: Objects and Classes, Data modeling Examples, Case Study An ATM, Structuring Classes with Inheritance and Polymorphism

UNIT V

Errors and Exceptions: Syntax Errors, Exceptions, Handling Exceptions, Raising Exceptions, User-defined Exceptions, Defining Clean-up Actions, Redefined Clean-up Actions. Graphical User Interfaces: The Behavior of Terminal Based Programs and GUI -Based, Programs, Coding Simple GUI-Based Programs, Other Useful GUI Resources. Programming: Introduction to Programming Concepts with Scratch.

Text Books:

1) Fundamentals of Python First Programs, Kenneth. A. Lambert, Cengage.
2) Python Programming: A Modern Approach, VamsiKurama, Pearson.

Reference Books:

1) Introduction to Python Programming, Gowrishankar.S, Veena A, CRC Press.
2) Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

II Year - I Semester	Code: 20CS3T05	L	T	P	C
		3	0	0	3

DIGITAL LOGIC DESIGN

Course Objectives:

- To study the basic philosophy underlying the various number systems, negative number representation, binary arithmetic, theory of Boolean algebra and map method for minimization of switching functions.
- To introduce the basic tools for design of combinational and sequential digital logic.
- To learn simple digital circuits in preparation for computer engineering.

Course Outcomes:

A student who successfully fulfills the course requirements will have demonstrated:

- An ability to define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.
- An ability to understand the different switching algebra theorems and apply them for logic functions.
- An ability to define the Karnaugh map for a few variables and perform an algorithmic reduction of logic functions.
- Students will be able to design various logic gates starting from simple ordinary gates to complex programmable logic devices & arrays.
- Students will be able to design various sequential circuits starting from flip-flop to registers and counters.

UNIT I

Digital Systems and Binary Numbers: Digital Systems, Binary Numbers, Octal and Hexadecimal Numbers, Complements of Numbers, Signed Binary Numbers, Arithmetic addition and subtraction, 4-bit codes: BCD, EXCESS 3, alphanumeric codes, 9's complement, 2421, etc..

UNIT II

Concept of Boolean algebra: Basic Theorems and Properties of Boolean algebra, Boolean Functions, Canonical and Standard Forms, Minterms and Maxterms. Gate level Minimization. Map Method, Three-Variable K-Map, Four Variable K-Maps. Products of Sum Simplification, Sum of Products Simplification, Don't – Care Conditions, NAND and NOR Implementation, Exclusive-OR Function.

UNIT III

Combinational Logic: Introduction, Analysis Procedure, Binary Adder–Subtractor, Binary Multiplier, Decoders, Encoders, Multiplexers, Demultiplexers, Priority Encoder, Code Converters, Magnitude Comparator, HDL Models of Combinational Circuits. Realization of Switching Functions Using PROM, PAL and PLA.

UNIT IV

Synchronous Sequential Logic: Introduction to Sequential Circuits, Storage Elements: Latches, Flip-Flops, RS-Latch Using NAND and NOR Gates, Truth Tables. RS, JK, T and D Flip Flops, Truth and Excitation Tables, Conversion of Flip Flops.

UNIT V

Registers and Counters: Registers, Shift Registers, Ripple Counters, Synchronous Counters, Ring Counter, Johnson Counter.

Text Books:

1. Digital Design, 5/e, M.Morris Mano, Michael D Ciletti, PEA.
2. Fundamentals of Logic Design, 5/e, Roth, Cengage.

Reference Books:

1. Digital Logic and Computer Design, M.Morris Mano, PEA.
2. Digital Logic Design, Leach, Malvino, Saha, TMH.
3. Modern Digital Electronics, R.P. Jain, TMH.

II Year - I Semester	Code: 20ML3L01	L	T	P	C
		0	0	3	1.5

UNIX AND SHELL PROGRAMMING LAB

List of Experiments

1. Study of Unix/Linux general purpose utility command list
   man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.
2. Perform the case study on:
   b) Study of vi editor.
   c) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
   d) Study of Unix/Linux file system (tree structure).
   e) Study of .bashrc, /etc/bashrc and Environment variables.
3. Write a shell script that takes a command –line argument and reports on whether it is directory, a file, or something else.
4. Write a shell script that accepts a file name starting and ending line numbers as arguments and displays all the lines between the given line numbers.
5. Write a shell script that computes the gross salary of a employee according to the following rules: i)If basic salary is < 1500 then HRA =10% of the basic and DA =90% of the basic. ii)If basic salary is >=1500 then HRA =Rs500 and DA=98% of the basic The basic salary is entered interactively through the key board.
6. Write a shell script that accepts two integers as its arguments and computers the value of first number raised to the power of the second number.
7. Write a shell script which receives two file names as arguments. It should check whether the two file contents are same or not. If they are same then second file should be deleted.
8. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
9. Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file.
10. Write a shell script to perform the following string operations: i)To extract a sub-string from a given string. ii)To find the length of a given string

II Year - I Semester	Code: 20ML3L01	L	T	P	C
		0	0	3	1.5

PYTHON PROGRAMMING LAB

Course Objectives:

The aim of Python Programming Lab is

- To acquire programming skills in core Python.
- To acquire Object Oriented Skills in Python
- To develop the skill of designing Graphical user Interfaces in Python
- To develop the ability to write database applications in Python

Course Outcomes:

By the end of this lab, the student is able to

- Write, Test and Debug Python Programs
- Use Conditionals and Loops for Python Programs
- Use functions and represent Compound data using Lists, Tuples and Dictionaries
- Use various applications using python

List of Experiments

1) Write a program that asks the user for a weight in kilograms and converts it to pounds. There are 2.2 pounds in a kilogram.
2) Write a program that asks the user to enter three numbers (use three separate input statements). Create variables called total and average that hold the sum and average of the three numbers and print out the values of total and average.
3) Write a program that uses a *for* loop to print the numbers 8, 11, 14, 17, 20, . . . , 83, 86, 89.
4) Write a program that asks the user for their name and how many times to print it. The program should print out the user's name the specified number of times.
5) Generate a random number between 1 and 10. Ask the user to guess the number and print a message based on whether they get it right or not.
6) Write a program that asks the user for two numbers and prints *Close* if the numbers are within .001 of each other and Not close otherwise.
7) Write a program that asks the user to enter a word and prints out whether that word contains any vowels.
8) In algebraic expressions, the symbol for multiplication is often left out, as in 3x+4y or 3(x+5). Computers prefer those expressions to include the multiplication symbol, like 3*x+4*y or 3*(x+5). Write a program that asks the user for an algebraic expression and then inserts multiplication symbols where appropriate.
9) Write a program that generates a list of 20 random numbers between 1 and 100.
Print the list.
Print the average of the elements in the list.
Print the largest and smallest values in the list.
Print the second largest and second smallest entries in the list
Print how many even numbers are in the list.
10) Write a program that asks the user for an integer and creates a list that consists of the factors of that integer.
11) Write a program that generates 100 random integers that are either 0 or 1. Then find the longest run of zeros, the largest number of zeros in a row. For instance, the longest run of zeros in [1,0,1,1,0,0,0,0,1,0,0] is 4.
12) Write a program that removes any repeated items from a list so that each item appears at most once. For instance, the list [1,1,2,3,4,3,0,0] would become [1,2,3,4,0].
13) Write a function called *sum_digits* that is given an integer num and returns the sum of the digits of num.
14) Write a function called *first_diff* that is given two strings and returns the first location in which the strings differ. If the strings are identical, it should return -1.
15) Write a function called *number_of_factors* that takes an integer and returns how many factors the number has.
16) Write a function called *is_sorted* that is given a list and returns True if the list is sorted and False otherwise.
17) Write a function called root that is given a number x and an integer n and returns $x^{1/n}$. In the function definition, set the default value of n to 2.
18) Write a function called primes that is given a number n and returns a list of the first n primes. Let the default value of n be 100.

19) Write a function called merge that takes two already sorted lists of possibly different lengths, and merges them into a single sorted list.
Do this using the sort method. (b) Do this without using the sort method.
20) Write a program that asks the user for a word and finds all the smaller words that can be made from the letters of that word. The number of occurrences of a letter in a smaller word can't exceed the number of occurrences of the letter in the user's word.
21) Write a program that reads a file consisting of email addresses, each on its own line. Your program should print out a string consisting of those email addresses separated by semicolons.
22) Write a class called Time whose only field is a time in seconds. It should have a method called *convert_to_minutes* that returns a string of minutes and seconds formatted as in the following example: if seconds is 230, the method should return '5:50'. It should also have a method called *convert_to_hours* that returns a string of hours, minutes, and seconds formatted analogously to the previous method.
23) Write a Python class to implement pow(x, n).
24) Write a Python class to reverse a string word by word.
25) Write a program that opens a file dialog that allows you to select a text file. The program then displays the contents of the file in a textbox.
26) Write a program to demonstrate Try/except/else.
27) Write a program to demonstrate try/finally and with/as.

II Year - I Semester	Code: 20CS3L08	L	T	P	C
		0	0	3	1.5

R-PROGRAMMING LAB

Course Objective: In this course student will learn about the fundamentals of R programming, standard R libraries, solid understanding of R functions, write programs using the R and gain skills in R programming Language, get acquaintances with Arrays, Files, Strings, Packages, and distributions using R.

Course Outcomes:

By the end of this lab, the student is able to

- Implement basic concepts of R programming, and its different module that includes conditional, looping, lists, Strings, Functions, Frames, Arrays, and File programming.
- Implement the concepts of R Script to extract the data from data frames and file operations.
- Implement the various statistical techniques using R.
- Extend the functionality of R by using add-on packages
- Use R Graphics and Tables to visualize results of various statistical operations on data

List of Experiments

Week 1:

Installing R and RStudio

Basic functionality of R, variable, data types in R

Week 2:

a) Implement R script to show the usage of various operators available in R language.

b) Implement R script to read person‘s age from keyboard and display whether he is eligible for voting or not.

c) Implement R script to find biggest number between two numbers.

d) Implement R script to check the given year is leap year or not.

Week 3:

a) Implement R Script to generate first N natural numbers.

b) Implement R Script to check given number is palindrome or not.

c) Implement R script to print factorial of a number.

d) Implement R Script to check given number is Armstrong or not.

Week 4:

a) Implement R Script to perform various operations on string using string libraries.

b) Implement R Script to check given string is palindrome or not.

c) Implement R script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it.

d) d) Implement R script for Call-by-value and Call-by-reference

Week 5:

a) Implement R Script to create a list.

b) Implement R Script to access elements in the list.

c) Implement R Script to merge two or more lists. Implement R Script to perform matrix operation

Week 6:

Implement R script to perform following operations:

d) various operations on vectors

e) Finding the sum and average of given numbers using arrays.

f) To display elements of list in reverse order.

g) Finding the minimum and maximum elements in the array.

Week 7:

a) Implement R Script to perform various operations on matrices

b) Implement R Script to extract the data from data frames.

c) Write R script to display file contents.

d) Write R script to copy file contents from one file to another

Week 8:

a) Implement R Script to create a Pie chart, Bar Chart, scatter plot and Histogram.
b) Implement R Script to perform mean, median, mode, range, summary, variance, standard deviation operations.
Introduction to ggplot2 graphics

Week 9:

a) Implement R Script to perform Normal, Binomial distributions.
b) Implement R Script to perform correlation, Linear and multiple regression.

Week 10:

Introduction to Non-Tabular Data Types: Time series, spatial data, Network data.
Data Transformations: Converting Numeric Variables into Factors, Date Operations, String Parsing, Geo coding

Week 11:

Introduction Dirty data problems: Missing values, data manipulation, duplicates, forms of data dates, outliers, spelling

Week 12:

Data sources: SQLite examples for relational databases, Loading SPSS and SAS files, Reading from Excel and Google Spreadsheets, API and web scraping examples

II Year - I Semester	Code: 20CS3S01	L	T	P	C
		1	0	2	2

JAVA SCRIPT

UNIT I

An introduction: An Introduction to JavaScript,Code editors, Developer console JavaScript Fundamentals: Hello, world!, Code structure, Variables, Data types, Interaction: alert, prompt, confirm, Type Conversions

UNIT II

Basic operators, maths, Comparisons, Conditional branching: if, '?', Logical operators, Loops: while and for The "switch" statement

UNIT III

Functions, Function expressions, Arrow functions, the basics JavaScript specials

UNIT IV

Objects: the basics: Objects, Number, Boolean, Strings, Math, Reg Exp, HTML DOM Date and time

UNIT V

JSON

II Year – I SEMESTER

L	T	P	C
2	0	0	0

ENVIRONMENTAL SCIENCE

Course Code: 20BS3M03

Syllabus:

UNIT – I Multidisciplinary nature of Environmental Studies: Definition, Scope and Importance – Sustainability: Stockholm and Rio Summit–Global Environmental Challenges: Global warming and climate change, Carbon Credits, acid rains, ozone layer depletion, population growth and explosion, effects. Role of information Technology in Environment and human health. Ecosystems: Concept of an ecosystem. - Structure and function of an ecosystem. - Producers, consumers and decomposers. - Energy flow in the ecosystem - Ecological succession. – Food chains, food webs and ecological pyramids. - Introduction, types, characteristic features, structure and function of Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems.

UNIT – II Natural Resources: Natural resources and associated problems Forest resources – Use and over – exploitation, deforestation – Timber extraction – Mining, dams and other effects on forest and tribal people Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems

Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, Sustainable mining of Granite, Literate, Coal, Sea and River sands.

Food resources: World food problems, changes caused by non-agriculture activities-effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity

Energy resources: Growing energy needs, renewable and non-renewable energy sources use of alternate energy sources Vs Oil and Natural Gas Extraction.

Land resources: Land as a resource, land degradation, Wasteland reclamation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.

UNIT – III Biodiversity and its conservation: Definition: genetic, species and ecosystem diversity-classification - Value of biodiversity: consumptive use, productive use, social- Biodiversity at national and local levels. India as a mega-diversity nation - Hot-spots of biodiversity - Threats to biodiversity: habitat loss, man-wildlife conflicts - Endangered and endemic species of India – Conservation of biodiversity: conservation of biodiversity.

UNIT – IV Environmental Pollution: Definition, Cause, effects and control measures of Air pollution, Water pollution, Soil pollution, Noise pollution, Nuclear hazards. Role of an individual in prevention of pollution. - Pollution case studies, Sustainable Life Studies. Impact of

Fire Crackers on Men and his well being. Solid Waste Management: Sources, Classification, effects and control measures of urban and industrial solid wastes. Consumerism and waste products, Biomedical, Hazardous and e – waste management.

UNIT – V Social Issues and the Environment: Urban problems related to energy –Water conservation, rain water harvesting-Resettlement and rehabilitation of people; its problems and concerns. Environmental ethics: Issues and possible solutions. Environmental Protection Act - Air (Prevention and Control of Pollution) Act. –Water (Prevention and control of Pollution) Act

-Wildlife Protection Act -Forest Conservation Act-Issues involved in enforcement of environmental legislation. -Public awareness.

UNIT – VI Environmental Management: Impact Assessment and its significance various stages of EIA, preparation of EMP and EIS, Environmental audit. Ecotourism, Green Campus – Green business and Green politics.
The student should Visit an Industry / Ecosystem and submit a report individually on any issues related to Environmental Studies course and make a power point presentation.

Text Books:
1. Environmental Studies, K. V. S. G. Murali Krishna, VGS Publishers, Vijayawada
2. Environmental Studies, R. Rajagopalan, 2nd Edition, 2011, Oxford University Press.
3. Environmental Studies, P. N. Palanisamy, P. Manikandan, A. Geetha, and K. Manjula Rani;
Pearson Education, Chennai

Reference:
1. Text Book of Environmental Studies, Deeshita Dave & P. Udaya Bhaskar, Cengage Learning.
2. A Textbook of Environmental Studies, Shaashi Chawla, TMH, New Delhi
3. Environmental Studies, Benny Joseph, Tata McGraw Hill Co, New Delhi
4. Perspectives in Environment Studies, Anubha Kaushik, C P Kaushik, New Age International
Publishers, 2014

II Year II Semester

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
DETAILED SYLLABUS – BR20
For U.G – B.Tech – *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

II Year – II SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20BS4T04	Numerical Methods and Complex Variables	3	0	0	3
2	20CS4T06	Java Programming	3	0	0	3
3	20AD4T02	Data Science	3	0	0	3
4	20CS4T07	Operating Systems	3	0	0	3
5	20CS4T08	Mathematical Foundations of Computer Science	3	0	0	3
6	20CS4L07	Java Programming Lab	0	0	3	1.5
7	20CS4L09	Operating Systems Lab	0	0	3	1.5
8	20AD4L02	Data Science Lab	0	0	3	1.5
9	20CS4S02	Skill Oriented Course - MongoDB	1	0	2	2
		Total	16	0	11	21.5

Numerical Methods and Complex Variable
(Common to all Branches)

Subject code: 20BS4T04

L	T	P	C
3	0	0	3

Course Objectives:

COB 1: To help the student Understand the numerical way of solving equations and integration.
COB 2: To become familiar with the concepts of Complex Functions.

Course Outcomes:

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

CO 1: Use Numerical Methods in Solving Equations. (L3)
CO 2: Calculate function value and its integration value by using different numerical methods. (L3)
CO 3: Understand Methods of finding numerical solution of differential equations. (L2)
CO 4: Use the knowledge of Beta and Gamma functions in evaluating improper integrals(L2).
CO 5: Understand Functions of complex variables and will be able to find analytic Function. (L2)
CO 6: Evaluate definite integral of given complex function using Cauchy's theorem and residues. (L5)

Unit I: - Solution of Algebraic and Transcendental Equations

Introduction- Bisection method – Method of false position (Regula-False Method) – Iteration method – Newton- Raphson method (One Variable)

Unit II: - Interpolation and Numerical Integration

Introduction- Errors in polynomial interpolation – Finite differences- Forward differences- Backward Differences –Central differences – Symbolic relations and separation of symbols - Differences of a Polynomial-Newton's formulae for interpolation – Interpolation with unequal intervals - Lagrange's Interpolation formula-Trapezoidal rule- Simpson's 1/3rd and 3/8th rule.

Unit III: - Numerical solution of Ordinary Differential Equations with initial conditions

Solution of ordinary differential equations with initial conditions by Taylor's series Method - Picard Method of successive Approximations Euler's method, Modified Euler's Method – Runge-Kutta method (second and fourth order).

UNIT IV: Special functions

Def of improper integral, Beta and Gamma functions- Properties - Relation between Beta and Gamma functions- Evaluation of improper integrals

Unit-V: - Functions of Complex variable & Complex Integration

Complex function, Real and Imaginary parts of Complex function, Limit**,** Continuity and Derivative of
Complex function, Cauchy-Riemann equations, Analytic function, Entire function, singular point, conjugate
Function**,** *C -R* equations in polar form, Harmonic functions, Milne-Thomson method**.**

Line integral of a complex function, Cauchy's theorem (only statement) -Cauchy's Integral Formula (only statement), Taylor's series(only statement),Laurent's series (only statement)-Zeros of an analytic function- Types of Singularities-pole of order m, simple pole- Residues, Residue theorem (only statement).
Evaluation of integrals of the type

(a) Improper real integrals $\int_{-\infty}^{\infty} f(x)dx$ (b) $\int_{0}^{2\pi} f(\cos\theta, \sin\theta)d\theta$

Text Books:

1. **B.S.GREWAL,** Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **B. V. Ramana**, Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.
3. **Dr T K V Iyengar**, **Dr B.Krishna Gandhi**, **S.Ranhanatham** and Dr. **M.V.S.S.N Prasad**, Engineering Mathematics, S.Chand & Company Ltd.

Reference Books:

1. **N.P.Bali & Manish Goyal**, A Text book of Engineering Mathematics, Lakshmi Publications.
2. **James ward brown, Ruel V.Churchill**, Complex variable and Applications, Mc.Graw Hill Publications
3. **ERWIN KREYSZIG,** Advanced Engineering Mathematics, 10th Edition, Wiley-India
4. **Steven C. Chapra**, Applied Numerical Methods with MATLAB for Engineering and Science, Tata Mc. Graw Hill Education

.

II Year - II Semester	Code: 20CS4T06	L	T	P	C
		3	0	0	3

JAVA PROGRAMMING

Course Objectives:

- Understanding the OOP's concepts, classes and objects, threads, files, applets, swings and act.
- This course introduces computer programming using the JAVA programming language with object-oriented programming principles.
- Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using Java for network level programming and middleware development

Course Outcomes:

- Understand Java programming concepts and utilize Java Graphical User Interface in Program writing.
- Write, compile, execute and troubleshoot Java programming for networking concepts.
- Build Java Application for distributed environment.
- Design and Develop multi-tier applications.
- Identify and Analyze Enterprise applications.

UNIT I

Introduction to OOP, procedural programming language and object oriented language, principles of OOP, applications of OOP, history of java, java features, JDK, JVM, program structure. Variables, primitive data types, identifiers, literals, operators, expressions, precedence rules and associatively, primitive type conversion and casting, flow of control.

UNIT II

Arrays, command line arguments, Classes and objects, class declaration, creating objects, methods, constructors and constructor overloading, garbage collector, importance of static keyword and examples, this keyword, nested classes.

UNIT III

Inheritance, types of inheritance, super keyword, final keyword, overriding and abstract class. Interfaces, creating the packages, using packages, importance of CLASSPATH and java.lang package. Exception handling, importance of try, catch, throw, throws and finally block, user defined exceptions, Assertions.

UNIT IV

Multithreading: introduction, thread life cycle, creation of threads, thread priorities, thread synchronization, communication between threads. Reading data from files and writing data to files, random access file.

UNIT V

. Introduction to Java FX, AWT: introduction, components and containers, Button, Label, Checkbox, Radio Buttons, List Boxes, Choice Boxes, Container class, Layouts, Menu and Scrollbar.

Text Books:

1. The complete Reference Java, 8th edition, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, SaurabhChoudary, Oxford.
3. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson.

Reference Books:

1) Introduction to java programming, 7th edition by Y Daniel Liang,Pearson
2) Constructive JAVA Programming, Sagayaraj, Denis, Karthik, Gajalakshmi, Universities Press.
3) JAVA Programming for Core and Advanced Learners, Sagayaraj, Denis, Karthik, Gajalakshmi, Universities Press.

.

II Year - II Semester	Code: 20AD4T02	L	T	P	C
		3	0	0	3

DATA SCIENCE

Course Objectives:

- To provide a strong foundation of fundamental concepts in Data Science.
- To provide a basic exposition to the goals and methods of Data Science.
- To familiarize with Python libraries for Data Analysis and Data Visualization.
- Produce Python code to statistically analyze a dataset.

Course Outcomes:

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

- Enumerate the history and foundations of Data Science
- Apply the basic principles of Data Science in problem solving
- Demonstrate various mathematical operations on arrays using NumPy
- Analyze and manipulate Data using Pandas
- Creating static, animated, and interactive visualizations using Matplotlib.

UNIT I

What is Data science? Datafication, Exploratory Data Analysis, The Data science process, A data scientist role in this process.

UNIT II

NumPy Basics: The NumPy ndarray: A Multidimensional Array Object, Creating ndarrays ,Data Types for ndarrays, Operations between Arrays and Scalars, Basic Indexing and Slicing, Boolean Indexing, Fancy Indexing, Data Processing Using Arrays, Expressing Conditional Logic as Array Operations, Methods for Boolean Arrays , Sorting , Unique.

UNIT III

Getting Started with pandas: Introduction to pandas, Library Architecture, Features, Applications, Data Structures, Series, Data Frame, Index Objects, Essential Functionality Reindexing, Dropping entries from an axis, Indexing, selection, and filtering),Sorting and ranking, Summarizing and Computing Descriptive Statistics, Unique Values, Value Counts, Handling Missing Data, filtering out missing data.

UNIT IV

Visualizing Data: matplotlib, Bar charts, Line charts, Scatter plots, Statistics: Describing a single set of data, Central tendencies, Dispersion, Correlation, Simpson's paradox, Probability: dependence and independence, Conditional Probability, Bayes's Theorem, Random Variables, Continuous distribution, Normal distribution.

UNIT V

Data Loading, storage and File formats: Reading and writing data in text formats, Reading text files in pieces, Writing data out to text format, JSON Data, XML and HTML: Web scripting, Binary Data Formats, Interacting with HTML and Web APIs, Interacting with Databases, Storing and Loading Data in MongoDB.

Text Books:

1. Wes McKinney, "Python for Data Analysis",O'REILLY, ISBN:978-1-449-31979-3, 1st edition, October 2012.
2. Cathy O'Neil, Rachel Schutt, Doing Data Science, Straight Talk from the Frontline. O'Reilly, 2013.
3. Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media, 2015

Reference Books:

1. Python: The Complete Reference, Martin C. Brown, McGraw Hill Education
2. Head First Python, Paul Barry, O'Reilly
3. J. Leskovec, A. Rajaraman, J.D. Ullman. Mining of Massive Datasets. Cambridge University Press. (Indian Edition; Online pdf is available for download)

II Year - II Semester	Code: 20CS4T07	L	T	P	C
		3	0	0	3

OPERATING SYSTEMS

Course Objectives:

The objectives of this course is to

- Introduce to the internal operation of modern operating systems
- Define, explain, processes and threads, mutual exclusion, CPU scheduling, deadlock, memory management, and file systems
- Understand File Systems in Operating System like UNIX/Linux and Windows
- Understand Input Output Management and use of Device Driver and Secondary Storage (Disk) Mechanism
- Analyze Security and Protection Mechanism in Operating System

Course Outcomes:

After learning, the course the students should be able to:

- Describe various generations of Operating System and functions of Operating System
- Describe the concept of program, process and thread and analyze various CPU Scheduling Algorithms and compare their performance
- Solve Inter Process Communication problems using Mathematical Equations by various methods
- Compare various Memory Management Schemes especially paging and Segmentation in Operating System and apply various Page Replacement Techniques
- Outline File Systems in Operating System like UNIX/Linux and Windows

UNIT I

Operating Systems Overview: Operating system functions, Operating system structure, Operating systems operations, Computing environments, Open-Source Operating Systems. System Structures: Operating System Services, User and Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, operating system debugging, System Boot.

UNIT II

Process Concept: Process scheduling, Operations on processes, Inter-process communication, Communication in client server systems. Multithreaded Programming: Multithreading models, Thread libraries, Threading issues. Process Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processors scheduling, Thread scheduling. Inter-process Communication: Race conditions, Critical Regions, Mutual exclusion with busy waiting, sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing, Barriers, Classical IPC Problems - Dining philosophers problem, Readers and writers problem.

UNIT III

Memory-Management Strategies: Introduction, Swapping, Contiguous memory allocation, Paging, Segmentation. Virtual Memory Management: Introduction, Demand paging, Copy on-write, Page replacement, Frame allocation, Thrashing, Memory-mapped files, Kernel memory allocation.

UNIT IV

Deadlocks: Resources, Conditions for resource deadlocks, Ostrich algorithm, Deadlock detection and recovery, Deadlock avoidance, Deadlock prevention.

UNIT V

File Systems: Files, Directories, File system implementation, management and optimization. Secondary-Storage Structure: Overview of disk structure, and attachment, Disk scheduling, RAID structure, Stable storage implementation.

Text Books:

1) Silberschatz A, Galvin P B, and Gagne G, Operating System Concepts, 9th edition, Wiley, 2013.
2) Tanenbaum A S, Modern Operating Systems, 3rd edition, Pearson Education, 2008.

Reference Books:

1) Dhamdhere D M, Operating Systems A Concept Based Approach, 3rd edition, Tata McGraw-Hill,2012.
2) Stallings W, Operating Systems -Internals and Design Principles, 6th edition, Pearson Education,2009
3) Nutt G, Operating Systems, 3rd edition, Pearson Education,2004.

II Year - II Semester	Code: 20CS4T08	L	T	P	C
		3	0	0	3

MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

Course Objectives:

This course is designed to:

- To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning
- To introduce a wide variety of applications. The algorithmic approach to the solution of problems is fundamental in discrete mathematics, and this approach reinforces the close ties between this discipline and the area of computer science

Course Outcomes:

At the end of the course student will be able to

- Demonstrate skills in solving mathematical problems
- Comprehend mathematical principles and logic
- Demonstrate knowledge of mathematical modeling and proficiency in using mathematical software
- Manipulate and analyze data numerically and/or graphically using appropriate Software
- Communicate effectively mathematical ideas/results verbally or in writing

UNIT I

Mathematical Logic: Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicate Calculus: Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

UNIT II

Set Theory: Sets: Operations on Sets, Principle of Inclusion-Exclusion, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams, Functions: Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties, Algebraic Structures: Algebraic Systems, Properties, Semi Groups and Monoids, Group, Subgroup and Abelian Group, Homomorphism, Isomorphism.

UNIT III

Combinatorics: Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems, Number Theory: Properties of Integers, Division Theorem, Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic, Fermat's and Euler's Theorems

UNIT IV

Recurrence Relations: Generating Functions, Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions, Recurrence Relations, and Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations.

UNIT V

Graph Theory: Basic Concepts, Graph Theory and its Applications, Sub graphs, Graph Representations: Adjacency and Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Bipartite and Planar Graphs, Euler's Theorem, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Prim's and Kruskal's Algorithms, BFS and DFS Spanning Trees.

Text Books:

1) Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGrawHill.
2) Elements of Discrete Mathematics-A Computer Oriented Approach, C. L. Liu and D. P. Mohapatra, 3rd Edition, Tata McGrawHill.

Reference Books:

1) Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel and T. P. Baker, 2nd Edition, Prentice Hall of India.
2) Discrete Mathematical Structures, Bernand Kolman, Robert C. Busby and Sharon Cutler Ross, PHI.
3) Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGrawHill.

II Year - II Semester	Code: **20CS4L07**	L	T	P	C
		0	0	3	1.5

JAVA PROGRAMMING LAB

List of Experiments

Exercise - 1 (Basics)
a). Write a JAVA program to display default value of all primitive data type of JAVA
b) Write a case study on **public static void main(250 words)**
Exercise - 2 (Operations, Expressions, Control-flow, Strings)
a). Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
(b) Write a JAVA program using StringBuffer to delete, remove character.
Exercise - 3 (Class, Objects)
a). Write a JAVA program to implement class mechanism. – Create a class, methods and invoke them inside main method.
b). Write a JAVA program to implement constructor.
Exercise - 4 (Methods)
a). Write a JAVA program to implement constructor overloading.
b). Write a JAVA program implement method overloading.
Exercise - 5 (Inheritance)
a). Write a JAVA program to implement multi level Inheritance
b). Write a java program for abstract class to find areas of different shapes
Exercise - 6 (Inheritance - Continued)
a). Write a JAVA program give example for "super" keyword.
b). Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?
Exercise - 7 (Exception)
a).Write a JAVA program that describes exception handling mechanism
b).Write a JAVA program Illustrating Multiple catch clauses.
c). Write a JAVA program that implements Runtime polymorphism
Exercise – 8 (User defined Exception)
a). Write a JAVA program to Illustrate try, catch.
b). Write a JAVA program to Illustrate finally
c).Write a JAVA program for creation of User Defined Exception
Exercise – 9 (Threads)
a). Write a JAVA program that creates threads by extending Thread class .First thread display"Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds ,(Repeat the same by implementing Runnable)
b). Write a program illustrating **isAlive** and **join ()**
c). Write a Program illustrating Daemon Threads.
d).Write a JAVA program Producer Consumer Problem
Exercise – 10 (Packages)
a). Write a JAVA program illustrate class path
b). Write a JAVA program to create an import a package.

II Year - II Semester	Code: **20AD4L02**	L	T	P	C
		0	0	3	1.5

DATA SCIENCE LAB

Course Objectives:

- To familiarize Python Libraries for Data Collection
- To Apply different Data Cleaning methods for Data Preparation
- To Apply appropriate statistical methods for Data Preparation
- To Understand characteristics of Data-by-Data Visualization techniques
- To Identify and apply appropriate Libraries for Data Analysis

Course Outcomes:

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

- Apply appropriate tools for Data Collection and Manipulation
- Will be able to identify and apply appropriate Data Cleaning techniques for Data Preparation
- Elucidate statistical measures to Analyze the nature of Data.
- Implement Data Visualization Methods for getting insights of Data.
- Will be able to Analyze Data by implementing Concepts of Data Preparation

WEEK1:

1. Install Jupyter/NumPy/Panda in Python
2. Study of Jupyter Notebook System

WEEK 2:

1. Write a Program in Python to Manipulate, Aggregate and Analyze data using NumPy
2. Write a Program in Python to Handle and Analyze data using Pandas
3. Write a Program in Python to Read and write different types of Files (csv, json, txt etc).

WEEK 3:

1. Write program to create a list, manipulate and slices it.
2. Create a new list and add elements to it from another list, and creates a matrix from two lists
3. Create same a, b steps for Tuple and Dictionary

WEEK 4:

Creating a NumPy Array

a. Basic ndarray
b. Array of zeros
c. Array of ones
d. Random numbers in ndarray
e. An array of your choice
f. Imatrix in NumPy
g. Evenly spaced ndarray

WEEK 5:

The Shape and Reshaping of NumPy Array

a. Dimensions of NumPy array
b. Shape of NumPy array
c. Size of NumPy array
d. Reshaping a NumPy array
e. Flattening a NumPy array
f. Transpose of a NumPy array

WEEK 6:

Indexing and Slicing of NumPy Array

a. Slicing 1-D NumPy arrays
b. Slicing 2-D NumPy arrays
c. Slicing 3-D NumPy arrays

d. Negative slicing of NumPy arrays

WEEK 7:

Expanding and Squeezing a NumPy Array

a. Expanding a NumPy array
b. Squeezing a NumPy array
c. Sorting in NumPy Arrays

WEEK 8:

Stacking and Concatenating Numpy Arrays

a. Stacking ndarrays
b. Concatenating ndarrays
c. Broadcasting in Numpy Arrays

WEEK 9:

Perform following operations using pandas

a. Creating dataframe
b. concat()
c. Setting conditions
d. Adding a new column

WEEK 10:

Perform following operations using pandas

a. Filling NaN with string
b. Sorting based on column values
c. groupby()

WEEK 11:

Read the following file formats using pandas

a. Text files
b. CSV files
c. Excel files
d. JSON files

WEEK 12:

Perform following visualizations using matplotlib

a. Bar Graph
b. Pie Chart
c. Box Plot
d. Histogram
e. Line Chart and Subplots
f. Scatter Plot

WEEK 13:

Given the *iris dataset*: https://archive.ics.uci.edu/ml/datasets/iris

1. How many rows does it contain? How many columns?
2. Compute the average petal length
3. Compute the average of all numerical columns
4. Extract the petal length outliers (i.e. those rows whose petal length is 50% longer than the average petal length)
5. Extract the petal length outliers (as above) for each iris species
6. Compute the standard deviation of all columns, for each iris species 6.Extract the petal length outliers (as above) for each iris species
7. Extract the group-wise petal length outliers, i.e. find the outliers (as above) for each iris species using group by (), aggregate, and merge ().

Write a python program to compute all the functionalities of the above-mentioned data.

WEEK 14:

Consider the sample data

people = ('G1','G2','G3','G4','G5','G6','G7','G8')
segments = 4
multi-dimensional data= [[3.40022085, 7.70632498, 6.4097905, 10.51648577, 7.5330039,

7.1123587, 12.77792868, 3.44773477],
[11.24811149, 5.03778215, 6.65808464, 12.32220677, 7.45964195, 6.79685302, 7.24578743, 3.69371847],
[3.94253354, 4.74763549, 11.73529246, 4.6465543, 12.9952182, 4.63832778, 11.16849999, 8.56883433],
[4.24409799, 12.71746612, 11.3772169, 9.00514257, 10.47084185, 10.97567589, 3.98287652, 8.80552122]]
Write a Python program to create stack bar plot and add label to each section.

Text Books:

4. Cathy O'Neil and Rachel Schutt. Doing Data Science, Straight Talk from The Frontline. O'Reilly.
5. Jure Leskovek, AnandRajaraman and Jeffrey Ullman.Mining of Massive Datasets. v2.1, Cambridge University Press.

Reference Books:

1. Joel Grus, Data Science from Scratch, O'Reilly Publications.
2. Davy Ceilen, Introducing Data Science: Big Data, Machine Learning, and More, Using Python Tools, DreamTech Publications.

II Year - II Semester	Code: **20CS4L09**	L	T	P	C
		0	0	3	1.5

OPERATING SYSTEMS LAB

List of Experiments

1. Simulate the following CPU scheduling algorithms

a) Round Robin b) SJF c) FCFS d) Priority

2. Loading executable programs into memory and execute System Call implementation-read(), write(), open () and close()

3. Multiprogramming-Memory management- Implementation of Fork(), Wait(), Exec() and Exit() System calls

4. Simulate all File allocation strategies

a) Sequenced, b)Indexed, c) Linked

5. Simulate MVT and MFT

6. Simulate all File Organization Techniques

a) Single level directory b) Two level c) Hierarchical d) DAG

7. Simulate Bankers Algorithm for Dead Lock Avoidance

8. Simulate Bankers Algorithm for Dead Lock Prevention.

9. Simulate all page replacement algorithms.

a) FIFO b) LRU c) LFU etc….

10. Simulate Paging Technique of memory management.

II Year - II Semester	Code: **20CS4L09**	L	T	P	C
		0	0	3	1.5

Skill Oriented Course: MONGO DB

Course Outcomes:

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

- Installing and configuring mongo DB in windows
- Perform all database operations using mongo DB
- Develop applications by integrating mongo DB with java/PHP.

List of Experiments:

1. Mongo DB installation and configuration in windows.
2. Demonstrate how to create and drop a database in Mongo DB.
3. Creating the Collection in Mongo DB on the fly
4. Creating collection with options before inserting the documents and drop the collection created.
5. Mongo DB insert document
   a. Insert single document
   b. Insert multiple documents in collection
6. Querying all the documents in json format and Querying based on the criteria.
7. Mongo DB update document
   a. Using update() method.
   b. Using save() method.
8. Mongo DB delete document from a collection.
   a. Using remove() method.
   b. Remove only one document matching your criteria
   c. Remove all documents
9. Mongo DB Projection
10. limit(), skip(), sort() methods in Mongo DB
11. Mongo DB indexing
    a. Create index in Mongo DB
    b. Finding the indexes in a collection
    c. Drop indexes in a collection
    d. Drop all the indexes
12. Mongo DB with java and PHP
    a. Create a simple application that uses Mongo DB with Java
    b. Create a simple application that uses Mongo DB with PHP

Web References:

1. https://beginnersbook.com/2017/09/mongodb-tutorial/

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DETAILED SYLLABUS FOR REGULATIONS – BR20

UG: B.Tech - ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

III Year – I SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20CS5T09	Computer Networks	3	0	0	3
2	20AD5T03	Machine Learning	3	0	0	3
3	20CS5T11	Data Base Management System	3	0	0	3
4	20ML5E01 20CS5E02 20CE5E01 20ME5E01	**Open Elective- I** 5. Artificial Intelligence 6. Object Oriented Programming 7. Interior Space Design 8. Industrial Automation and Robotics	3	0	0	3
5	20CS5D01 20CS5D03 20AD5D01 20AD5D02	**Professional Elective- I** 1. Computer Graphics 2. Software Engineering 3. Object Oriented Analysis and Design 4. DevOps/**MOOCs -1**	3	0	0	3
6	20CS5L12	DBMS Lab	0	0	3	1.5
7	20AD5L03	Machine Learning Lab	0	0	3	1.5
8	20CS5P01	Summer Internship	0	0	3	1.5
9	20AD5S01	Skill Advanced Course: Continuous Integration and Continuous Delivery using DevOps	1	0	2	2
10	20HS5M04	Foreign Language	2	0	0	0
		Total	18	0	11	21.5

III Year - I Semester	Code: 20CS5T09	L	T	P	C
		3	0	0	3

COMPUTER NETWORKS

Course Outcomes:

After completing this course the student must demonstrate the knowledge and ability to:

1. Independently understand basic computer network technology and identify the different types of network topologies and protocols.
2. Understand Network models and Physical layer
3. Study the techniques used in data link layer.
4. Understand the routing strategies for an IP based networking

UNIT I

Introduction: OSI overview, TCP/IP and other networks models, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.

UNIT II

Physical Layer – Guided Transmission Media, Digital Modulation and Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Code Division Multiplexing.

UNIT III

The Data Link Layer - Services Provided to the Network Layer – Framing – Error Control Flow Control, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols. **Random Access**: ALOHA, MAC addresses, carrier sense multiple access (CSMA), CSMA/CD, CSMA/CA, controlled Access: Reservation, Polling, Token Passing, channelization: FDMA, TDMA, CDMA.

UNIT IV

Network Layer: Virtual circuit and Datagram subnets-Routing algorithm shortest path routing, Flooding, Hierarchical routing, Broad cast, Multi cast, distance vector routing.

Transport Layer: The transport service, elements of transport protocols, congestion control, the internet transport protocols.

UNIT V

Application Layer –The Domain Name System: The DNS Name Space, Resource Records, Name Servers, Electronic Mail: Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery, The wireless web: WAP—The Wireless Application Protocol

Text Books:

1. Computer Networks—Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI
2. Data Communications and Networks–Behrouz A. Forouzan. Third Edition TMH.

Reference Books:

1.An Engineering Approach to Computer Networks- S. Keshav, 2ndEdition, Pearson Education
2.Understanding communications and Networks, 3rd Edition, W.A.Shay, Thomson
3.Computer Networks, 5ed, David Patterson, Elsevier

III Year - I Semester	Code: 20AD5T03	L	T	P	C
		3	0	0	3

MACHINE LEARNING

Course Objectives:

- Identify problems that are amenable to solution by ANN methods, and which ML methods may be suited to solving a given problem.
- Formalize a given problem in the language/framework of different ANN methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).

Course Outcomes:

After the completion of the course, student will be able to

- Explain the fundamental usage of the concept Machine Learning system
- Demonstrate on various regression Technique
- Analyze the Ensemble Learning Methods

UNIT I

Introduction- Artificial Intelligence, Machine Learning, Deep learning, Types of Machine Learning Systems, Main Challenges of Machine Learning. **Statistical Learning**: Introduction, Supervised and Unsupervised Learning, Training and Test Loss, Tradeoffs in Statistical Learning, Estimating Risk Statistics, Sampling distribution of an estimator, Empirical Risk Minimization.

UNIT II

Supervised Learning(Regression/Classification):Basic Methods: Distance based Methods, Nearest Neighbours, Decision Trees, Naive Bayes, **Linear Models:** Linear Regression, Logistic Regression, Generalized Linear Models, Support Vector Machines, **Binary Classification:** Multiclass/Structured outputs, MNIST, Ranking.

UNIT III

Ensemble Learning and Random Forests: Introduction, Voting Classifiers, Bagging and Pasting, Random Forests, Boosting, Stacking. **Support Vector Machine:** Linear SVM Classification, Nonlinear SVM Classification SVM Regression, Naïve Bayes Classifiers.

UNIT IV

Unsupervised Learning Techniques: Clustering, K-Means, Limits of K-Means, Using Clustering for Image Segmentation, Using Clustering for Preprocessing, Using Clustering for Semi-Supervised Learning, DBSCAN, Gaussian Mixtures. Dimensionality Reduction: The Curse of Dimensionality, Main Approaches for Dimensionality Reduction, PCA, Using Scikit-Learn, Randomized PCA, Kernel PCA.

UNIT V

Neural Networks and Deep Learning: Introduction to Artificial Neural Networks with Keras, Implementing MLPs with Keras, Installing TensorFlow 2, Loading and Preprocessing Data with TensorFlow.

Text Books:

Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Publications, 2019

Data Science and Machine Learning Mathematical and Statistical Methods,Dirk P. Kroese, Zdravko I. Botev, Thomas Taimre, Radislav Vaisman,25th November 2020

Reference Books:

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, The Elements of Statistical Learning, Springer 2009
2. Christopher Bishop, Pattern Recognition and Machine Learning, Springer, 2007

Machine Learning Probabilistic Approach, Kevin P. Murphy, MIT Press, 2012.

III Year - I Semester	20CS5T11	L	T	P	C
		3	0	0	3

DATA BASE MANAGEMENT SYSTEM

Course Objectives:

1. To introduce about database management systems
2. To give a good formal foundation on the relational model
3. To introduce the concepts of basic SQL as a universal Database language

Course Outcomes:

1. Describe fundamental concepts a relational database
2. Create, maintain and manipulate a relational database using SQL
3. Apply Conceptual and Logical database design

UNIT I

Introduction: Database system, Characteristics (Database Vs File System), Database Users (Actors on Scene, Workers behind the scene), Advantages of Database systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance and data independence; Three tier schema architecture for data independence; Database system structure, environment, Centralized and Client Server architecture for the database.

UNIT II

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance

BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions (Date and Time, Numeric, String conversion).

UNIT III

Entity Relationship Model: Introduction, Basic features of ER model, Representation of entities, attributes, entity set, relationship, relationship set, constraints, ER diagrams Generalization/specialization and Aggregation.

SQL: Creating tables with relationships, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, views(updatable and non-updatable), relational set operations.

UNIT IV

Schema Refinement (Normalization): Purpose of Normalization or schema refinement, concept of functional dependency, Closure of functional dependency and attribute closure, Normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-Codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF), Fifth Normal Form (5NF).

UNIT V

TRANSACTION MANAGEMENT Transaction, properties of transactions, transaction log, and transaction management with SQL using commit rollback and save point. Concurrency control for lost updates, uncommitted data, in consistent retrievals and the Scheduler. Concurrency control with locking methods: lock granularity, lock types, two phase locking for ensuring serializability, deadlocks, Concurrency control with time stamp ordering: Wait/Die and Wound/Wait Schemes. Indexing Techniques: Indexing, Cluster Indexes, Primary and Secondary Indexes , Index data Structures, Hash Based Indexing, B+ Trees: Searching, Insertion, Deletion

Text Books:

1.Database System Concepts by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, 7th Edition, McGraw-Hill Education, 2019.

2. Database Management Systems by Raghu Ramakrishnan, Johannes Gehrke, 3rd Edition., McGraw-Hill Education (India), 2014.

Reference Books:

1.Database Principles: Fundamentals of Design, Implementation, and Management by Steven Morris, Keeley Crockett, Carlos Coronel, Craig Blewett, Cengage, 2020.

2. Fundamentals of Database Systems by RamezElmasri, Shamkant B. Navathe, 7th Edition, Pearson Education India, 2015.

3. Introduction to Database Systems by C J Date, 8th Edition, Pearson Education, 2009.

III Year - I Semester	Code: 20ML5E01	L	T	P	C
		3	0	0	3

ARTIFICIAL INTELLIGENCE

Course Objectives:

Course Objectives:

- To provide a strong foundation of fundamental concepts in Artificial Intelligence.
- To provide a basic exposition to the goals and methods of Artificial Intelligence.

Course Outcomes:

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

- Enumerate the history and foundations of Artificial Intelligence
- Apply the basic principles of AI in problem solving
- Choose the appropriate representation of Knowledge

UNIT I

Introduction: What Is AI?, The Foundations of Artificial Intelligence, The History of Artificial Intelligence, The State of the Art, Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

UNIT II

Problem Solving: Problem-Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies, Informed (Heuristic) Search Strategies, Local Search Algorithms and Optimization Problems, Searching with Nondeterministic Actions.

UNIT III

Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

UNIT IV

Knowledge Representation: Knowledge-Based Agents, Logic, Propositional Logic: A Very Simple Logic, Ontological Engineering, Categories and Objects, Events, Mental Events and Mental Objects, Reasoning Systems for Categories, The Internet Shopping World.

UNIT V

Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

Text Books:

1) Stuart Russell and Peter Norvig, "Artificial Intelligence: A Modern Approach" , 3rd Edition, Pearson
2) Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

Reference Books:

1. Artificial intelligence, A modern Approach , 2nd edition, Stuart Russel, Peter Norvig, PEA
2. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rd edition, TMH
3. Introduction to Artificial Intelligence, Patterson, PHI

III Year - I Semester	Code : 20CS5E02	L	T	P	C
		3	0	0	3

OBJECT ORIENTED PROGRAMMING

Course Objectives:

- Understanding the OOP's concepts, classes and objects, threads, files, applets, swings and act.
- This course introduces computer programming using the JAVA programming language with object-oriented programming principles.

Course Outcomes:

- Understand Java programming concepts and utilize Java Graphical User Interface in Program writing.
- Write, compile, execute and troubleshoot Java programming for networking concepts.
- Build Java Application for distributed environment.

UNIT I

Introduction to OOP, procedural programming language and object oriented language, principles of OOP, applications of OOP, history of java, java features,JDK, JVM, program structure. Variables, primitive data types, identifiers, literals, operators, expressions, precedence rules and associatively, primitive type conversion and casting, flow of control.

UNIT II

Arrays, command line arguments, Classes and objects, class declaration, creating objects, methods, constructors and constructor overloading, garbage collector, importance of static keyword and examples, this keyword, nested classes.

UNIT III

Inheritance, types of inheritance, super keyword, final keyword, overriding and abstract class. Interfaces, creating the packages, using packages, importance of CLASSPATH and java.lang package. Exception handling, importance of try, catch, throw, throws and finally block, user defined exceptions, Assertions.

UNIT IV

Multithreading: introduction, thread life cycle, creation of threads, thread priorities, thread synchronization, communication between threads. Reading data from files and writing data to files, random access file

UNIT V

Introduction to Java FX, AWT: introduction, components and containers, Button, Label, Checkbox, Radio Buttons, List Boxes, Choice Boxes, Container class, Layouts, Menu and Scrollbar.

Text Books:

1. The complete Reference Java, 8th edition, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, SaurabhChoudary, Oxford.
3. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson.

Reference Books:

1) Introduction to java programming, 7th edition by Y Daniel Liang,Pearson
2) Constructive JAVA Programming, Sagayaraj, Denis, Karthik, Gajalakshmi, Universities Press.
3) JAVA Programming for Core and Advanced Learners, Sagayaraj, Denis, Karthik, Gajalakshmi, Universities Press.

III Year - I Semester	Code: 20CS5D01	L	T	P	C
		3	0	0	3

COMPUTER GRAPHICS

Course Objectives:

- ✓ To develop, design and implement two and three dimensional graphical structures.
- ✓ To enable students to acquire knowledge Multimedia compression and animations.
- ✓ To learn Creation, Management and Transmission of Multimedia objects

Course Outcomes:

- ✓ Illustrate the basics of computer graphics, different graphics systems and applications of computer graphics with various algorithms for line, circle and ellipse drawing objects for 2D transformations.
- ✓ Illustrate able to create the general software architecture of programs that use 3D object sets with computer graphics
- ✓ Apply projections and visible surface detection techniques for display of 3D scene on 2D screen

UNIT I

Introduction to Graphics: Application areas of Computer Graphics, overview of graphics systems, video-display devices, graphics monitors and work stations and input devices. 2D Primitives: Output primitives-Line, Circle and Ellipse drawing algorithms, Attributes of output primitives, Two dimensional Geometric transformations, Two dimensional viewing Line, Polygon, Curve and Text clipping algorithms.

UNIT II

3D Concepts: Parallel and Perspective projections, Three dimensional object representation- Polygons, Curved lines, Splines, Quadric Surfaces, Visualization of data sets, 3D transformations, Viewing, Visible surface identification.

UNIT III

Graphics Programming: Color Models- RGB, YIQ, CMY, HSV, Animations -General Computer Animation, Raster, Keyframe. Graphics programming using OPENGL-Basic graphics primitives, Drawing three dimensional objects, Drawing three dimensional scenes.

UNIT IV

Rendering: Introduction to shading models, Flat and Smooth shading, Adding texture to faces, Adding shadows of objects, Building a camera in a program, Creating shaded objects

UNIT V

Overview of Ray Tracing: Intersecting rays with other primitives, Adding Surface texture, Reflections and Transparency, Boolean operations on Objects.

Text Books:

1) Donald Hearn, Pauline Baker, Computer Graphics – C Version, second edition, Pearson Education,2004.
2) Schaum's Outline of Computer Graphics Second Edition, Zhigang Xiang, Roy A. Plastock.

Reference Books:

1) James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics- Principles and practice, Second Edition in C, Pearson Education, 2007.
2) F.S. Hill, Computer Graphics using OPENGL, Second edition, Pearson Education, 2003.

IV Year - I Semester	Code: 20CS5D03	L	T	P	C
		3	0	0	3

SOFTWARE ENGINEERING

Course Objectives:

This course is designed to:

- ✓ Give exposure to phases of Software Development, common process models including Waterfall, and the Unified Process, and hands-on experience with elements of the agile process
- ✓ Give exposure to a variety of Software Engineering practices such as requirements analysis and specification, code analysis, code debugging, testing, traceability, and version control
- ✓ Give exposure to Software Design techniques

Course Outcomes:

Students taking this subject will gain software engineering skills in the following areas:

- ✓ Ability to transform an Object-Oriented Design into high quality, executable code
- ✓ Skills to design, implement, and execute test cases at the Unit and Integration level
- ✓ Compare conventional and agile software methods

UNIT I

The Nature of Software, The Unique Nature of WebApps, Software Engineering, The Software Process, Software Engineering Practice, Software Myths. A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology.

UNIT II

Agility, Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, A Tool Set for the Agile Process, Software Engineering Knowledge, Core Principles, Principles That Guide Each Framework Activity, Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

UNIT III

Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling, Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements Modelling, Requirements Modeling for WebApps.

UNIT IV

Design within the Context of Software Engineering, The Design Process, Design Concepts, The Design Model, Software Architecture, Architectural Genres, Architectural Styles, Assessing Alternative Architectural Designs, Architectural Mapping Using Data Flow, Components, Designing Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, Designing Traditional Components, Component-Based Development.

UNIT V

The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation, Elements of Software Qualtiy Assurance, SQA Tasks, Goals & Metrics, Statistical SQA, Software Reliability, A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Test Strategies for WebApps, Validation Testing, System Testing, The Art of Debugging, Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing.

Text Books:

1) Software Engineering a practitioner's approach, Roger S. Pressman, Seventh Edition, McGraw Hill Higher Education.
2) Software Engineering, Ian Sommerville, Ninth Edition, Pearson.

Reference Books:

1) Software Engineering, A Precise Approach, PankajJalote, Wiley India, 2010.
2) Software Engineering, Ugrasen Suman, Cengage.

III Year - I Semester	Code: 20AD5D01	L	T	P	C
		3	0	0	3

OBJECT ORIENTED ANALYSIS AND DESIGN

Course Objectives:

1. To understand how to solve complex problems.
2. Analyze and design solutions to problems using object oriented approach.
3. Study the notations of Unified Modeling Language

Course Outcomes:

1. Ability to find solutions to the complex problems using object oriented approach.
2.Represent classes, responsibilities and states using UML notation
3.Identify classes and responsibilities of the problem domain

UNIT I

Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems, Evolution of Object Model, Foundation of Object Model, Elements of Object Model, Applying the Object Model.

UNIT II

Classes and Objects: Nature of object, Relationships among objects, Nature of a Class, Relationship among Classes, Interplay of Classes and Objects, Identifying Classes and Objects, Importance of Proper Classification, Identifying Classes and Objects, Key abstractions and Mechanisms.

UNIT III

Introduction to UML: Why we model, Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class diagrams, Object diagrams.

UNIT IV

Basic Behavioral Modeling: Interactions, Interaction diagrams, Use cases, Use case Diagrams, Activity Diagrams.

UNIT V

Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

Text Books:

1. “Object- Oriented Analysis And Design with Applications”, Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, 3rd edition, 2013, PEARSON.
2. “The Unified Modeling Language User Guide”, Grady Booch, James Rumbaugh, Ivar Jacobson, 12th Impression, 2012, PEARSON.

Reference Books:

1. “Object-oriented analysis and design using UML”, Mahesh P. Matha, PHI
2. “Head first object-oriented analysis and design”, Brett D. McLaughlin, Gary Pollice, Dave West, O‟Reilly

III Year - I Semester	Code: 20AD5D02	L	T	P	C
		3	0	0	3

Dev Ops

Course Objectives:

This course is designed to:

- ➔ Adapt the software Engineering practices that combine Software Development and IT operations for Quality Software
- ➔ Enumerate the principles of continuous development and deployment, automation of configuration management, inter-team collaboration, and IT service agility

Course Outcomes:

1. Illustrate the Phases of Software Development life cycle (L2)
2. Appraise the values and principles of agile software development (L5)
3. Create the Instance of applications
4. Interpret the aspects in user's context
5. Demonstrate the Continuous Integration, Delivery and Deployment

UNIT I

Phases of Software Development life cycle. Values and principles of agile software development

UNIT II

Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.

UNIT III

DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes

UNIT IV

CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment , Benefits of CI/CD, Metrics to track CICD practices

UNIT V

Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment

Text Books:

1.The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations, Gene Kim , John Willis , Patrick Debois , Jez Humb,1st Edition, O'Reilly publications, 2016.
2. What is Devops? Infrastructure as code, 1st Edition, Mike Loukides ,O'Reilly publications, 2012.

Reference Books:

1) Building a DevOps Culture, 1st Edition, Mandi Walls, O'Reilly publications, 2013.
2) Achieving DevOps: A Novel About Delivering the Best of Agile, DevOps, and microservices, 1st Edition, Dave Harrison, Knox Lively, Apress publications, 2019

III Year - I Semester	Code : 20CS5L12	L	T	P	C
		0	0	3	1.5

DBMS LAB

Course Objectives:

1. Populate and query a database using SQL DDL/DML Commands
2. Declare and enforce integrity constraints on a database
3. Writing Queries using advanced concepts of SQL
4. Programming PL/SQL including procedures, functions, cursors and triggers

Course Outcomes:

1. Utilize SQL to execute queries for creating database and performing data manipulation operations
2. Apply Queries using SQL
3. Build PL/SQL programs including stored procedures, functions, cursors and triggers

DBMS PROGRAMS

1 Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.

2 Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.

3 Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

4 Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)

5 i. Create a simple PL/SQL program which includes declaration section, executable section and exception – Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found) ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.

6 Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.

7 Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, User defined Exceptions, RAISE APPLICATION ERROR.

8 Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.

9 Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.

10 Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.

11 Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers

		0	0	3	1.5

MACHINE LEARNING LAB

Course Objectives:

This course will enable students to learn and understand different Data sets in implementing the machine learning algorithms.

Course Outcomes:

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

- Implement procedures for the machine learning algorithms
- Design and Develop Python programs for various Learning algorithms
- Apply appropriate data sets to the Machine Learning algorithms
- Develop Machine Learning algorithms to solve real world problems

Experiments:

Experiment-1:

Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.

Experiment-2:

For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.

Experiment-3:

Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.

Experiment-4:

Exercises to solve the real-world problems using the following machine learning methods: a) Linear Regression b) Logistic Regression c) Binary Classifier

Experiment-5: Develop a program for Bias, Variance, Remove duplicates , Cross Validation

Experiment-6: Write a program to implement Categorical Encoding, One-hot Encoding

Experiment-7:

Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.

Experiment-8:

Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions.

Experiment-9: Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Experiment-10:

Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

Experiment-11: Apply EM algorithm to cluster a Heart Disease Data Set. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.

Experiment-12: Exploratory Data Analysis for Classification using Pandas or Matplotlib.

Experiment-13:

Write a Python program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set

Experiment-14:

Write a program to Implement Support Vector Machines and Principle Component Analysis

Experiment-15:

Write a program to Implement Principle Component Analysis

Text Books:

1. Machine Learning – Tom M. Mitchell, MGH
2. Fundamentals of Speech Recognition By Lawrence Rabiner and Biing – Hwang Juang.

Reference Books:

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

III Year - I Semester	Code : 20AD5S01	L	T	P	C
		1	0	2	2

Skill Advanced Course: Continuous Integration and Continuous Delivery using DevOps

Course Outcomes:

At the end of the Course, Student will be able to:

- Understand the why, what and how of DevOps adoption
- Attain literacy on Devops
- Align capabilities required in the team
- Create an automated CICD pipeline using a stack of tools

Exercise 1:

Reference course name :Software engineering and Agile software development

Get an understanding of the stages in software development lifecycle, the process models, values and principles of agility and the need for agile software development. This will enable you to work in projects following an agile approach to software development.

Solve the questions given in the reference course name to gauge your understanding of the topic

Exercise 2:

Reference course name: Development & Testing with Agile: Extreme Programming

Get a working knowledge of using extreme automation through XP programming practices of test first development, refactoring and automating test case writing.

Solve the questions in the "Take test" module given in the reference course name to gauge your understanding of the topic

Exercise 3:

Module name : DevOps adoption in projects

It is important to comprehend the need to automate the software development lifecycle stages through DevOps. Gain an understanding of the capabilities required to implement DevOps, continuous integration and continuous delivery practices.

Solve the questions given in Quiz1, Quiz2, Quiz 3

Exercise 4:

Module name :Implementation of CICD with Java and open source stack

Configure the web application and Version control using Git using Git commands and version control operations.

Exercise 5:

Module Name: Implementation of CICD with Java and open source stack

Configure a static code analyzer which will perform static analysis of the web application code and identify the coding practices that are not appropriate. Configure the profiles and dashboard of the static code analysis tool.

Exercise 6:

Module Name: Implementation of CICD with Java and open source stack

Write a build script to build the application using a build automation tool like Maven. Create a folder structure that will run the build script and invoke the various software development build stages. This script should invoke the static analysis tool and unit test cases and deploy the application to a web application server like Tomcat.

Exercise 7:

Module Name: Implementation of CICD with Java and open source stack

Configure the Jenkins tool with the required paths, path variables, users and pipeline views.

Exercise 8:

Module name: Implementation of CICD with Java and open source stack

Configure the Jenkins pipeline to call the build script jobs and configure to run it whenever there is a change made to an application in the version control system. Make a change to the background color of the landing page of the web application and check if the configured pipeline runs.

Exercise 9:

Module name: Implementation of CICD with Java and open source stack

Create a pipeline view of the Jenkins pipeline used in Exercise 8. Configure it with user defined messages.

Exercise 10 :

Module name: Implementation of CICD with Java and open source stack
In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for static analysis of code.
Exercise 11:
Module name :Implementation of CICD with Java and open source stack
In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for static unit testing.
Exercise 12:
Module name :Course end assessment
In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for code coverage.
Text Books: 1. Learning Continuous Integration with Jenkins: A beginner's guide to implementing Continuous Integration and Continuous Delivery using Jenkins - Nikhil Pathania ,Packt publication **[https://www.amazon.in/Learning-Continuous-Integration-Jenkins-Pathania/dp/1785284835]** 2. Jenkins 2 – Up and Running: Evolve Your Deployment Pipeline for Next Generation Automation - Brent Laster, O'Reilly publication **[https://www.amazon.in/Jenkins-2-Running-Brent-Laster/dp/ 1491979593]**
Web Links: (Courses mapped to Infosys Springboard platform) 1.https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_013382690411003904735_shared/overview [Software Engineering and Agile software development] 2.https://infyspringboard.onwingspan.com/en/viewer/html/lex_auth_013501578194976768104 67 [Development & Testing with Agile: Extreme Programming] 3.https://infyspringboard.onwingspan.com/en/viewer/html/lex_auth_0135389891719249922 6_shared [DevOps CICD]

[AM VENKATA CHALAMAYYA ENGINERING COLLEGE
ODALAREVU – 533 210, Andhra Pradesh, India

III Year – II SEMESTER

DEPARTMENT OF COMPUTER SCIENCE AND

BR20 Syllabus for AIML, BVCEC w.e.f 2020-21

ENGINEERING
DETAILED SYLLABUS FOR REGULATIONS – BR20
UG: B.Tech - *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*

III Year – II SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20CS6T12	Compiler Design	3	0	0	3
2	20CS6T13	Data Mining and Data Warehousing	3	0	0	3
3	20CS6T14	Deep Learning	3	0	0	3
4	20AD6D04 20AD6D05 20CS6D07 20AD6D06	**Professional Elective -II** 1. Design and Analysis of Algorithms 2. Software Project Management 3.Distributed Systems 4. Data Wrangling in Data Science/**MOOCs-2**	3	0	0	3
5	20CS6E03 20ME6E02 20CE6E02 20EC6E02	**Open Elective- II** (Inter Disciplinary) **5.** Mean Stack Development **6.** Operations Research **7.** Green Building **8.** Internet of Things	2	0	2	3
6	20CS6L13	Compiler Design Lab	0	0	3	1.5
7	20CS6L14	Data Mining Lab	0	0	3	1.5
8	20AD6L04	Deep Learning with Tensorflow Lab	0	0	3	1.5
9	20CS6S04	Skill Advanced course: Game Development	1	0	2	2
10	20HS6M05	Universal Human Values	2	0	0	0
		Total	17	0	13	21.5

III YEAR II SEM	Code: 20CS6T12	L	T	P	C
		3	0	0	3

COMPILER DESIGN

Course Objectives:
Understand the basic concept of compiler design, and its different phases which will be helpful to construct new tools like LEX, YACC, etc.

Course Outcomes: .
At the end of the course, student will be able to

- Demonstrate phases in the design of compiler
- Organize Syntax Analysis, Top Down and LL(1) grammars
- Design Bottom Up Parsing and Construction of LR parsers

UNIT I
Lexical Analysis: Language Processors, Structure of a Compiler, Lexical Analysis, The Role of the Lexical Analyzer, Bootstrapping, Input Buffering, Specification of Tokens, Recognition of Tokens, Lexical Analyzer Generator-LEX, Finite Automata, Regular Expressions and Finite Automata, Design of a Lexical Analyzer Generator.

UNIT II
Syntax Analysis: The Role of the Parser, Context-Free Grammars, Derivations, Parse Trees, Ambiguity, Left Recursion, Left Factoring, **Top Down Parsing:** Pre Processing Steps of Top Down Parsing, Backtracking, Recursive Descent Parsing, LL (1) Grammars, Non-recursive Predictive Parsing, Error Recovery in Predictive Parsing.

UNIT III
Bottom Up Parsing: Introduction, Difference between LR and LL Parsers, Types of LR Parsers, Shift Reduce Parsing, SLR Parsers, Construction of SLR Parsing Tables, More Powerful LR Parses, Construction of CLR (1) and LALR Parsing Tables, Dangling Else Ambiguity, Error Recovery in LR Parsing, Handling Ambiguity Grammar with LR Parsers.

UNIT IV
Syntax Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's. **Intermediate Code Generation:** Variants of Syntax Trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow, Backpatching, Intermediate Code for Procedures.

UNIT V
Run Time Environments: Storage Organization, Run Time Storage Allocation, Activation Records, Procedure Calls, Displays, **Code Optimization:** The Principle Sources of Optimization, Basic Blocks, Optimization of Basic Blocks, Structure Preserving Transformations, Flow Graphs, Loop Optimization, Data-Flow Analysis, Peephole Optimization
Code Generation: Issues in the Design of a Code Generator, Object Code Forms, Code Generation Algorithm, Register Allocation and Assignment.

Text Books:
1) Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson.
2) Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning.

Reference Books:
1) Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
2) The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH
3) Writing compilers and interpreters, R. Mak, 3rd edition, Wiley student edition.

e-Resources:
1) https://nptel.ac.in/courses/106/104/106104123/

II YEAR II SEM		L	T	P	C

	Subject Code: 20CS6T13	3	0	0	3

DATA MINING AND DATA WAREHOUSING

Course Objectives:

To Understand about Data Warehouse and Data Mining to extract knowledge from data repository for data analysis, frequent pattern, classification , Clustering and prediction.

Course Outcomes:

- Understand Importance of extraction of Knowledge from huge data and Data mining functionalities.
- Understand the data pre-processing techniques aggregation, sampling, dimensionality reduction.
- Understand multi-dimensional data model like OLAP and Architecture of Data Warehouse

UNIT I

Introduction: What Motivated Data Mining? Why Is It Important, Data Mining—On What Kind of Data, Data Mining Functionalities—What Kinds of Patterns can Be Mined? Are All of the Patterns Interesting?, Data Mining Task Primitives, Major Issues in Data Mining.**(Han & Kamber)**

UNIT II

Data Pre-processing: Why Pre-process the Data? Descriptive Data Summarization, Data Cleaning, Data Integration and Transformation, Data Reduction, Data Discretization and Concept Hierarchy Generation, **Data Warehouse and OLAP Technology: An Overview:**

What Is a Data Warehouse? A Multidimensional Data Model, Data Warehouse Architecture, **(Han & Kamber)**

UNIT III

Association Analysis: Basic Concepts and Algorithms: Introduction, Frequent Item Set generation, Rule generation, compact representation of frequent item sets, FP-Growth Algorithm.**(Tan & Vipin)**

UNIT IV

Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Working of Decision Tree, building a decision tree, methods for expressing an attribute test conditions, measures for selecting the best split, Algorithm for decision tree induction. Classification: Alterative Techniques, Bayes' Theorem, Naïve Bayesian Classification, Bayesian Belief Networks

UNIT V

Cluster Analysis: Basic Concepts and Algorithms: What Is Cluster Analysis? Different Types of Clustering, Different Types of Clusters, K-means, The Basic K-means Algorithm, K-means: Additional Issues, Bisecting K-means, K-means and Different Types of Clusters, Strengths and Weaknesses, K-means as an Optimization Problem, Agglomerative Hierarchical clustering, Basic Agglomerative Hierarchical clustering Algorithm, Specific Techniques, DBSCAN, Traditional Density: center-Based Approach, The DBSCAN Algorithm, Strengths and Weaknesses.**(Tan &Vipin)**

Text Books:

1. Introduction to Data Mining: Pang- Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.
2. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier.

Reference Books:

1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning.
2. Data Mining :Introductory and Advanced topics: Dunham, Pearson.

III YEAR II SEM	Subject Code: 20AD6T04	L	T	P	C
		3	0	0	3

DEEP LAERNING

Course Objectives:

The main objective of the course is to make students:

- Learn deep learning methods for working with sequential data,
- Learn deep recurrent and memory networks,
- Learn deep Turing machines

Course Outcomes:

After the completion of the course, student will be able to

- Demonstrate the fundamental concepts learning techniques of Artificial Intelligence, Machine Learning and Deep Learning.
- Discuss the Neural Network training, various random models**.**
- Explain the Techniques of Keras, TensorFlow, Theano and CNTK

UNIT I:

Fundamentals of Deep Learning: Artificial Intelligence, History of Machine learning: Probabilistic Modeling, Early Neural Networks, Kernel Methods, Decision Trees, Random forests and Gradient Boosting Machines, **Fundamentals of Machine Learning:** Four Branches of Machine Learning, Evaluating Machine learning Models, Overfitting and Underfitting. **[Text Book 2]**

UNIT II: Introducing Deep Learning: Biological and Machine Vision, Human and Machine Language, Artificial Neural Networks, Training Deep Networks, Improving Deep Networks. **[Text Book3]**

UNIT III: Neural Networks: Anatomy of Neural Network, Introduction to Keras: Keras, TensorFlow, Theano and CNTK, Setting up Deep Learning Workstation, Classifying Movie Reviews: Binary Classification, Classifying newswires: Multiclass Classification. **[Text Book 2]**

Convolutional Neural Networks: Nerual Network and Representation Learing, Convolutional Layers, Multichannel Convolution Operation, **Recurrent Neural Networks:** Introduction to RNN, RNN Code, PyTorch Tensors: Deep Learning with PyTorch, CNN in PyTorch. **[Text Book 3]**

Interactive Applications of Deep Learning: Machine Vision, Natural Language processing, Generative Adversial Networks, Deep Reinforcement Learning. **[Text Book 1]**

Deep Learning Research: Autoencoders**,** Deep Generative Models: Boltzmann Machines Restricted Boltzmann Machines, Deep Belief Networks. **[Text Book 1]**

Text Books:

Deep Learning- Ian Goodfellow, Yoshua Bengio and Aaron Courvile, MIT Press, 2016

Deep Learning with Python - Francois Chollet, Released December 2017, Publisher(s): Manning Publications, ISBN: 9781617294433

Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence - Jon Krohn, Grant Beyleveld, Aglaé Bassens, Released September 2019, Publisher(s): Addison-Wesley Professional, ISBN: 9780135116821

Reference Books:

Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.

Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.

Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-Hill Education, 2004.

	Subject Code: 20AD6D04	3	0	0	3

DESIGN AND ANALYSIS OF ALGORITHMS

Course Objectives:

Upon completion of this course, students will be able to do the following:

- Ability to understand, analyze and denote time complexities of algorithms
- To introduce the different algorithmic approaches for problem solving through numerous example problems

Course Outcomes:

After the completion of the course, student will be able to

- Analyze the performance of a given algorithm, denote its time complexity using the asymptotic notation for recursive and non-recursive algorithms
- List and describe various algorithmic approaches and Solve problems using divide and conquer &greedy Method
- Synthesize efficient algorithms dynamic programming approaches

UNIT I

Introduction: Algorithm, Pseudo code for expressing algorithms, performance Analysis-Space complexity, Time complexity, Asymptotic Notation-Big oh notation, Omega notation, Theta notation, small oh notation and Little oh notation, probabilistic analysis.

UNIT II

Divide and conquer: General method, applications-Binary search, Quick sort, Merge sort, Defective Chess board

UNIT III

Greedy method: General method, applications-Job sequencing with deadlines, knapsack problem, spanning trees, Minimum cost spanning trees, Single source shortest path problem.

UNIT IV

Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search tree, String Edition Problem, 0/1 knapsack problem, All pairs shortest path problem, *Optimal* Merge Pattern,

UNIT V

Backtracking: General method, applications-n-queen problem, sum of sub sets problem, graph coloring, Hamiltonian cycles.

Text Books:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
2.Design and Analysis of Algorithms , S Sridhar, Oxford
3. Design and Analysis of Algorithms, Parag Himanshu Dave, Himansu Balachandra Dave, 2nd ed, Pearson Education.

Reference Books:

1. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.
2.Introduction to Algorithms, second edition, T. H. Cormen, C.E. Leiserson, R. L. Rivest and. Stein, PHI Pvt. Ltd.

III YEAR II SEM	Subject Code: 20AD6D05	L	T	P	C
		3	0	0	3

SOFTWARE PROJECT MANAGEMENT

Course Objectives:

At the end of the course, the student shall be able to:

- To describe and determine the purpose and importance of project management from the perspectives of planning, tracking and completion of project
- To compare and differentiate organization structures and project structures

Course Outcomes:

Upon the completion of the course students will be able to:-

- Apply the process to be followed in the software development life-cycle models.
- Apply the concepts of project management & planning.
- Implement the project plans through managing people, communications and change.

UNIT I

Conventional Software Management: The waterfall model, conventional software Management performance. Evolution of Software Economics: Software Economics, pragmatic software cost estimation. Improving Software Economics: Reducing Software product size, improving software processes, improving team effectiveness, improving automation, Achieving required quality, peer inspections.

UNIT II

The Old Way and The New: The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process. Life Cycle Phases: Engineering and production stages, inception, Elaboration, construction, transition phases. Artifacts of The Process: The artifact sets, Management artifacts, Engineering artifacts, programmatic artifacts.

UNIT III

Model Based Software Architectures: A Management perspective and technical perspective.

Work Flows of the Process: Software process workflows, Iteration workflows. Checkpoints of the Process: Major mile stones, Minor Milestones, Periodic status assessments.

UNIT IV

Iterative Process Planning: Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning. Project Organizations and Responsibilities: Line-of-Business Organizations, Project Organizations, evolution of Organizations.

UNIT V

Process Automation: Automation Building blocks, The Project Environment. Project Control and Process Instrumentation: The seven core Metrics, Management indicators, quality indicators, life cycle expectations, pragmatic Software Metrics, Metrics automation. Project Estimation and Management: COCOMO model, Critical Path Analysis, PERT technique, Monte Carlo approach (Text book 2)

Text Books:

1) Software Project Management, Walker Royce, Pearson Education, 2005.
2) Software Project Management, Bob Hughes, 4th edition, Mike Cotterell, TMH.

Reference Books:

1) Software Project Management, Joel Henry, Pearson Education.
2) Software Project Management in practice, Pankaj Jalote, Pearson Education, 2005.
3) Effective Software Project Management, Robert K.Wysocki, Wiley,2006.

III Year - II Semester	Code: 20CS6D07	L	T	P	C
		3	0	0	3

DISTRIBUTED SYSTEMS

Course Outcomes:

- Develop a familiarity with distributed file systems.
- Describe important characteristics of distributed systems and the salient architectural features of such systems.
- Describe the features and applications of important standard protocols which are used in distributed systems.
- Gaining practical experience of inter-process communication in a distributed environment

UNIT I

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

UNIT II

Interprocess Communication: Introduction, The API for the Internet Protocols- The Characteristics of Interprocess communication, Sockets, UDP Datagram Communication, TCP Stream Communication; External Data Representation and Marshalling; Client Server Communication; Group Communication- IP Multicast- an implementation of group communication, Reliability and Ordering of Multicast.

UNIT III

Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects- Object Model, Distributed Object Modal, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI

UNIT IV

Operating System Support: Introduction, The Operating System Layer, Protection, Processes and Threads – Address Space, Creation of a New Process, Threads.

UNIT V

Transactions & Replications: Introduction, System Model and Group Communication, Concurrency Control in Distributed Transactions, Distributed Dead Locks, Transaction Recovery; Replication-Introduction, Passive (Primary) Replication, Active Replication.

Text Books:

1. Ajay D Kshemkalyani, MukeshSighal, “Distributed Computing, Principles, Algorithms and Systems”, Cambridge.
2. George Coulouris, Jean Dollimore, Tim Kindberg, “Distributed Systems- Concepts and Design”, Fourth Edition, Pearson Publication

Reference Books:

1.Distributed-Systems-Principles-Paradigms-Tanenbaum PHI

III Year - II Semester	Code: 20AD6D06	L	T	P	C
		3	0	0	3

DATA WRANGLING IN DATA SCIENCE

Course Outcomes:

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

- Identify and execute the basic data format.
- Perform the computations with Excel and pdf files
- Understand the concepts of data cleanup
- Explore and analyze the Image and video data
- Understand the concepts web scraping

UNIT I

INTRODUCTION TO DATA WRANGLING: Data Wrangling- Importance of Data Wrangling -How is Data Wrangling performed- Tasks of Data Wrangling-Data Wrangling Tools-Introduction to Python-Python Basics-Data Meant to be Read by Machines-CSV Data-JSON Data-XML Data

UNIT II

WORKING WITH EXCEL FILES AND PDFS: Installing Python Packages-Parsing Excel Files-Parsing Excel Files -Getting Started with Parsing-PDFs and Problem Solving in Python-Programmatic Approaches to PDF Parsing-Converting PDF to Text-Parsing PDFs Using pdf miner-Acquiring and Storing Data-Databases: A Brief Introduction-Relational Databases: MySQL and PostgreSQL-Non-Relational Databases: NoSQL-When to Use a Simple File-Alternative Data Storage

UNIT III

DATA CLEANUP: Why Clean Data?- Data Cleanup Basics-Identifying Values for Data Cleanup-Formatting Data-Finding Outliers and Bad Data-Finding Duplicates-Fuzzy Matching-RegEx Matching-Normalizing and Standardizing the Data-Saving the Data-Determining suitable Data Cleanup-Scripting the CleanupTesting with New Data

UNIT IV

DATA EXPLORATION AND ANALYSIS: Exploring Data-Importing Data-Exploring Table Functions-Joining Numerous Datasets-Identifying Correlations-Identifying Outliers-Creating Groupings-Analyzing Data-Separating and Focusing the DataPresenting Data-Visualizing the Data-Charts-Time-Related Data-Maps-Interactives-Words-Images, Video, and Illustrations-Presentation Tools-Publishing the Data-Open Source Platforms

UNIT V

WEB SCRAPING: What to Scrape and How-Analyzing a Web Page-Network/Timeline-Interacting with JavaScript-In-Depth Analysis of a Page-Getting Pages-Reading a Web Page-Reading a Web Page with LXML-XPath-Advanced Web Scraping-Browser-Based Parsing-Screen Reading with Selenium-Screen Reading with Ghost.PySpidering the Web-Building a Spider with Scrapy-Crawling Whole Websites with Scrapy.

Text Books:

1. Data Wrangling with Python, Jacqueline Kazil& Katharine Jarmul, O'Reilly Media, Inc,2016
2. Data Wrangling with Python: Creating actionable data from raw sources,,Dr. TirthajyotiSarkar, ShubhadeepPackt Publishing Ltd,2019

Reference Books:

1. Hands-On Data Analysis with Pandas, Stefanie Molin,Packt Publishing Ltd,2019
2. Practical Data Wrangling, Allan Visochek, Packt Publishing Ltd,2017
3. Principles of Data Wrangling: Practical Techniques for Data Preparation, TyeRattenbury, Joseph M. Hellerstein, Jeffrey Heer, Sean Kandel, Connor Carreras, , O'Reilly Media, Inc,2017

III Year - II Semester	Code: 20CS6E03	L	T	P	C
		3	0	0	3

MEAN STACK DEVELOPMENT

Course Outcomes:

After the completion of the course, student will be able to

- Enumerate the Basic Concepts of Web & Markup Languages
- Develop web Applications using Scripting Languages & Frameworks
- Make use of Express JS and Node JS frameworks
- Illustrate the uses of web services concepts like restful, react js
- Apply Deployment Techniques & Working with cloud platform

UNIT I

Introduction to Web: Internet and World Wide Web, Domain name service, Protocols: HTTP, FTP, SMTP. Html5 concepts, CSS3, Anatomy of a web page. XML: Document type Definition, XML schemas, Document object model, XSLT, DOM and SAX Approaches.

UNIT II

JavaScript: The Basic of JavaScript: Objects, Primitives Operations and Expressions, Control Statements, Arrays, Functions, Constructors, Pattern Matching using Regular Expressions. Angular Java Script Angular JS Expressions: ARRAY, Objects, $eval, Strings, Angular JS Form Validation & Form Submission, Single Page Application development using Angular JS.

UNIT III

Node.js: Introduction, Advantages, Node.js Process Model, Node JS Modules. Express.js: Introduction to Express Framework, Introduction to Nodejs , What is Nodejs, Getting Started with Express, Your first Express App, Express Routing, Implementing MVC in Express, Middleware, Using Template Engines, Error Handling , API Handling , Debugging, Developing Template Engines, Using Process Managers, Security & Deployment.

UNIT IV

RESTful Web Services: Using the Uniform Interface, Designing URIs,
Web Linking, Conditional Requests. React Js: Welcome to React, Obstacles and Roadblocks, React's Future, Keeping Up with the Changes, Working with the Files, Pure React, Page Setup, The Virtual DOM, React Elements, ReactDOM, Children, Constructing Elements with Data, React Components, DOM Rendering, Factories.

UNIT V

Mongo DB: Introduction, Architecture, Features, Examples, Database Creation & Collection in Mongo DB. Deploying Applications: Web hosting & Domains, Deployment Using Cloud Platforms.

Text Books:

1) Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2) Web Technologies, Uttam K Roy, Oxford
3) Pro Mean Stack Development, ELadElrom, Apress
4) Restful Web Services Cookbook, Subbu Allamraju, O'Reilly
5) JavaScript & jQuery the missing manual, David sawyer mcfarland, O'Reilly
6) Web Hosting for Dummies, Peter Pollock, John Wiley Brand

Reference Books:

1) Ruby on Rails up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly (2006).
2) Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012).
3) Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
4) An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning.

III Year – II Semester	Code: 20CS6L13	L	T	P	C
		0	0	3	1.5

COMPILER DESIGN LAB

Course Objectives:
To enlighten the student with knowledge base in compiler design and its applications

Course Outcomes: The end of the course student will be able to

- Design simple lexical analyzers
- Determine predictive parsing table for a CFG
- Apply Lex and Yacc tools
- Examine LR parser and generating SLR Parsing table
- Relate Intermediate code generation for subset C language

Experiments:

1. Write a C program to identify different types of Tokens in a given Program.
2. Write a Lex Program to implement a Lexical Analyzer using Lex tool.
3. Write a C program to Simulate Lexical Analyzer to validating a given input String.
4. Write a C program to implement the Brute force technique of Top down Parsing.
5. Write a C program to implement a Recursive Descent Parser.
6. Write C program to compute the *First* and *Follow* Sets for the given Grammar.
7. Write a C program for eliminating the left recursion and left factoring of a given grammar
8. Write a C program to check the validity of input string using Predictive Parser.
9. Write a C program for implementation of LR parsing algorithm to accept a given input string.
10. Write a C program for implementation of a Shift Reduce Parser using Stack Data Structure to accept a given input string of a given grammar.
11. Write a C program for generating the three address code of a given expression/statement.
12. Write a C program for implementation of a Code Generation Algorithm of a given expression/statement.

Text Books & Reference Books :

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson Publishers, 2007.
2. John R Levine, Tony Mason, Doug Brown, "Lex and Yacc", Orielly, 2nd Edition, 2009.

III YEAR II SEM	Code: 20CS6L14	L	T	P	C
		0	0	3	1.5

DATA MINING LAB

Course Objectives:

The main objective of the course is to

- Inculcate Conceptual, Logical, and Physical design of Data Warehouses OLAP applications and OLAP deployment
- Design a data warehouse or data mart to present information needed by management in a form that is usable
- Emphasize hands-on experience working with all real data sets.
- Test real data sets using popular data mining tools such as WEKA, Python Libraries
- Develop ability to design various algorithms based on data mining tools.

Course Outcomes:

By the end of the course student will be able to

- Design a data mart or data warehouse for any organization
- Extract knowledge using data mining techniques and enlist various algorithms used in information analysis of Data Mining Techniques
- Demonstrate the working of algorithms for data mining tasks such as association rule mining, classification for realistic data
- Implement and Analyze on knowledge flow application on data sets and Apply the suitable visualization techniques to output analytical results

List of Experiments:

1. Creation of a Data Warehouse.
   - Build Data Warehouse/Data Mart (using open source tools like Pentaho Data Integration Tool, Pentaho Business Analytics; or other data warehouse tools like Microsoft-SSIS, Informatica, Business Objects,etc.,)
   - Design multi-dimensional data models namely Star, Snowflake and Fact Constellation schemas for any one enterprise (ex. Banking, Insurance, Finance, Healthcare, manufacturing, Automobiles, sales etc).
   - Write ETL scripts and implement using data warehouse tools.
   - Perform Various OLAP operations such slice, dice, roll up, drill up and pivot
2. Explore machine learning tool "WEKA"
   - Explore WEKA Data Mining/Machine Learning Toolkit.
   - Downloading and/or installation of WEKA data mining toolkit.
   - Understand the features of WEKA toolkit such as Explorer, Knowledge Flow interface, Experimenter, command-line interface.
   - Navigate the options available in the WEKA (ex. Select attributes panel, Preprocess panel, Classify panel, Cluster panel, Associate panel and Visualize panel)
   - Study the arff file format Explore the available data sets in WEKA. Load a data set (ex. Weather dataset, Iris dataset, etc.)
   - Load each dataset and observe the following:
     1. List the attribute names and they types
     2. Number of records in each dataset
     3. Identify the class attribute (if any)
     4. Plot Histogram
     5. Determine the number of records for each class.
     6. Visualize the data in various dimensions
3. Perform data preprocessing tasks and Demonstrate performing association rule mining on data sets
   - Explore various options available in Weka for preprocessing data and apply Unsupervised filters like Discretization, Resample filter, etc. on each dataset
   - Load weather. nominal, Iris, Glass datasets into Weka and run Apriori Algorithm with different support and confidence values.
   - Study the rules generated. Apply different discretization filters on numerical attributes and run the Apriori association rule algorithm. Study the rules generated.
   - Derive interesting insights and observe the effect of discretization in the rule generation process.

4. Demonstrate performing classification on data sets
   - Load each dataset into Weka and run 1d3, J48 classification algorithm. Study the classifier output. Compute entropy values, Kappa statistic.
   - Extract if-then rules from the decision tree generated by the classifier, Observe the confusion matrix.
   - Load each dataset into Weka and perform Naïve-bayes classification and k-Nearest Neighbour classification. Interpret the results obtained.
   - Plot RoC Curves
   - Compare classification results of ID3, J48, Naïve-Bayes and k-NN classifiers for each dataset, and deduce which classifier is performing best and poor for each dataset and justify.
5. Demonstrate performing clustering of data sets
   - Load each dataset into Weka and run simple k-means clustering algorithm with different values of k (number of desired clusters).
   - Study the clusters formed. Observe the sum of squared errors and centroids, and derive insights.
   - Explore other clustering techniques available in Weka.
   - Explore visualization features of Weka to visualize the clusters. Derive interesting insights and explain.
6. Demonstrate knowledge flow application on data sets
   - Develop a knowledge flow layout for finding strong association rules by using Apriori, FP Growth algorithms
   - Set up the knowledge flow to load an ARFF (batch mode) and perform a cross validation using J48 algorithm
   - Demonstrate plotting multiple ROC curves in the same plot window by using j48 and Random forest tree
7. Demonstrate ZeroR technique on Iris dataset (by using necessary preprocessing technique(s)) and share your observations
8. Write a java program to prepare a simulated data set with unique instances.
9. Write a Python program to generate frequent item sets / association rules using Apriori algorithm
10. Write a program to calculate chi-square value using Python. Report your observation.
11. Write a program of Naive Bayesian classification using Python programming language.
12. Implement a Java program to perform Apriori algorithm
13. Write a program to cluster your choice of data using simple k-means algorithm using JDK
14. Write a program of cluster analysis using simple k-means algorithm Python programming language.
15. Write a program to compute/display dissimilarity matrix (for your own dataset containing at least four instances with two attributes) using Python
16. Visualize the datasets using matplotlib in python.(Histogram, Box plot, Bar chart, Pie chart etc.,)

III YEAR II SEM	Code: 20AD6L04	L	T	P	C
		0	0	3	1.5

DEEP LEARNING WITH TENSORFLOW LAB

Course Outcomes:

On completion of this course, the student will be able to

- Implement deep neural networks to solve real world problems
- Choose appropriate pre-trained model to solve real time problem
- Interpret the results of two different deep learning models

List of Experiments:

1. Implement multilayer perceptron algorithm for MNIST Hand written Digit Classification.
2. Design a neural network for classifying movie reviews (Binary Classification) using IMDB dataset.
3. Design a neural Network for classifying news wires (Multi class classification) using Reuters dataset.
4. Design a neural network for predicting house prices using Boston Housing Price dataset.
5. Build a Convolution Neural Network for MNIST Hand written Digit Classification.
6. Build a Convolution Neural Network for simple image (dogs and Cats) Classification
7. Use a pre-trained convolution neural network (VGG16) for image classification.
8. Implement one hot encoding of words or characters.
9. Implement word embeddings for IMDB dataset.
10. Implement a Recurrent Neural Network for IMDB movie review classification problem.

Text Books:

1. Reza Zadeh and BharathRamsundar, “Tensorflow for Deep Learning”, O’Reilly publishers, 2018

References:

1. https://github.com/fchollet/deep-learning-with-python-notebooks

III YEAR II SEM	Code: 20CS6S04	L	T	P	C
		1	0	2	2
Skill Advanced course: Game Development					
1	Mini – Game Project				
2	Coin Collection Game				
3	Adventure Game				
4	Car Race Game				
5	Puzzle Game				
Software's					
1	Unity 5.6.7f1				
2	Blender 2.79				
3	Krita				
4	World Machine				

ENGINEERING
DETAILED SYLLABUS FOR REGULATIONS – BR20
UG: B.Tech - *ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING*
IV Year – I SEMESTER

S.No	Course Code	Courses	L	T	P	Credits
1	20CS7D09 20AD7D07 20AD7D08 20CS7D18	**Professional Elective-III** 1. Cryptography and Network Security 2 Reinforcement Learning 3. Social Media Analytics 4. Block Chain Technologies	3	0	0	3
2	20CS7D13 20CS7D16 20CS7D11 20AD7D09	**Professional Elective-IV** 5. Software Testing Methodologies 6. Computer Vision 7. Adhoc and Sensor Networks 8. NOSQL Databases	3	0	0	3
3	20AD7D10 20AD7D11 20AD7D12 20AD7D13	**Professional Elective-V** 1. Social Network Analysis 2. Recommender Systems 3. AI Chatbots 4. Data Visualization	3	0	0	3
4	20CS7E04 20CE7E03 20EC7E03 20EE7E03	**Open Elective -III** 5. Nature Inspired Computing Techniques 6. Disaster Management 7. Embedded Systems 8. Electrical Distribution Systems	2	0	2	3
5	20CS7E05 20EC7E04 20ME7E04 20CE7E04	**Open Elective- IV** 5. Secure Coding Techniques 6. 5G Communications 7. 3D Printing Technologies 8. Project Management	2	0	2	3
6	20HS7E01	**Humanity and Social Science Elective** 1. Resource Management 2. Industrial Management 3. Management Science 4. IPR and Patents	3	0	0	3
7	20HS7S01	English for Employability	1	0	2	2
8	20CS7P02	Industrial/Research Internship 2 months after III_I Evaluated during IV_I	0	0	6	3
		Total	17	0	12	23

		3	0	0	3

CRYPTOGRAPHY AND NETWORK SECURITY

Course Outcomes:

By the end of the course the student

- ✓ Identify information security goals, classical encryption techniques
- ✓ Compare and apply different encryption and decryption techniques
- ✓ Apply the knowledge of cryptographic checksums and evaluate the performance of different message digest algorithms
- ✓ Apply different digital signature algorithms to achieve authentication
- ✓ Apply network security basics, analyze different attacks on networks and evaluate the performance of firewalls and security protocols like SSL, IPSec, and PGP

UNIT I

Introduction: Elements of Information security, Security attacks, services & mechanisms, A model of network security, Symmetric Cipher Model, Substitution Techniques, Transportation Techniques, Steganography

UNIT II

Block Ciphers & Symmetric Key Cryptography

Traditional Block Cipher Structure, Block Cipher Design Principles, Data Encryption Standard, AES-Structure, Transformation functions, Key Expansion.

UNIT III

Number Theory: Prime and Relatively Prime Numbers, Modular Arithmetic, Fermat's and Euler's Theorems, The Chinese Remainder theorem.

Public Key Cryptography: Principles, public key cryptography algorithms, RSA Algorithms, Diffie Hellman Key Exchange, Elgamal encryption & decryption.

UNIT IV

Cryptographic Hash Functions & Digital Signatures

Application of Cryptographic hash Functions, Requirements & Security, Secure Hash Algorithm, Message Authentication Functions, Requirements & Security, and HMAC & CMAC. Digital Signatures, NIST Digital Signature Algorithm. Key management & distribution.

UNIT V

Transport Level Security: Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Shell (SSH), **Electronic Mail Security:** Pretty Good Privacy (PGP) and S/MIME.**P Security:** IP Security Overview, IP Security Architecture.

Text Books:

1. Cryptography & Network Security: Principles and Practices, William Stallings, PEA, Sixth edition.
2. Introduction to Computer Networks & Cyber Security, Chwan Hwa Wu, J.David Irwin, CRC press

Reference Books:

1. Everyday Cryptography, Fundamental Principles & Applications, Keith Martin, Oxford
2. Network Security & Cryptography, Bernard Menezes, Cengage, 2010

IV Year - I Semester	Code: 20AD7D07	L	T	P	C
		3	0	0	3

REINFORCEMENT LEARNING

Course Objectives:

- Learn various approaches to solve decision problems with functional models and algorithms for task formulation, Tabular based solutions, Function approximation solutions, policy gradients and model based reinforcement learning.

Course Outcomes:

By completing the course the students will be able to:

- Understand basic concepts of Reinforcement learning
- Identifying appropriate learning tasks for Reinforcement learning techniques
- Understand various methods and applications of reinforcement learning

UNIT I

Introduction: Reinforcement Learning, Examples, Elements of Reinforcement Learning, Limitations and Scope, An Extended Example: Tic-Tac-Toe

Multi-armed Bandits: A k-armed Bandit Problem, Action-value methods, The 10-armed Testbed, Incremental Implementation, Tracking a Nonstationary Problem, Optimistic Initial Values, Upper –Confidence-Bound Action Selection, Gradient Bandit Algorithm

UNIT II

Finite Markov Decision Process: The Agent-Environment Interface, Goals and Rewards, Returns and Episodes, Unified Notataion for Episodic and Continuing Tasks, Policies and Value Functions,

Dynamic Programming: Policy Evaluation, Policy Improvement, Policy Iteration, Value Iteration, Asynchronous Dynamic Programming, Generalized Policy Iteration, Efficiency of Dynamic Programming

UNIT III

Monte Carlo Methods: Monte Carlo Prediction, Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Discontinuing-aware Importance Sampling, Per-decision Importance Sampling

UNIT IV

Off-policy Methods with Approximation: Semi-gradient Methods, Examples of Off-policy Divergence, The Deadly Triad, Linear Value-function Geometry, Gradient Descent in the Bellman Error, The Bellman Error is not Learnable, Gradient-TD methods, Emphatic-TD methods, Reducing Variance

UNIT V

Policy Gradient Methods: Policy Approximation and its Advantages, The Policy Gradient Theorem, REINFOECE: Monte Carlo Policy Gradient, REINFORCE with Baseline, Actor-Critic Methods, Policy Gradient for Continuing Problems, Policy Parameterization fr Continuous Actions

Applications and Case Studies:TD-Gammon, Samuel's Checkers Player, Watson's Daily Double Wagering, Optimizing Memory Control, Personalized Web Services

Text Books:

1. R. S. Sutton and A. G. Bart,. "Reinforcement Learning - An Introduction," MIT Press, 2018.

Reference Books:

1. Szepesvári, Csaba, "Algorithms for Reinforcement Learning," United States: Morgan & Claypool, 2010.
2. Puterman, Martin L., "Markov Decision Processes: Discrete Stochastic Dynamic Programming," Germany: Wiley, 2014.

IV Year - I Semester	Code: 20AD7D08	L	T	P	C
		3	0	0	3

SOCIAL MEDIA ANALYTICS

Course Objectives:

- Understand and deal with any social media network, strategy, or campaign.

Course Outcomes:

By completing the course the students will be able to:

- Understand social media categories and types of social media analytics
- Understand the impact of social media analytics integration with and affects other areas of business.

UNIT I

Introduction: Foundation for Analytics, Evolution of Data and the Digital Gap, Social Media Data Sources: Offline and Online, Definition of Social Media, Data Sources in Social Media Channels, Estimated vs. Factual Data Sources, Public and Private Data, Data Gathering in Social Media Analytics, Social Media Network Support of Data Collection, API: Application Programming Interface, Web Crawling or Scraping,

UNIT II

From Data to Insights: Example of a Single Metric Giving Actionable Insight, An Example of a Metric Leading to New Questions, Creating a Plan to Shape Data into Insights, The Planning Stage: Projecting Possible Insights, Analysis of a Social Media Post, The process of Comparison, Data Aggregation, Calculations and Display, Data Display, Social Media and Big Data, Potential Challenges

UNIT III

Analytics in Social Media: Types of Analytics in Social Media, Analytics or Channel Analytics, Social Media Listening: Keyword and Mention-Based Analysis, Demographics, Interests and Sentiment, Advertising Analytics: Focus on Conversions and ROI of Paid Social Media Campaigns, Conversions: The Key to Digital and Social Advertising, CMS Analytics: Measuring the Performance of the Content Management Team, CRM Analytics: Customer Support and Sales via Social Media

UNIT IV

Dedicated vs. Hybrid Tools : Common to all Tools, Dedicated Tools, Advantages of Dedicated Tools, Disadvantages of Dedicated tools, Hybrid Tools, Dedicated Tools with Hybrid Features, Advantages of Hybrid Tools, Disadvantages of Hybrid Tools, Data Integration Tools, Advantages of Data Integration Tools, Disadvantages of Data Integration Tools.

UNIT V

Social Network Landscape: Concept and UX on Social Networks, Features and Their Strategic Value, Interactivity: How Social is the Network, Content Flow on Social Network

The Analytics Process: Analysis is Comparison, Investigation beyond Social Analytics, Shaping a Method: The End Game for an Analyst, The Analysis Circle, Dynamic Cycles.

Text Books:

1.Alex Goncalves, "Social Media Analytics Strategy-Using Data to Optimize Business Performance," Apress, 2017

Reference Books:

1. Qiu, Liangfei., Kumar, Subodha, "Social Media Analytics and Practical Applications: The Change to the Competition Landscape," United States: CRC Press, 2021.
2. Sponder, Marshall, "Social Media Analytics: Effective Tools for Building, Interpreting, and Using Metrics," United Kingdom: McGraw-Hill Education, 2011.

IV Year - I Semester	Code: 20CS7D18	L	T	P	C
		3	0	0	3

BLOCKCHAIN TECHNOLOGIES

Course Objectives:

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

- Understand how block chain systems (mainly Bit coin and Ethereum) work and to securely interact with them,
- Design, build, and deploy smart contracts and distributed applications,
- Integrate ideas from block chain technology into their own projects.

Course Outcomes:

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

- Demonstrate the foundation of the Block chain technology and understand the processes in payment and funding.
- Identify the risks involved in building Block chain applications.
- Review of legal implications using smart contracts.
- Choose the present landscape of Blockchain implementations and Understand Crypto currency markets
- Examine how to profit from trading crypto currencies.

UNIT I

Introduction, Scenarios, Challenges Articulated, Blockchain, Blockchain Characteristics, Opportunities Using Blockchain, History of Blockchain. Evolution of Blockchain : Evolution of Computer Applications, Centralized Applications, Decentralized Applications, Stages in Blockchain Evolution, Consortia, Forks, Public Blockchain Environments, Type of Players in Blockchain Ecosystem, Players in Market.

UNIT II

Blockchain Concepts: Introduction, Changing of Blocks, Hashing, Merkle-Tree, Consensus, Mining and Finalizing Blocks, Currency aka tokens, security on blockchain, data storage on blockchain, wallets, coding on blockchain: smart contracts, peer-to-peer network, types of blockchain nodes, risk associated with blockchain solutions, life cycle of blockchain transaction.

UNIT III

Architecting Blockchain solutions: Introduction, Obstacles for Use of Blockchain, Blockchain Relevance Evaluation Framework, Blockchain Solutions Reference Architecture, Types of Blockchain Applications.
Cryptographic Tokens, Typical Solution Architecture for Enterprise Use Cases, Types of Blockchain Solutions, Architecture Considerations, Architecture with Blockchain Platforms, Approach for Designing Blockchain Applications.

UNIT IV

Ethereum Blockchain Implementation: Introduction, Tuna Fish Tracking Use Case, Ethereum Ecosystem, Ethereum Development, Ethereum Tool Stack, Ethereum Virtual Machine, Smart Contract Programming, Integrated Development Environment, Truffle Framework, Ganache, Unit Testing, Ethereum Accounts, MyEtherWallet, Ethereum Networks/Environments, Infura, Etherscan, Ethereum Clients, Decentralized Application, Metamask, Tuna Fish Use Case Implementation, OpenZeppelin Contracts

UNIT V

Hyperledger Blockchain Implementation, Introduction, Use Case – Car Ownership Tracking, Hyperledger Fabric, Hyperledger Fabric Transaction Flow, FabCar Use Case Implementation, Invoking Chaincode Functions Using Client Application.

Text Books:

1) Ambadas, Arshad Sarfarz Ariff, Sham "Blockchain for Enterprise Application Developers", Wiley
2) Andreas M. Antonpoulos, "Mastering Bitcoin: Programming the Open Blockchain" , O'Reilly

Reference Books:

1) Blockchain: A Practical Guide to Developing Business, Law, and Technology Solutions, Joseph Bambara, Paul R. Allen, Mc Graw Hill.
2) Blockchain: Blueprint for a New Economy, Melanie Swan, O'Reilly

IV Year - I Semester	Code : 20CS7D13	L	T	P	C
		3	0	0	3

SOFTWARE TESTING METHODOLOGIES

Course Objectives:

1. To study the fundamental concepts of software testing which includes objectives, process, criteria, strategies, and methods.
2. To discuss various software testing types and levels of testing like black and white box testing along with levels unit test, integration, regression, and system testing.

Course Outcomes:

1. Know the basic concepts of software testing and its essentials.
2. Able to identify the various bugs and correcting them after knowing the consequences of the bug.
3. Use of program's control flow as a structural model is the corner stone of testing.
4. Performing functional testing using control flow and transaction flow graphs.

UNIT I

Introduction:-Purpose of testing, Dichotomies,model for testing, consequences of bugs, taxonomy of bugs, Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

UNIT II

Transaction Flow Testing:-transaction flows, transaction flow testing techniques. Dataflow testing:- Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

UNIT III

Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

UNIT IV

Paths, Path products and Regular expressions:- path products &path expression, reduction procedure, applications, regular expressions & flow anomaly detection. Logic Based Testing:-overview, decision tables,path expressions,kv charts, specifications.

UNIT V

State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips. Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools

Text Books:

1. Software Testing techniques – Boris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

Reference Books:

1. The craft of software testing – Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)

IV Year - I Semester	Code: 20CS7D16	L	T	P	C
		3	0	0	3

COMPUTER VISION

Course Objectives:

To introduce students the fundamentals of image formation; To introduce students the major ideas, methods, and techniques of computer vision and pattern recognition; To develop an appreciation for various issues in the design of computer vision and object recognition systems;

Course Outcomes:

After completing the course Student will be able to:

1. Identify basic concepts, terminology, theories, models and methods in the field of computer vision,
2. Describe known principles of feature detection and matching,
3. Describe basic methods of computer vision related to image stitching, photography like high dynamic range imaging and blur removal.

UNIT I

Introduction: Image Formation: Geometric Primitives and Transformation, Photometric Image Formation, Digital Camera, Image Processing: Point Operators, Linear Filtering, More Neighborhood Operators, Fourier Transforms, Pyramids and Wavelets, Geometric Transformations, Global Optimization.

UNIT II

Feature Detection and Matching: Points and Patches, Edges, Lines, Segmentation: Active Contours, Split and Merge, Mean Shift and Mode Finding, Normalized Cuts, Feature-Based Alignment: 2D and 3D Feature-based Alignment, Pose Estimation, Geometric Intrinsic Calibration.

UNIT III

Structure and Motion: Triangular, Two-frame Structure from Motion, Factorization, Bundle Adjustment, Constrained Structure and Motion, Dense Motion Estimation: Translation Alignment, Parametric Motion, Spline-based Motion, Optical Flow, Layered motion

UNIT IV

Image Stitching: Motion Models, Global Alignment, Composing, Computational Photography: Photometric Calibration, High Dynamic Range Imaging, Super-Resolution and Blur Removal, image Matting and Compositing, Texture Analysis and Synthesis.

UNIT V

3D Reconstruction: Shape From X, Active Range Finding, Surface Representation, Point-based Representation, Volumetric Representation, Model-based Reconstruction, Recovering Texture Maps and Albedos, Image- based Rendering: View Interpolation, Layered Depth Images, Light Fields and Lumigraphs, Environment Mattes, Video-based Rendering.

Text Books:

1. Richard Szeliski, Computer Vision: Algorithms and Applications, Springer-Verlag London Limited, 2011.
2. Simon J.D Prince, Computer Vision: Models, Learning and Inference, 1st Edition, 2012.

Reference Books:

1. Computer Vision - A modern approach, by D. Forsyth and J. Ponce, Prentice Hall Robot Vision, by B. K. P. Horn, McGraw-Hill.
2. Haralick& Shapiro, “Computer and Robot Vision”, Vol II
3. G_erardMedioni and Sing Bing Kang “Emerging topics in computer vision”100

IV Year - I Semester	Code: 20CS7D11	L	T	P	C
		3	0	0	3

ADHOC AND SENSOR NETWORKS

Course Outcomes:

By the end of the course the student

- Able to think and develop new applications in Manets and WSN.
- Able to take any new technical issue related to these new thrust areas and come up with a solution(s).
- Able to develop algorithms/protocols for Manets and WSN.
- Understand the issues and solutions of various layers of Manets, namely MAC layer, Network Layer & Transport Layer in Manets and WSN.
- Understand the issues in *Security and Sensor Network and Tools.*

UNIT I

Introduction to Ad Hoc Networks: Characteristics of MANETs, applications of MANETs, and challenges of MANETs. Routing in MANETs: Criteria for classification, Taxonomy of MANET routing algorithms, Topology based routing algorithms, Position based routing algorithms,

UNIT II

Data Transmission: Broadcast storm problem, Broadcasting, Multicasting and Geo casting

UNIT III

TCP over Ad Hoc: TCP protocol overview, TCP and MANETs, and Solutions for TCP over Ad hoc networks.

UNIT IV

Basics of Wireless Sensors and Applications: Applications, Classification of sensor networks, Architecture of sensor networks, Physical layer, MAC layer, Link layer.

UNIT V

Security: Security in ad hoc networks, Key management, Secure routing, Cooperation in MANETs, and Intrusion detection systems.

Text Books:

1. Ad hoc and Sensor Networks - Theory and Applications, by Carlos Cordeiro and Dharma P. Agrawal, World Scientific Publications, March 2006, ISBN 981-256-681-3.
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science ISBN: 978-1-55860-914-3, (Morgan Kauffman)

Reference Books:

1. AD HOC AND SENSOR NETWORKS: THEORY AND APPLICATIONS 2ND EDITION, KINDLE EDITION

IV Year - I Semester	Code: 20AD7D09	L	T	P	C
		3	0	0	3

NoSQL DATABASES

Course Objectives:

From the course the student will

- To understand the relational database design principles
- To become familiar with the basic issues of transaction processing and concurrency control
- To become familiar with database storage structures and access techniques

Course Outcomes:

After the completion of the course, student will be able to do the following

- Identify what type of NoSQL database to implement based on business requirements (key-value, document, full text, graph, etc.)
- Apply NoSQL data modeling from application specific queries
- Use Atomic Aggregates and denormalization as data modelling techniques to optimize query processing

UNIT I

Introduction to NoSQL: Definition And Introduction, Sorted Ordered Column-Oriented Stores, Key/Value Stores, Document Databases, Graph Databases, Examining Two Simple Examples, Location Preferences Store, Car Make And Model Database, Working With Language Bindings.

UNIT II

Interacting with NoSQL: If NoSql Then What, Language Bindings For NoSQL Data Stores, Performing Crud Operations, Creating Records, Accessing Data, Updating And Deleting Data.

UNIT III

NoSQL Storage Architecture: Working With Column-Oriented Databases, Hbase Distributed Storage Architecture, Document Store Internals, Understanding Key/Value Stores In Memcached And Redis, Eventually Consistent Non-Relational Databases.

UNIT IV

NoSQL Stores: Similarities Between Sql And Mongodb Query Features, Accessing Data From Column-Oriented Databases Like Hbase, Querying Redis Data Stores, Changing Document Databases, Schema Evolution In Column-Oriented Databases, Hbase Data Import And Export, Data Evolution In Key/Value Stores.

UNIT V

Indexing and Ordering Data Sets : Essential Concepts Behind A Database Index, Indexing And Ordering In Mongodb, Creating and Using Indexes In Mongodb, Indexing And Ordering In Couchdb, Indexing In Apache Cassandra.

Text Books:

1) Pramod Sadalage and Martin Fowler, NoSQL Distilled, Addison-Wesley Professional, 2012.
2) Dan McCreary and Ann Kelly, Making Sense of NoSQL, Manning Publications, 2013.

Reference Books:

1) Shashank Tiwari, Professional NoSQL, Wrox Press, Wiley, 2011, ISBN: 978-0-470-94224-6
2) Gaurav Vaish, Getting Started with NoSQL, Packt Publishing, 2013.

IV Year - I Semester	Code: 20AD7D10	L	T	P	C
		3	0	0	3

SOCIAL NETWORK ANALYSIS

Course Outcomes:

After completing the course student should:

- Know basic notation and terminology used in network science
- Be able to visualize, summarize and compare networks
- Illustrate basic principles behind network analysis algorithms
- Develop practical skills of network analysis in R programming language
- Be capable of analyzing real work networks

UNIT I

Social Network Analysis: Preliminaries and definitions, Erdos Number Project, Centrality measures, Balance and Homophily.

UNIT II

Random graph models**:** Random graphs and alternative models, Models of network growth, Navigationin social Networks, Cohesive subgroups, Multidimensional Scaling, Structural equivalence, roles and positions.

UNIT III

Network topology and diffusion, Contagion in Networks, Complex contagion, Perco Percolation and information, Navigation in Networks Revisited

UNIT IV

Small world experiments, small world models, origins of small world, Heavy tails,SmallDiameter,Clusteringofconnectivity,TheErdosRenyiModel,ClusteringModels

UNIT V

Network structure -Important vertices and page rank algorithm, towards rational dynamics in networks, basics of game theory, Coloring and consensus, biased voting, network formation games, networkstructureandequilibrium,behavioralexperiments,Spatialandagent-basedmodels

Text Books:

1. S.Wasserman and K.Faust.“Social Network Analysis:Methods and Applications”, Cambridge University Press
2.D.Easley and J.Kleinberg,“Networks,Crowds and Markets:Reasoningabou tahighly connected world” ,Cambridge UniversityPress,1st edition,2010

Reference Books:

1.Maarten van Steen.“Graph Theory and Complex Networks.An Introduction”,2010
2.Reza Zafarani,MohammedAliAbbasi,HuanLiu.“SocialMediaMining:AnIntroduction”.Cambridge UniversityPress 2014
3.MaksimTsvetovat and Alexander Kouznetsov. “Social Network Analysis for Startups”. O’ReillyMedia,2011

IV Year - I Semester	Code: 20AD7D11	L	T	P	C
		3	0	0	3

RECOMMENDER SYSTEMS

Course Objectives:

To develop state-of-the-art recommender systems that automates a variety of choice-making strategies with the goal of providing affordable, personal, and high-quality recommendations

Course Outcomes:

By completing the course the students will be able to:

- Understand the basic concepts of recommender systems
- Carry out performance evaluation of recommender systems based on various metrics
- Implement machine-learning and data-mining algorithms in recommender systems data sets.
- Design and implement a simple recommender system.

UNIT I

An Introduction to Recommender Systems: Goals of Recommender Systems, Basic Models of Recommender Systems, Collaborative Filtering Models, Content-Based Recommender Systems, Knowledge-Based Recommender Systems, Domain-Specific Challenges in Recommender Systems, Advanced Topics and Applications.

UNIT II

Neighborhood-Based Collaborative Filtering: Key Properties of Ratings Matrices, Predicting Ratings with Neighborhood-Based Methods, Clustering and Neighborhood-Based Methods, Dimensionality Reduction and Neighborhood Methods, A Regression Modeling View of Neighborhood Methods, Graph Models for Neighborhood-Based Methods

UNIT III

Model-Based Collaborative Filtering: Decision and Regression Trees, Rule-Based Collaborative Filtering, Naïve Bayes Collaborative Filtering, Latent Factor Models, Integrating Factorization and Neighborhood Models

UNIT IV

Content-Based Recommender Systems: Basic Components of Content-Based Systems, Preprocessing and Feature Extraction, Learning User Profiles and Filtering, Content-Based Versus Collaborative Recommendations
Knowledge-Based Recommender Systems: Constraint-Based Recommender Systems, Case-Based Recommenders, Persistent Personalization in Knowledge-Based Systems.

UNIT V

Evaluating Recommender Systems: Evaluation Paradigms, General Goals of Evaluation Design, Design Issues in Offline Recommender Evaluation, Accuracy Metrics in Offline Evaluation, Limitations of Evaluation Measures

Text Books:

1. Charu .C. Aggarwal, Recommender Systems: The Textbook, Springer, 2016.

Reference Books:

1. Jannach D., Zanker M. and FelFering A., Recommender Systems: An Introduction, Cambridge University Press(2011), 1st ed.
2. Ricci F., Rokach L., Shapira D., Kantor B.P., Recommender Systems Handbook, Springer(2011), 1st ed.
3. Manouselis N., Drachsler H., Verbert K., Duval E., Recommender Systems For Learning, Springer (2013), 1st ed.
4. J. Leskovec, A. Rajaraman and J. Ullman, Mining of massive datasets, 2nd Ed., Cambridge, 2012

IV Year - I Semester	Code: 20AD7D12	L	T	P	C
		3	0	0	3

AI CHATBOTS

Course Objectives:

- Learn how artificial intelligence powers chatbots, get an overview of the bot ecosystem and bot anatomy, and study different types of bots and use cases.
- Identify best practices for defining a chatbot use case, and use a rapid prototyping framework to develop a use case for a personalized chatbot.

Course Outcomes:

- Develop an in-depth understanding of conversation design, including on boarding, flows, utterances, entities, and personality.
- Design, build, test, and iterate a fully-functional, interactive chatbot using a commercial platform.

Deploy the finished chatbot for public use and interaction

UNIT I

Introduction: Benefits from Chatbots for a Business, A Customer-Centric Approach in Financial Services, Chatbots in the Insurance Industry, Conversational Chatbot Landscape,
Identifying the Sources of Data: Chatbot Conversations, Training Chatbots for Conversations, Personal Data in Chatbots, Introduction to the General Data Protection Regulation (GDPR)

UNIT II

Chatbot Development Essentials: Customer Service-Centric Chatbots, Chatbot Development Approaches, Rules-Based Approach, AI-Based Approach, Conversational Flow, Key Terms in Chatbots, Utterance, Intent, Entity, Channel, Human Takeover, Use Case: 24x7 Insurance Agent

UNIT III

Building a Chatbot Solution: Business Considerations, Chatbots Vs Apps, Growth of Messenger Applications, Direct Contact Vs Chat, Business Benefits of Chatbots, Success Metrics, Customer Satisfaction Index, Completion Rate, Bounce Rate, Managing Risks in Chatbots Service, Generic Solution Architecture for Private Chatbots

UNIT IV

Natural Language Processing, Understanding, and Generation: Chatbot Architecture, Popular Open Source NLP and NLU Tools, Natural Language Processing, Natural Language Understanding, Natural Language Generation, Applications

UNIT V

Introduction to Microsoft Bot, RASA, and Google Dialog flow: Microsoft Bot Framework, Introduction to QnA Maker, Introduction to LUIS, Introduction to RASA, RASA Core, RASA NLU, Introduction to Dialog flow
Chatbot Integration Mechanism: Integration with Third-Party APIs, Connecting to an Enterprise Data Store, Integration Module

Text Books:

1. Abhishek Singh, KarthikRamasubramanian, Shrey Shivam, “Building an Enterprise Chatbot: Work with Protected Enterprise Data Using Open Source Frameworks”, ISBN 978-1-4842-5034-1, Apress,2019

Reference Books:

1. Janarthanam and Srini, Hands-on chatbots and conversational UI development: Build chatbots and voice user interfaces with C (1 ed.), Packt Publishing Ltd, 2017. ISBN 978-1788294669.
2. Galitsky, Boris., Developing Enterprise Chatbots (1 ed.), Springer International Publishing, 2019. ISBN 978-303004298

IV Year - I Semester	Code: 20AD7D13	L	T	P	C
		3	0	0	3

DATA VISUALIZATION

Course Outcomes:

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

- Understand basics of Data Visualization
- Implement visualization of distributions

UNIT I

INTRODUCTION TO VISUALIZATION: Visualizing Data-Mapping Data onto Aesthetics, Aesthetics and Types of Data, Scales Map Data Values onto Aesthetics, Coordinate Systems and Axes- Cartesian Coordinates, Nonlinear Axes, Coordinate Systems with Curved Axes, Color Scales-Color as a Tool to Distinguish, Color to Represent Data Values, Color as a Tool to Highlight, Directory of Visualizations- Amounts, Distributions, Proportions, x–y relationships, Geospatial Data

UNIT II

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 Once-Visualizing Distributions Along the VerticalAxis, Visualizing Distributions Along the Horizontal Axis

UNIT III

VISUALIZING ASSOCIATIONS & TIME SERIES: Visualizing Proportions-A Case for Pie Charts, A Case for Side-by-Side Bars, A Case for Stacked Bars and Stacked Densities, Visualizing Proportions Separately as Parts of the Total ,Visualizing Nested Proportions- Nested Proportions Gone Wrong, Mosaic Plots and Tree maps, Nested Pies ,Parallel Sets. Visualizing Associations Among Two or More Quantitative Variables-Scatter plots, Correlograms, Dimension Reduction, Paired Data. Visualizing Time Series and Other Functions of an Independent Variable-Individual Time Series , Multiple Time Series and Dose–Response Curves, Time Series of Twoor More Response Variables

UNIT IV

VISUALIZING UNCERTIANITY: Visualizing Trends-Smoothing, Showing Trends with a Defined Functional Form, Detrending and Time-Series Decomposition, Visualizing Geospatial Data-Projections, Layers, Choropleth Mapping, Cartograms, Visualizing Uncertainty-Framing Probabilities as Frequencies, Visualizing the Uncertainty of Point Estimates, Visualizing the Uncertainty of Curve Fits, Hypothetical Outcome Plots

UNIT V

PRINCIPLE OF PROPORTIONAL INK: The Principle of Proportional Ink-Visualizations Along Linear Axes, Visualizations Along Logarithmic Axes, Direct Area Visualizations, Handling Overlapping Points-Partial Transparency and Jittering, 2DHistograms, Contour Lines, Common Pitfalls of Color Use-Encoding Too Much or Irrelevant Information ,Using Non monotonic Color Scales to Encode Data Values, Not Designing for Color-Vision Deficiency

Text Books:

1. Claus Wilke, “Fundamentals of Data Visualization: A Primer on Making Informative and Compelling Figures”, 1st edition, O’Reilly Media Inc, 2019.
2. Ossama Embarak, Data Analysis and Visualization Using Python: Analyze Data to Create Visualizations for BI Systems, Apress, 2018

Reference Books:

1. Tony Fischetti, Brett Lantz, R: Data Analysis and Visualization, O’Reilly, 2016

IV Year - I Semester	Code: 20CS7E04	L	T	P	C
		3	0	0	3

NATURE INSPIRED COMPUTING TECHNIQUES

Course Objectives:

Learn the theoretical foundations of Nature Inspired Computing techniques, how they can be used to solve problems, and in which areas are most useful and effective.

Course Outcomes:

By completing the course the students will be able to:

- Understand the strengths, weaknesses and appropriateness of nature-inspired algorithms.
- Apply nature-inspired algorithms to optimization, design and learning problems.

UNIT I

Analysis of Algorithms: Analysis of Optimization Algorithms, Nature Inspired Algorithms, Parameter Tuning and Parameter Control: Parameter Tuning, Hyper optimization, Multi objective View, Parameter Control, Simulated Annealing: Algorithm, Basic Convergence Properties, Stochastic Tunneling

UNIT II

Genetic Algorithms: Introduction, Role of Genetic Operators, Choice of Parameters, GA Variants, **Differential Evolution:** Introduction, Differential Evolution, Variants, Choice of Parameters, Convergence Analysis, **Particle Swarm Optimization:** Swarm Intelligence, PSO Algorithm, Accelerated PSO, Binary PSO

UNIT III

Firefly Algorithms: Firefly Behavior, Standard Firefly Algorithm Variations of Light Intensity and Attractiveness, Controlling Randomization, Firefly Algorithms in Applications

Cuckoo Search: Cuckoo Breeding Behavior, Levy Flights, Cuckoo Search: Special Cases of Cuckoo Search, Variants of Cuckoo Search, Global Convergence, Applications

UNIT IV

Bat Algorithms: Echolocation of Bats: Behavior of Microbats, Acoustics of Echolocation, Bat Algorithms: Movement of Virtual Bats, Loudness and Pulse Emission, Binary Bat Algorithm, Variants of the Bat Algorithm, Convergence Analysis, Applications: Continuous Optimization, Combinatorial Optimization and Scheduling, Inverse Problems and Parameter Estimation, Classifications, Clustering and Data Mining, Image Processing, Fuzzy Logic and Other Applications

UNIT V

Flower Pollination Algorithms: Introduction, Characteristics of Flower Pollination, Flower Pollination Algorithms, Multi-Objective Flower Pollination Algorithms, Validation and Numerical Experiments: Single-Objective Test Functions, Multi-Objective Test Functions, Applications: Single-Objective Design Benchmarks, Multi-Objective Design Benchmarks

Text Books:

1. "Nature-Inspired Optimization Algorithms", Yang, Xin-She, Elsevier Science, 2014.

Reference Books:

1."Nature-Inspired Computing and Optimization: Theory and Applications," Germany: Springer International Publishing, 2017.

IV Year - I Semester	Code: 20CS7E05	L	T	P	C
		3	0	0	3

SECURE CODING TECHNIQUES

Course Outcomes:

At the end of the Course, student will be able to:

- Differentiate the objectives of information security
- Understand the trend, reasons and impact of the recent Cyber attacks
- Understand OWASP design principles while designing a web application

UNIT I

Network and Information security Fundamentals: Network Basics, Network Components, Network Types, Network Communication Types, Introduction to Networking Models, Cyber Security Objectives and Services, Other Terms of Cyber Security, Myths Around Cyber Security, Myths Around Cyber Security, Recent Cyber Attacks, Generic Conclusion about Attacks, Why and What is Cyber Security, Categories of Attack

UNIT II

Introduction to Cyber security: Introduction to OWASP Top 10, A1 Injection, A1 Injection Risks Root Causes and its Mitigation, A1 Injection, A2 Broken Authentication and Session Management, A7 Cross Site Scripting XSS,A3 Sensitive Data Exposure, A5 Broken Access Control, A4 XML External Entity (XEE), A6 Security Misconfiguration, A7 Missing Function Level Access Control, A8 Cross Site Request Forgery CSRF, A8 Insecure Deserialization, A9 Using Components With Known Vulnerabilities.

UNIT III

Secure coding practices and OWASP Top 10: Declarative Security, Programmatic Security, Concurrency, Configuration, Cryptography, Input and Output Sanitization, Error Handling, Input Validation, Logging and auditing, Session Management, Exception Management, Safe APIs, Type Safety, Memory Management, Tokenizing, Sandboxing, Static and dynamic testing, vulnerability scanning and penetration testing

UNIT IV

Secure coding practices in C/C++ and Java: Potential Software Risks in C/C++, Defensive coding, Preventative Planning, Clean Code, Iterative Design, Assertions, Pre Post Conditions, Low level design inspections, Unit Tests Java- Managing Denial of Service, Securing Information, Data Integrity, Accessibility and Extensibility, Securing Objects, Serialization Security

UNIT V

Secure coding in Python: Interactive Python Scripting, Python Variables, Conditionals, Loops, Functions, External Modules, File operations, Web requests

Text Books:

1. Networking Fundamentals, 2019 edition, Packt, Author: Gordon Davies
2. Principles of Information Security, Authors: Michael E. Whitman and Herbert J. Mattord, Course technology incorp.

Web Reference :

1.https://www.stealthlabs.com/blog/infographic-top-15-cybersecurity- myths-vs-reality/
2. https://microage.ca/cybersecurity-layering-approach/
3.https://www.synopsys.com/glossary/what-is-threat-modeling.html#:~:text=Threat%20modeling%20is%20a%20structured,An%20abstraction%20of%20the%20system