

# **LAKIREDDY BALI REDDY COLLEGE OF ENGINEERING (AUTONOMOUS), MYLAVARAM**

## **B.Tech. (Computer Science and Engineering) R20 Regulations (w.e.f. 2020-21)**

### **I - SEMESTER**

1. **Course Code:** 20FE01  
**Course Title:** Professional Communication-I  
**Contact Hours:** 2 (L) + 0 (T) + 0 (P)  
**Credits:** 2  
**CIE:** 30 | **SEE:** 70 | **Total:** 100
2. **Course Code:** 20FE03  
**Course Title:** Differential Equations  
**Contact Hours:** 2 (L) + 1 (T) + 0 (P)  
**Credits:** 3  
**CIE:** 30 | **SEE:** 70 | **Total:** 100
3. **Course Code:** 20FE07  
**Course Title:** Applied Physics  
**Contact Hours:** 2 (L) + 1 (T) + 0 (P)  
**Credits:** 3  
**CIE:** 30 | **SEE:** 70 | **Total:** 100
4. **Course Code:** 20CS01  
**Course Title:** Programming for Problem Solving using C  
**Contact Hours:** 3 (L) + 0 (T) + 0 (P)  
**Credits:** 3  
**CIE:** 30 | **SEE:** 70 | **Total:** 100
5. **Course Code:** 20EE02  
**Course Title:** Basic Electrical and Electronics Engineering  
**Contact Hours:** 3 (L) + 0 (T) + 0 (P)  
**Credits:** 3  
**CIE:** 30 | **SEE:** 70 | **Total:** 100
6. **Course Code:** 20FE51  
**Course Title:** Professional Communication Skills Lab  
**Contact Hours:** 0 (L) + 0 (T) + 2 (P)  
**Credits:** 1  
**CIE:** 15 | **SEE:** 35 | **Total:** 50
7. **Course Code:** 20FE54  
**Course Title:** Applied Physics Lab  
**Contact Hours:** 0 (L) + 0 (T) + 3 (P)  
**Credits:** 1.5  
**CIE:** 15 | **SEE:** 35 | **Total:** 50
8. **Course Code:** 20CS51  
**Course Title:** Programming for Problem Solving using C Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

9. **Course Code:** 20IT51

**Course Title:** IT Workshop

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

**Total Contact Hours:** 12 (L) + 2 (T) + 11 (P)

**Total Credits:** 19.5

**Total CIE:** 210 | **Total SEE:** 490 | **Grand Total:** 700

## II - SEMESTER

1. **Course Code:** 20FE02

**Course Title:** Professional Communication-II

**Contact Hours:** 2 (L) + 0 (T) + 0 (P)

**Credits:** 2

**CIE:** 30 | **SEE:** 70 | **Total:** 100

2. **Course Code:** 20FE04

**Course Title:** Linear Algebra and Transformation Techniques

**Contact Hours:** 2 (L) + 1 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

3. **Course Code:** 20FE06

**Course Title:** Engineering Chemistry

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

4. **Course Code:** 20CS05

**Course Title:** Python Programming

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

5. **Course Code:** 20CS03

**Course Title:** Data Structures

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

6. **Course Code:** 20MC01

**Course Title:** Constitution of India

**Contact Hours:** 2 (L) + 0 (T) + 0 (P)

**Credits:** 0

**CIE:** 30 | **SEE:** 70 | **Total:** 100

7. **Course Code:** 20FE53

**Course Title:** Engineering Chemistry Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

8. **Course Code:** 20CS54

**Course Title:** Python Programming Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

9. **Course Code:** 20CS53

**Course Title:** Data Structures Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

10. **Course Code:** 20CS55

**Course Title:** Shell Scripting

**Contact Hours:** 0 (L) + 0 (T) + 2 (P)

**Credits:** 1

**CIE:** 15 | **SEE:** 35 | **Total:** 50

**Total Contact Hours:** 15 (L) + 1 (T) + 11 (P)

**Total Credits:** 19.5

**Total CIE:** 240 | **Total SEE:** 560 | **Grand Total:** 800

### III - SEMESTER

1. **Course Code:** 20FE09

**Course Title:** Probability & Statistics

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

2. **Course Code:** 20CS04

**Course Title:** Discrete Mathematical Structures

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

3. **Course Code:** 20CS07

**Course Title:** Database Management Systems

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

4. **Course Code:** 20CS08

**Course Title:** Computer Organization

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

5. **Course Code:** 20CS06

**Course Title:** Software Engineering

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

6. **Course Code:** 20CS57

**Course Title:** Database Management Systems Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

7. **Course Code:** 20CS58

**Course Title:** Software Engineering Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

8. **Course Code:** 20CS59

**Course Title:** Computer Organization Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

**Total Contact Hours:** 15 (L) + 0 (T) + 12 (P)

**Total Credits:** 21.5

**Total CIE:** 240 | **Total SEE:** 560 | **Grand Total:** 800

#### IV - SEMESTER

1. **Course Code:** 20CS09

**Course Title:** Theory of Computation

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

2. **Course Code:** 20CS10

**Course Title:** Operating Systems

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

3. **Course Code:** 20CS11

**Course Title:** Computer Networks

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

4. **Course Code:** 20CS12

**Course Title:** Object-Oriented Programming through Java

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

5. **Course Code:** 20CS13

**Course Title:** Design and Analysis of Algorithms

**Contact Hours:** 3 (L) + 0 (T) + 0 (P)

**Credits:** 3

**CIE:** 30 | **SEE:** 70 | **Total:** 100

6. **Course Code:** 20CS14

**Course Title:** Computer Networks Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

7. **Course Code:** 20CS15

**Course Title:** Java Programming Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

8. **Course Code:** 20CS16

**Course Title:** Operating Systems Lab

**Contact Hours:** 0 (L) + 0 (T) + 3 (P)

**Credits:** 1.5

**CIE:** 15 | **SEE:** 35 | **Total:** 50

**Total Contact Hours:** 15 (L) + 0 (T) + 12 (P)

**Total Credits:** 22.5

**Total CIE:** 240 | **Total SEE:** 560 | **Grand Total:** 800

## V - SEMESTER

1. **Course Code:** 20CS12

**Course Title:** Computer Networks

**Contact Hours/Week:** 3

**Credits:** 3

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

2. **Course Code:** 20AD04

**Course Title:** Machine Learning

**Contact Hours/Week:** 3

**Credits:** 3

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

3. **Course Code:** 20CS13

**Course Title:** Theory of Computation

**Contact Hours/Week:** 3

**Credits:** 3

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

4. **Program Elective – I** (Choose one)

- 20CS14 Computer Graphics
  - 20CS15 Principles of Programming Languages
  - 20CS16 Principles of Artificial Intelligence
- Contact Hours/Week:** 3

**Credits: 3**

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

**5. Open Elective – I**

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

**Laboratory Courses**

**6. Course Code: 20CS60**

**Course Title:** Computer Networks Lab

**Contact Hours/Week: 0**

**Credits: 1.5**

**Scheme of Valuation:** CIE: 15, SEE: 35, Total: 50

**7. Course Code: 20AD54**

**Course Title:** Machine Learning Lab

**Contact Hours/Week: 0**

**Credits: 1.5**

**Scheme of Valuation:** CIE: 15, SEE: 35, Total: 50

**8. Course Code: 20CSS3**

**Course Title:** Mean Stack Technologies

**Contact Hours/Week: 1**

**Credits: 2**

**Scheme of Valuation:** CIE: 0, SEE: 50, Total: 50

**9. Course Code: 20PI01**

**Course Title:** Summer Internship

**Contact Hours/Week: 0**

**Credits: 1.5**

**Scheme of Valuation:** CIE: 0, SEE: 50, Total: 50

**TOTAL:** 16 Contact Hours, 10 Credits, CIE: 180, SEE: 520, Total: 700

**VI - SEMESTER**

**1. Course Code: 20CS17**

**Course Title:** Information Security

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

**2. Course Code: 20CS18**

**Course Title:** Compiler Design

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

**3. Course Code: 20CS19**

**Course Title:** Bigdata Analytics

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation:** CIE: 30, SEE: 70, Total: 100

**4. Program Elective – II (Choose one)**

- 20CS20 Pattern Recognition
- 20CS21 Information Retrieval Systems
- 20CS22 Distributed Operating Systems

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

**5. Open Elective – II**

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

**Laboratory Courses**

**6. Course Code: 20CS61**

**Course Title: Information Security Lab**

**Contact Hours/Week: 0**

**Credits: 1.5**

**Scheme of Valuation: CIE: 15, SEE: 35, Total: 50**

**7. Course Code: 20CS62**

**Course Title: Data Analytics and Visualization Lab**

**Contact Hours/Week: 0**

**Credits: 1.5**

**Scheme of Valuation: CIE: 15, SEE: 35, Total: 50**

**8. Course Code: 20CS63**

**Course Title: Server-Side Scripting Lab**

**Contact Hours/Week: 0**

**Credits: 1.5**

**Scheme of Valuation: CIE: 15, SEE: 35, Total: 50**

**9. Course Code: 20HSS1**

**Course Title: Soft Skills Course**

**Contact Hours/Week: 1**

**Credits: 2**

**Scheme of Valuation: CIE: 0, SEE: 50, Total: 50**

**TOTAL: 16 Contact Hours, 11 Credits, CIE: 195, SEE: 505, Total: 700**

**VII - SEMESTER**

**1. Program Elective - III (Choose one)**

- 20CS23 Software Testing Methodologies
- 20CS24 Cloud Computing
- 20AD07 Deep Learning

**Contact Hours/Week: 3**

**Credits: 3**

**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

**2. Program Elective - IV (Choose one)**

- 20CS25 Software Project Management

- 20AD09 Natural Language Processing
- 20CS26 Soft Computing  
**Contact Hours/Week: 3**  
**Credits: 3**  
**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

3. **Program Elective - V (Choose one)**

- 20CS27 Social Network Analysis
- 20CS28 Computer Vision
- 20CS29 Block Chain Technology  
**Contact Hours/Week: 3**  
**Credits: 3**  
**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

4. **Open Elective – III**

**Contact Hours/Week: 3**  
**Credits: 3**  
**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

5. **Open Elective – IV**

**Contact Hours/Week: 3**  
**Credits: 3**  
**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

6. **Course Code: 20HS02**

**Course Title: Management Science for Engineers**  
**Contact Hours/Week: 3**  
**Credits: 3**  
**Scheme of Valuation: CIE: 30, SEE: 70, Total: 100**

**Laboratory Courses**

7. **Course Code: 20PI02**

**Course Title: Industrial/Research Internship**  
**Contact Hours/Week: 0**  
**Credits: 3**  
**Scheme of Valuation: CIE: 0, SEE: 50, Total: 50**

8. **Course Code: 20CSS4**

**Course Title: Continuous Integration and Delivery using DevOps**  
**Contact Hours/Week: 1**  
**Credits: 2**  
**Scheme of Valuation: CIE: 50, SEE: 50, Total: 100**

**TOTAL: 19 Contact Hours, 2 Credits, CIE: 180, SEE: 520, Total: 700**

**VIII - SEMESTER**

1. **Course Code: 20PI03**

**Course Title: Project Work**  
**Contact Hours/Week: 0**  
**Credits: 12**  
**Scheme of Valuation: CIE: 60, SEE: 140, Total: 200**



L	T	P	Cr.
2	0	0	2

**Pre-requisites:** Nil

**Course Educational Objectives:** To improve English language proficiency of the students in various aspects like vocabulary, grammar, communication skills, listening skills, reading & writing skills.

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

- CO1** : Write sentences and paragraphs using proper grammatical structures and word forms (**Remember – L1**)
- CO2** : Comprehend the given text by employing suitable strategies for skimming and scanning and draw inferences (**Understand – L2**)
- CO3** : Write summaries of reading texts using correct tense forms & appropriate structures (**Remember – L1**)
- CO4** : Write Formal Letters, Memos & E-Mails (**Apply – L3**)
- CO5** : Edit the sentences/short texts by identifying basic errors of grammar/vocabulary/syntax (**Understand – L2**)

### Unit - I

**Exploration** - 'A Proposal to Girdle the Earth – Nellie Bly'; Reading: Skimming for main idea; Scanning for specific information; Grammar & Vocabulary: Content Words; Function Words; Word Forms: verbs, nouns, adjectives and adverbs; Nouns: Countable and Uncountable, Singular and Plural forms; Wh - Questions; Word Order in Sentences; Writing: Paragraph Analysis; Paragraph Writing; Punctuation and Capital Letters

### Unit – II

**On Campus-** 'The District School as it Was by One Who Went to it – Warren Burton'; Reading: Identifying Sequence of Ideas; Grammar & Vocabulary: Cohesive Devices: Linkers/Signposts/Transition signals, Synonyms, Meanings of Words/Phrases in the context; Writing: Memo Drafting.

### Unit – III

**Working Together-** 'The Future of Work'

Reading: Making basic inferences; Strategies to use text clues for comprehension; Summarizing; Grammar & Vocabulary: Verbs: Tenses; Reporting Verbs for Academic Purpose; Writing: Rephrasing what is read; Avoiding redundancies and repetitions; Abstract Writing/ Summarizing.

### Unit – IV

**'A.P.J.Abdul Kalam'**; Grammar & Vocabulary: Direct & Indirect Speech; Articles and their Omission; Writing: E-Mail Drafting.

### Unit – V

**'C.V.Raman'**; Grammar & Vocabulary: Subject-Verb Agreement; Prepositions; Writing: Formal Letter Writing.

**Text Books:**

1. Prabhavati. Y & et al, “English All Round – Communication Skills for Undergraduate Learners”, Orient BlackSwan, Hyderabad, 2019.
2. “Panorama – A Course on Reading”, A collection of prose selections, Oxford University Press, New Delhi, 2016.

**Reference Books:**

1. Swan, M., “Practical English Usage”, Oxford University Press, 2016.
2. Kumar,S and Latha, P, “Communication Skills”, Oxford University Press, 2018.
3. Rizvi Ashraf M., “Effective Technical Communication”, Tata Mc Graw Hill, New Delhi, 2008.
4. Baradwaj Kumkum, “Professional Communication”, I.K. International Publishing House Pvt. Ltd., New Delhi, 2008.  
Wood, F.T., “Remedial English Grammar”, Macmillan, 2007.

B.Tech. (I Sem.)

20FE03 - DIFFERENTIAL EQUATIONS

L	T	P	Cr.
2	1	0	3

**Pre-requisites:** Nil

**Course Educational Objective:** The objective of this course is to introduce the first order and higher order differential equations, functions of several variables. The students also learn solving of first order partial differential equations.

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

**CO1:** Apply first order and first-degree differential equations to find orthogonal trajectories (**Apply – L3**).

**CO2:** Distinguish between the structure and methodology of solving higher order differential equations with constant coefficients (**Understand – L2**).

**CO3:** Apply various Numerical methods to solve initial value problem (**Apply – L3**).

**CO4:** Generate the infinite series for continuous functions and investigate the functional dependence (**Understand – L2**).

**CO5:** Solve partial differential equations using Lagrange's method (**Apply – L3**).

## UNIT –I

### Differential Equations of First Order and First Degree

Differential equations of first order and first degree –Exact and Non Exact differential Equations, Applications of differential equations – Orthogonal Trajectories.

## UNIT –II

### Linear Differential Equations of Higher Order

Homogeneous and Non Homogeneous Linear differential equations of second and higher order with constant coefficients with R.H.S. functions  $e^{ax}$ ,  $\sin(ax+b)$ ,  $\cos(ax+b)$ ,  $x^n$ ,  $e^{ax}V(x)$ ,  $xV(x)$ , Method of variation of parameters.

## UNIT – III

### Numerical solution of Ordinary Differential Equations

Numerical solution of Ordinary Differential equations, Solution by Taylor's series - Picard's Method of successive approximations.

Euler's Method - Runge- Kutta Methods.

## UNIT –IV

### Functions of several variables

Generalized Mean Value Theorem (without proof), Maclaurin's series, Functions of several variables, Jacobians (Cartesian and polar coordinates), Functional dependence. Maxima and Minima of function with two variables.

**UNIT – V****Partial Differential Equations**

Formation of Partial Differential Equations by elimination of arbitrary constants and arbitrary functions. Solution of first order and first degree linear partial differential equation – Lagrange's method.

**TEXTBOOKS:**

1. B.S. Grewal, "*Higher Engineering Mathematics*", 42<sup>nd</sup> Edition, Khanna Publishers, New Delhi, 2012.
2. B. V. Ramana, "*Higher Engineering Mathematics*", 1<sup>st</sup> Edition, TMH Publications, New Delhi, 2010

**Reference Books:**

1. M. D. Greenberg, "*Advanced Engineering Mathematics*", 2<sup>nd</sup> Edition, TMH Publications, New Delhi, 2011.
2. Erwin Kreyszig, "*Advanced Engineering Mathematics*", 8<sup>th</sup> Edition, John Wiley & sons, New Delhi, 2011.
3. W.E. Boyce and R. C. DiPrima, "*Elementary Differential Equations*", 7<sup>th</sup> Edition, John Wiley & sons, New Delhi, 2011.
4. S. S. Sastry, "*Introductory Methods of Numerical Analysis*", 5<sup>th</sup> Edition, PHI Learning Private Limited, New Delhi, 2012.

B.Tech. (I Sem.)

20FE07 - APPLIED PHYSICS

L	T	P	Cr.
2	1	0	3

**Pre-requisites:** Nil

**Course Educational Objectives:** It enables the students to understand the fundamental concepts of optics, quantum mechanics, free electron theory of metals, semiconductors, dielectrics and their applications.

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

**CO1:** Define the nature of interference and diffraction (**Remember – L1**).

**CO2:** Apply the lasers and optical fibers in different fields (**Apply – L3**).

**CO3:** Estimate the electrical conductivity of metals (**Understand – L2**).

**CO4:** Analyze the properties of semiconducting materials (**Understand – L2**).

**CO5:** Classify the different types of magnetic and dielectric materials (**Understand – L2**).

## UNIT – I

### Wave Optics

**Interference:** Principle of super position, Conditions for Interference, Interference in thin parallel film by reflection, Newton's rings (reflection), working principle of Interferometer.

**Diffraction:** Introduction, Fraunhofer diffraction at single slit- Diffraction due to circular aperture, Diffraction Grating- Resolving power of Grating.

## UNIT – II

### Lasers and optical fibers

**Lasers:** Introduction - Principle of laser (absorption, spontaneous and stimulated emission of radiation), Einstein Coefficients – Nd-YAG laser, Helium Neon laser- applications.

**Optical Fibers:** Optical Fiber principle, Structure of optical fiber, numerical aperture and acceptance angle, types of optical fibers - applications.

## UNIT – III

### Principles of Quantum Mechanics & Free electron theory

**Principles of quantum mechanics:** de Broglie Hypothesis, Davisson - Germer experiment, Schrodinger time independent and dependent wave equations, physical significance of the wave function – particle in a box.

**Free electron theory**

Classical free electron theory- Postulates, Advantages and Draw backs, Fermi-Dirac distribution function-Temperature dependence of Fermi- Dirac distribution function, Classification of Solids on the basis of Band theory.

## UNIT – IV

### Semiconductor physics

Conductivity of Intrinsic and Extrinsic semiconductors, Drift and Diffusion Current, Einstein relation, Hall Effect, Differences between direct and indirect Band Gap semiconductors, Solar Cell, Applications of Solar Cells.

**UNIT – V:****Magnetic & Dielectric materials**

Magnetic parameters, Classification of magnetic materials-Diamagnetic, paramagnetic and ferromagnetic materials, Hysteresis loop, soft and hard magnetic materials, Applications of Ferro magnetic materials

Dielectrics: polarization - Electronic and ionic polarization, orientation polarization (Qualitative), Local field, ClaussiusMosotti equation, Applications of dielectric materials.

**TEXTBOOKS:**

1. V. Rajendran, “*Engineering Physics*”, TMH, New Delhi, 6<sup>th</sup> Edition, 2014.
2. M.N. Avadhanulu, P.G. Kshirsagar, “*Engineering Physics*”, S. Chand & Co., 2<sup>nd</sup> Edition, 2014.

**REFERENCE BOOKS:**

1. M.N. Avadhanulu, TVS Arun Murthy, “*Applied Physics*”, S. Chand & Co., 2<sup>nd</sup> Edition, 2007.
2. P.K. Palani Samy, “*Applied Physics*”, Sci. Publ. Chennai, 4<sup>th</sup> Edition, 2016.
3. P. Sreenivasa Rao, K Muralidhar, “*Applied Physics*”, Him. Publi. Mumbai, 1<sup>st</sup> Edition, 2016.
4. HitendraK Mallik , AK Singh “ *Engineering Physics*”, TMH, New Delhi, 1<sup>st</sup> Edition, 2009.

B.Tech. (I Sem.)

**20CS01 - PROGRAMMING FOR PROBLEM SOLVING USING C**

L	T	P	Cr.
3	0	0	3

Pre-requisite : Nil

**Course Educational Objective:** The Objective of the course is to make learn the basic elements of C programming, control structures, derived data types, Modular programming, user defined structures, basics of files and its I/O operations.

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

- CO1:** Familiar with syntax and semantics of the basic programming language constructs. (**Understand - L2**)
- CO2:** Construct derived data types like arrays in solving problem. (**Apply - L3**)
- CO3:** Decompose a problem into modules and reconstruct it using various ways of user-defined functions. (**Apply - L3**)
- CO4:** Use user-defined data types like structures and unions and its applications to solve problems. (**Apply - L3**)
- CO5:** Discuss various file I/O operations and its application. (**Understand - L2**)

**UNIT – I**

**Introduction to Problem solving through C-Programming:** Problem Specification, Algorithm / pseudo code, flowchart, examples.

**C-Programming:** Structure of C program, identifiers, basic data types and sizes, Constants, variables, Input-output statements, A sample c program, operators, expressions, type conversions, conditional expressions, precedence of operators and order of evaluation.

**Control statements:** if, if else, else if ladder and switch statements, while, do-while and for statements, break, continue, go to and labels.

**UNIT – II**

**Arrays-** concept, declaration, definition, accessing elements, storing elements, two dimensional and multi-dimensional arrays.

**Character Arrays:** declaration, initialization, reading, writing strings, string handling functions, pre-processor Directives, and macros.

**Applications of Arrays:** Linear search, Binary search, Bubble Sort.

**UNIT – III**

**Pointers-** concepts, declaring & initialization of pointer variables, pointer expressions, pointer arithmetic, pointers and arrays, pointers and character arrays, pointers to pointers.

**Functions:** basics, category of functions, parameter passing techniques, recursive functions-comparison with Iteration, Functions with arrays, Standard library functions, dynamic memory management functions, command line arguments.

**Storage classes** - auto, register, static and extern.

**UNIT – IV**

**Derived types-** structures- declaration, definition, and initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self-referential structures, unions, typedef.

**UNIT – V**

**Files** – concept of a file, text files and binary files, streams, standard I/O, Formatted I/O, file I/O operations, error handling.

**Textbook:**

1. ReemaThareja, Programming in C, Oxford University Press, 2nd Edition, 2015

**Reference books:**

1. Jeri R.Hanly, Elliot B.Koffman, Problem Solving and Program Design in C, Pearson Publishers, 7th Edition, 2013
2. E Balagurusamy, Computer Programming, McGraw Hill Education, 8th Edition.
3. C: The Complete Reference, McGraw Hall Education, 4<sup>th</sup> Edition.
4. PradeepDey, Manas Ghosh, Programming in C, Oxford University Press, 2nd Edition, 2011.
5. Stephen G.Kochan, Programming in C, Pearson Education, 3rd Edition, 2005.



**20EE02 - BASIC ELECTRICAL AND ELECTRONICS  
ENGINEERING**

**B.Tech. ( ISem.)**

L	T	P	Cr.
3	0	0	3

**Prerequisite:** Physics

**Course Educational Objective:** This course enables student to illustrate the basics of applied electricity and electronics.

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

- CO1:** Apply network reduction techniques to simplify electrical circuits (**Apply – L3**).
- CO2:** Illustrate the working principle of DC machines and transformers (**Understand – L2**).
- CO3:** Understand V-I characteristics of semiconductor devices (**Understand – L2**).
- CO4:** Illustrate the configuration of Transistors and their applications (**Understand – L2**).

**UNIT – I: Electrical Circuit Fundamentals**

Basic definitions, Types of elements-active and passive, Ohm's Law, Kirchhoff's Laws-Network reduction techniques- series, parallel, star to delta, delta to star transformations, source transformation (for resistive networks), mesh analysis, nodal analysis (Basic problems).

**UNIT – II: DC Network Theorems and AC Fundamentals**

Theorems-Superposition, Thevenin's, Norton's and Maximum Power Transfer (Basic problems in DC excitation only)

Peak, R.M.S, average, instantaneous values, form factor and peak factor– periodic waveforms – Phase and Phase difference –concepts of reactance, impedance, susceptance and admittance, real, reactive and apparent powers, Power Factor- resonance-bandwidth-quality factor.

**UNIT – III: DC Machine Fundamentals and Single-Phase Transformers**

DC generator principle, constructional details, emf equation, types of generators (Theory only).

DC motor principle, Back emf, types of motor (Theory only).

Construction and Principle of operation of single-phase transformers-Emf equation

**UNIT-IV: P-N Junction Diode and Zener Diode**

**P-N Junction Diode:** Operation and V-I characteristics of PN junction diode, Rectifiers-Half Wave Rectifier, Full Wave Rectifier-Bridge type, Zener Diode-Voltage Regulator.

**UNIT – V: Transistors**

Construction, Principle of Operation, Symbol, CB, CE configurations, JFET, MOSFET and application of transistor as an amplifier (Theory only).

**TEXTBOOKS:**

1. A.Sudhakar and Shyammohan S Palli, “Electrical Circuits” Tata McGraw-Hill, 3<sup>rd</sup> Edition.2017
2. M.S.Sukhija, T.K.Nagsarkar, “Basic Electrical and Electronics Engineering”, Oxford University Press, 2016 Edition.

**REFERENCE BOOK(S):**

1. Kothari and Nagarath, “Basic Electrical Engineering”, TMH Publications, 3<sup>rd</sup> Edition.2013
2. G.S.N.Raju, “Electronic Devices and Circuits”, I.K.International.2006

B.Tech. (I Sem.)

**20FE51 - PROFESSIONAL COMMUNICATION  
SKILLS LAB**

L	T	P	Cr.
0	0	2	1

**Pre-requisites** : Nil

**Course Educational Objective:** To improve the proficiency of students in English with an emphasis on better communication in formal and informal situations; Develop speaking skills required for expressing their knowledge and abilities and to face interviews with confidence.

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

- CO1:** Introduce oneself and others using appropriate language and details (**Understand – L2**)
- CO2:** Comprehend short talks and speak clearly on a specific topic using error free English (**Understand – L2**)
- CO3:** Report effectively after participating in informal discussions ethically (**Remember –L1**)
- CO4:** Interpret data aptly, ethically & make oral presentations (**Apply – L3**)

**Syllabus: Professional Communication Skills Lab (PCS) shall have two parts:**

- **Computer Assisted Language Learning (CALL) Lab** for 60 students with 60 systems, LAN facility and English language software for self- study by learners.
- **Interactive Communication Skills (ICS) Lab** with movable chairs and audio-visual aids with a P.A System, a T. V., a digital stereo – audio & video system and camcorder etc.

**Exercise – I**

**CALL Lab: Understand** - Sentence structure

**ICS Lab: Practice** - Listening: Identifying the topic, the context and specific information  
Speaking: Introducing oneself and others

**Exercise – II**

**CALL Lab: Understand** - Framing questions

**ICS Lab: Practice** - Listening: Answering a series of questions about main idea and supporting ideas after listening to audio text  
Speaking: Discussing in pairs/small groups on specific topics; Delivering short structured talks using suitable cohesive devices (JAM)

**Exercise – III**

**CALL Lab: Understand** - Comprehension practice – Strategies for Effective Communication

**ICS Lab: Practice** - Listening: Listening for global comprehension and summarizing  
Speaking: Discussing specific topics in pairs/small groups, reporting what is discussed

#### Exercise – IV

**CALL Lab: Understand-** Features of Good Conversation – Strategies for Effective Communication.

**ICS Lab: Practice** -Listening: Making predictions while listening to conversations/transactional dialogues with/without video

Speaking: Role – plays – formal & informal – asking for and giving information / directions / instructions / suggestions

#### Exercise – V

**CALL Lab: Understand** - Features of Good Presentation, Methodology of Group Discussion

**ICS Lab: Practice** - Introduction to Group Discussions

Listening: Answering questions, identifying key terms and understanding concepts

Speaking: Formal Oral & Poster presentations on topics from academic contexts without the use of PPT

#### Lab Manual:

1. Prabhavati. Y & et al, “English All Round – Communication Skills for Undergraduate Learners”, Orient BlackSwan, Hyderabad, 2019.

#### Suggested Software:

1. Digital Mentor: Globarena, Hyderabad, 2005
2. Sky Pronunciation Suite: Young India Films, Chennai, 2009
3. Mastering English in Vocabulary, Grammar, Spelling, Composition, Dorling Kindersley, USA, 2001
4. Dorling Kindersley Series of Grammar, Punctuation, Composition, USA, 2001
5. Oxford Talking Dictionary, The Learning Company, USA, 2002
6. Learning to Speak English - 4 CDs. The Learning Company, USA, 2002  
Cambridge Advanced Learners English Dictionary (CD). Cambridge University Press, New Delhi, 2008.

L	T	P	Cr.
0	0	3	1.5

**Pre-requisites :** Nil

**Course Educational Objective:** This course enables the students to acquire theoretical ideas, analytical techniques, and graphical analysis, by completing a host of experiments with the procedures and observational skills for appropriate use of simple and complex apparatus.

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

- CO1:** Analyze the wave characteristics of light (**Understand – L2**).
- CO2:** Estimate the magnetic field using Stewart's and Gee's apparatus (**Understand – L2**).
- CO3:** Verify the characteristics of semiconductor diodes (**Apply – L3**).
- CO4:** Determine the acceptance angle and numerical aperture of optical fiber (**Apply – L3**).
- CO5:** Improve report writing skills and individual teamwork with ethical values (**Understand – L2**).

## List of Experiments

(Any of the 10 experiments are required to be conducted)

### GENERAL EXPERIMENTS:

1. Determine the energy band gap of a semiconductor Diode.
2. Study the characteristics of Zener Diode.
3. Study the magnetic field along the axis of a current carrying circular coil using Stewart's & Gee's apparatus and to verify Biot - Savart's law.
4. Study the characteristics of Solar cell
5. Determination of dielectric constant by charging and discharging method.
6. Study the characteristics of Photo diode.
7. Determination of resistivity of semiconductor by four probe method.

### OPTICS LAB EXPERIMENTS:

8. Determine the wavelength of a laser radiation.
9. Determine the width of a single slit by forming diffraction pattern.
10. Determine the Radius of Curvature of a Plano - Convex lens by forming Newton's Rings.
11. Determine the Wavelengths of various spectral lines by using diffraction grating.
12. Resolving power of grating.
13. Determine the acceptance angle and numerical aperture of a fiber.
14. Measure the bending losses in the optical fiber cable at different wavelengths.

B.Tech. (I Sem.)

**20CS51 - PROGRAMMING FOR PROBLEM SOLVING USING C LAB**

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : NIL

**Course Educational Objective:** The objective of the course is to learn the basic elements of C Programming Structures like Data Types, Expressions, Control Statements, and Various I/O Functions and to solve simple mathematical problems using control structures. Design and implementation of various software components, which solve real world problems.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

**CO1:** Apply control structures of C in solving computational problems. (**Apply - L3**)

**CO2:** Implement derived data types & use modular programming in problem solving (**Apply- L3**)

**CO3:** Implement user defined data types and perform file operations. (**Apply- L3**)

**CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**# of modules at most 10 can be taught and all the modules should be in line with theory.**

Module 1: Introduction to Raptor Tool.

Module 2: Problem solving using Raptor Tool.

Module 3: Exercise Programs on Basics of C-Program.

Module 4: Exercise Programs on Control Structures.

Module 5: Exercise Programs on Loops & nesting of Loops.

Module 6: Exercise Programs on Arrays & Strings.

Module 7: Exercise Programs on Pointers.

Module 8: Exercise Programs on Functions.

Module 9: Exercise Programs on user defined data types.

Module 10: Exercise Programs on Files.

B.Tech. (I Sem.)

20IT51 - IT WORKSHOP

L	T	P	Cr.
0	0	3	1.5

Pre-requisite : NIL

**Course Educational Objective:**

The objective of the course is to impart knowledge about the components of PC, Assembling PC, Installation of OS, softwares like MS-Office, LaTeX and concepts related to Networking, Internet as well as antivirus.

**COURSE OUTCOMES (COs):** At the end of this course, the student will be able to:

- CO1:** Identify the basic hardware components, keyboard shortcuts, assembling and disassembling of the system (PC). **(Understand- L2)**
- CO2:** Demonstrate Operating System installation, apply various commands of linux operating system, and networking. **(Apply - L3)**
- CO3:** Create web pages using HTML, documents using applications like LaTeX, Google forms and use application software packages: MS-Word, MS-Excel, MS-Power Point to create documents and presentation. **(Apply- L3)**
- CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**PC Hardware****Week 1**

Task: Identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor.

**PC Assembling****Week 2**

Task: Every student should disassemble and assemble the PC back to working condition. Lab instructors should verify the work and follow it up with a Viva. In addition, students need to go through the video, which shows the process of assembling a PC. A video would be given as part of the course content.

## **Operating System Installation**

### **Week 3**

**Task 1:** Every student should individually install MS windows and Linux/ Ubuntu OS on the personal computer. Lab instructor should verify the installation and follow it up with a Viva.

**Task 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

## **Network & Internet**

### **Week 4**

**Task 1: Networking Commands:**

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

### **Week 5**

**Task 1 :Internet Services:**

- WebBrowserusageandadvancedsettingslikeLAN,proxy,content,privacy,security,cookies, extensions/ plugins
- Malicious Software : Virus,Worm,Logic Bomb,Trojan Horse
- Antivirusinstallation,configuringa firewall,blockingpop-ups
- Mailcreationandusage,CreatingaDigitalProfileonLinkedIn

## **Productivity Tools**

### **Week 6**

**Task 1:** Basic HTML tags, Introduction to HTML5 and its tags, Introduction to CSS3andits properties. Preparation of a simple website/ homepage

Every student should develop your home page using HTML Consisting of your photo, name, address and education details as table and your skill set as a list.



## Office Tools

### Week 7

Task-1: Demonstration and Practice on Text Editors like Notepad++, Sublime Text, Atom, Brackets, Visual code, etc

### Week 8

Task-1: Demonstration and practice on Microsoft Word, PowerPoint, Microsoft Excel

### Week 9

Task-1: Demonstration and practice on LaTeX and produce professional pdf documents.

### Week 10

Task-1: Creating online documents using Google docs- Create and share Bio-data form.

### References:

1. Comdex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech.
2. Introduction to Information Technology, ITL Education Solutions limited, Pearson Education.
3. Computer Fundamentals, Anita Goel, Pearson Education, 2017
4. PC Hardware Trouble Shooting Made Easy, TMH
5. <https://www.google.com/docs/about/>
6. <https://support.google.com/sites/answer/6372878?hl=en>

B.Tech.(IISem.)

20FE02 - PROFESSIONAL COMMUNICATION - II

L	T	P	Cr.
2	0	0	2

**Pre-requisites:** Nil

**Course Educational Objective:** To improve English language proficiency of the students in various aspects like vocabulary, grammar, communication skills, listening skills, reading & writing skills.

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

- CO1:** Produce a coherent paragraph interpreting a figure/graph/chart/table (**Understand – L2**)
- CO2:** Comprehend the given texts thoroughly by guessing the meanings of the words contextually (**Understand – L2**)
- CO3:** Use language appropriately for describing /comparing/contrasting/giving directions & suggestions (**Remember – L1**)
- CO4:** Write formal/informal dialogues with an understanding of verbal/non-verbal features of communication. (**Understand – L2**)
- CO5:** Write well structured essays; Reports & Résumé (**Apply – L3**)

**UNIT - I**

**Fabric of Change-** ‘H.G. Wells and the Uncertainties of Progress – Peter J. Bowler’; Reading: Studying the use of Graphic elements in texts; Grammar & Vocabulary: Quantifying Expressions; Adjectives and adverbs; Comparing and Contrasting; Degrees of Comparison; Writing: Information Transfer

**UNIT - II**

**Tools for Life -** ‘Leaves from the Mental Portfolio of a Eurasian – Sui Sin Far’; Reading: Global Comprehension; Detailed Comprehension; Grammar & Vocabulary: Active & Passive Voice; Idioms & Phrases; Writing: Structured Essays using suitable claims and evidences

**UNIT - III**

**‘Homi Jahangir Bhabha’;**

Grammar & Vocabulary: Words often confused; Common Errors; Writing: Incident & Investigation Reports

**UNIT - IV**

**‘Jagadish Chandra Bose’;** Grammar & Vocabulary: Use of Antonyms; Correction of Sentences; Writing: Dialogue Writing

**UNIT - V**

**‘Prafulla Chandra Ray’;** Grammar & Vocabulary: Analogy; Sentence Completion; Writing: Writing a Résumé

**TEXT BOOKS:**

1. Prabhavati. Y & et al, “English All Round – Communication Skills for Undergraduate Learners”, Orient Black Swan, Hyderabad, 2019.
- 2 “The Great Indian Scientists” published by Cengage Learning India Pvt. Ltd., Delhi, 2017

**REFERENCE BOOKS:**

1. Swan, M., “Practical English Usage”, Oxford University Press, 2016.
2. Kumar,S and Latha, P, “Communication Skills”, Oxford University Press, 2018.
3. Rizvi Ashraf M., “Effective Technical Communication”, Tata Mc Graw Hill, New Delhi, 2008.
4. Baradwaj Kumkum, “Professional Communication”, I.K.International Publishing House Pvt.Ltd., New Delhi, 2008.
5. Wood,F.T., “Remedial English Grammar”, Macmillan, 2007.

B.Tech. (II Sem.)

**20FE04 - LINEAR ALGEBRA AND  
TRANSFORMATION TECHNIQUES**

L	T	P	Cr.
2	1	0	3

**Pre-requisites:** Nil

**Course Educational Objective:** In this course, students learn Matrix Algebra and introduced with transformation techniques such as Laplace Transforms and Z – Transforms.

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

- CO1:** Investigate the consistency of the system of equations and solve them (**Apply – L3**).  
**CO2:** Determine the eigen vectors and inverse, powers of a matrix using Cayley – Hamilton theorem (**Apply – L3**).  
**CO3:** Use the concepts of Laplace Transforms to various forms of functions. (**Understand – L2**).  
**CO4:** Solve ordinary differential equations by using Laplace Transforms (**Apply – L3**).  
**CO5:** Apply Z - Transforms to solve difference equations(**Apply – L3**).

**UNIT – I****System of Linear Equations**

Matrices - Rank- Echelon form, Normal form, PAQ form– Solution of Linear Systems – Homogeneous system of equations and Non-Homogeneous system of equations.

**UNIT – II****Eigen Values and Eigen Vectors**

Eigen values – Eigen Vectors – Properties – Cayley-Hamilton Theorem – Inverse and Powers of a matrix by using Cayley-Hamilton Theorem.

**UNIT – III****Laplace Transforms**

Laplace transforms of standard functions –Linear Property - Shifting Theorems, Change of Scale Property  
 Multiplication and Division by 't' - Transforms of derivatives and integrals – Unit step function – Dirac's delta function.

**UNIT – IV****Inverse Laplace Transforms**

Inverse Laplace transforms– Linear Property - Shifting Properties - Convolution theorem, Applications of Laplace transforms to ordinary differential equations.

**UNIT – V****Z-Transforms**

Z-transform – properties – Damping rule – Shifting rule – Initial and final value theorems - Inverse Z –transform - Convolution theorem – Solution of difference equation by Z-transforms.

**TEXTBOOKS:**

1. B.S. Grewal, “*Higher Engineering Mathematics*”, 42<sup>nd</sup> Edition, Khanna Publishers, New Delhi, 2012.
2. B. V. Ramana, “*Higher Engineering Mathematics*”, 1<sup>st</sup> Edition, TMH Publications, New Delhi, 2010.

**REFERENCE BOOKS:**

1. M. D. Greenberg, “*Advanced Engineering Mathematics*”, 2<sup>nd</sup> Edition, TMH Publications, New Delhi, 2011.
2. Erwin Kreyszig, “*Advanced Engineering Mathematics*”, 8<sup>th</sup> Edition, John Wiley & sons, New Delhi, 2011.
3. W.E. Boyce and R. C. DiPrima, “*Elementary Differential Equations*”, 7<sup>th</sup> Edition, John Wiley & sons, New Delhi, 2011.

L	T	P	Cr.
3	0	0	3

B.Tech (II Sem)

**20FE06 - ENGINEERING CHEMISTRY****Pre-requisites:** Nil

**Course Educational Objectives:** It enables the students to understand the fundamental concepts of chemistry and to provide them with the knowledge of industrial problems and finding the solutions. It helps to strengthen the basic concepts of electrochemistry, corrosion, nanotechnology, polymers, liquid crystals and analytical techniques.

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

- CO1:** Apply Nernst Equation for calculating electrode cell potentials and compare batteries different applications (**Apply – L3**).
- CO2:** Apply principles of corrosion for design and effective maintenance of various equipment (**Apply – L3**).
- CO3:** Analyse the suitability of advanced materials like nano materials in electronics and medicine (**Understand – L2**).
- CO4:** Identify the importance of liquid crystals, polymers in advanced technologies (**Understand – L2**).
- CO5:** Apply the principles of analytical techniques in chemical analysis (**Apply – L3**).

**UNIT – I****Electro Chemistry & Batteries**

Types of Electrodes - Calomel Electrode, Glass Electrode, Calculation of EMF of Cell, Applications of Nernst Equation & Electro chemical Series, Batteries -Lead-acid Battery, Lithium ion Battery, H<sub>2</sub> – O<sub>2</sub> Fuel Cell, Mg - Cu reserve battery.

**UNIT – II****Science of Corrosion**

Dry Corrosion (Direct Chemical corrosion) - Types of dry corrosion-oxidative corrosion, Pilling Bed worth rule, corrosion by other gases and liquid metal corrosion; Wet Corrosion (Electro Chemical corrosion) - Mechanism- oxygen absorption, hydrogen evolution, types of wet corrosion, Galvanic Corrosion, Concentration Cell Corrosion, passivity and Galvanic series; Factors Influencing Corrosion - Nature of metal (Purity, position in galvanic series, relative area of cathode & anode, nature of surface film) and nature of environment (temperature, humidity, atmospheric pollution and nature of ions in the medium); Control of Corrosion: Cathodic Protection - Sacrificial anode and impressed current methods, electro plating and metal cladding.

**UNIT – III****Chemistry of Engineering Materials**

Nano Materials - Extraordinary changes observed at nano size of materials and reasons, types of nano-materials, Gas-Phase Synthesis of nanomaterials, Applications; Materials in Electronic devices: Very brief note on raw materials that make IC units of CPU, GPU, RAM, PCBs, hard disks and other electronic devices with special reference to polymers;

Molecular Switches - Characteristics of Molecular motors and machines, Rotaxanes and Catenanes as artificial molecular machines, prototypes – linear motions in rotaxanes, an acid-base controlled molecular shuttle, molecular elevator, automated light-powered molecular motor.

**UNIT – IV****Liquid Crystals & Polymers**

Liquid crystals - Identification and structural aspects of molecules to form liquid crystals; Classification of liquid crystals - Thermo tropic liquid crystals and types, lyotropic liquid crystals. Mechanism of working of liquid crystals and applications; Polymers - Differences between thermoplasts and thermosets, Types of polymerization with examples; Plastics - Preparation properties and engineering applications of P.M.M.A, Teflon, Polycarbonate; Rubbers - Structure of raw rubber and vulcanized rubber, Preparation properties and engineering applications of Polyurethane, Buna-S, conducting polymers; Bio-degradable polymers - PLA & PGA (Polylactic Acid and Polyglycolic Acid).

**UNIT – V****Analytical Techniques**

Types of analysis; Physical analysis: Analysis of physical characteristics; Chemical analysis: Gravimetric and volumetric analysis (basic concept only); Instrumental analysis: Electro analytical techniques – Introduction; Conductometric techniques: strong acid-strong base and strong acid-weak base, weak acid -strong base and weak acid -weak base & advantages; Potentiometric techniques: Acid-base and oxidation-reduction titrations-advantages; Colorimetric techniques: Principle and determination of iron by using thiocyanate as a reagent.

**TEXTBOOKS**

1. Shikha Agarwal, “A Text book of Engineering Chemistry”, Cambridge University Press, New Delhi, 1<sup>st</sup> Edition, 2015.
2. Jain, Jain, “A textbook of Engineering Chemistry”, Dhanpat Rai Publishing Company, New Delhi, 16<sup>th</sup> Edition, 2015.

**REFERENCE BOOKS**

1. Shashi Chawla, “A Text book of Engineering Chemistry”, Dhanpat Rai Publishing Company, New Delhi, 3<sup>rd</sup> Edition, 2003.
2. S.S. Dara, S.S. Umare, “A Text book of Engineering Chemistry”, S. Chand Publications, New Delhi, 12<sup>th</sup> Edition, 2010.
3. Prasanta Rath, B. Rama Devi, Ch. Venkata Ramana Reddy, SubhenduChakroborty, “Engineering Chemistry”, Cengage Learning India, 1<sup>st</sup> Edition, 2019.

B.Tech. (II Sem.)

20CS05 - PYTHON PROGRAMMING

L	T	P	Cr.
3	0	0	3

Pre-requisite : Problem Solving Skills

**Course Educational Objective:** The objective of the course is to provide basic knowledge of python. python programming is intended for software engineers, system analysts, program managers and user support personnel who wish to learn the Python programming language for Problem solving and programming capability.

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

- CO1:** Identify basic building blocks of python to solve mathematical problems. (**Understand-L2**)
- CO2:** Apply the in-built data structures like list, tuple, set and dictionary for solving problems. (**Apply - L3**)
- CO3:** Use exception-handling mechanism to catch run-time errors (**Apply - L3**)
- CO4:** Demonstrate compelling concepts about string manipulation, regular expressions, and file handling. (**Understand - L2**)
- CO5:** Demonstrate object-oriented programming principles of python. (**Understand - L2**)

### UNIT-I

**Introduction to Python:** History of Python, Usage of Python interpreter, Python Shell, Indentation, Python Built-in types, Variables, Assignment, Input-Output Statements, Simple Programs, Identifiers and keywords, Literals. **Operators:** Arithmetic operators, Relational operators, Logical operators, Assignment operators, Bit-wise operators, Python Membership Operators (in & not in), Python Identity Operators (is & is not), Operator precedence.

**Control Structures:** Conditional Statements - if, if-else, Nested if-else. Jumping Statements - continue, break, and pass. Python Loops - while, for, Nested loops with Programming Examples, Mathematical Functions and Constants (import math), Random Number Functions.

### UNIT-II

**Lists:** Concept, Creating and Accessing Elements, Updating & Deleting Lists, basic List Operations, Reverse, Indexing, Slicing and Matrices, Built-in List Functions.

**Tuples:** Introduction, Creating & Deleting Tuples, Accessing values in a Tuple, Updating tuples, Delete Tuple Elements, basic Tuple Operations, Indexing, Slicing and Matrices, built- in tuple Functions.

**Sets:** Concept, Operations.

**Dictionaries:** Introduction, Accessing values in dictionaries, working with dictionaries, Properties, Functions.

### UNIT-III

**Functions:** Defining a Function, Calling a Function, Types of Functions, Function Arguments, Anonymous functions, Global and Local Variables, Recursion

**Modular Design:** Creating modules, import statement, from, Date and Time Module.

**Exception Handling:** Exception, Exception Handling, except clause, Try, finally clause, User Defined Exceptions.



#### UNIT-IV

**Python strings:** Concept, Slicing, Escape characters, String Special Operations, String formatting Operator, Triple Quotes, Raw String, Unicode Strings, and Built-in String methods.

**Regular Expression Operations:** Using Special Characters, Regular Expression Methods, Named Groups in Python Regular Expressions, Regular Expression with glob Module.

**Files:** Introduction to files, file operations- Read, write, and search

#### UNIT-V

**Object Oriented Programming OOP in Python:** Classes, 'self-variable', Methods, Constructor Method, Inheritance, Overriding Methods, and Data hiding.

#### TEXTBOOKS:

1. Reema Thareja, “Python Programming Using Problem Solving Approach”, Oxford Publications
2. Python for Everybody: Exploring Data In Python 3 by [Dr. Charles Russell Severance](#), [Sue Blumenberg](#)

#### Reference books:

1. Gowrishankar S and Veena A, “Introduction to Python Programming”, CRC Press, Taylor, and Francis Group – A CHAPMAN & HALLBOOK.
2. R. Nageswara Rao, “Core python programming”, Dreamtech, 2017.
3. Y. Daniel Liang, “Revel for Introduction to Python Programming and Data Structures”, Pearson Publications.

B.Tech. (II Sem.)

20CS03 - DATA STRUCTURES

L	T	P	Cr.
3	0	0	3

Pre-requisite : Programming Language

**Course Educational Objectives:**

The objective of the course is to make students familiar with writing algorithms to implement different data structures like stacks, queues, trees and graphs, and various sorting techniques.

**Course Outcomes (COs):** *At the end of this course, the student will be able to*

- CO 1: Write the algorithms for various operations on list using arrays and linked list and analyze the time complexity of its operations.(**Understand - L2**)
- CO 2: Apply linear data structures like stack and queue in problem solving.(**Apply - L3**)
- CO 3: Demonstrate various searching and sorting techniques and compare their computational complexities in terms of space and time.(**Understand - L2**)
- CO 4: Write the algorithms for various operations on binary trees, binary search trees and AVL trees.(**Understand - L2**)
- CO 5: Demonstrate graph traversal techniques and hashing techniques.(**Understand - L2**)

**UNIT - I****Algorithm Analysis:**

Introduction to Algorithm, Algorithm Analysis, Asymptotic Notations.

**Introduction to arrays and Abstract Data Type (ADT)**

**Lists:** List using arrays and linked list- Singly Linked List, Doubly Linked List, Circular LinkedList.

**UNIT – II**

**Stacks:** Stack ADT, Implementation using arrays and linked list.

Applications of stacks: Infix to postfix expression conversion, Evaluation of Postfix expressions and balancing the symbols.

**Queues:**

Queue: Queue ADT, Implementation of Queue using arrays and linked list, circular queue, DEQUE

**UNIT - III**

**Sorting:** Bubble sort, Insertion Sort, Selection sort, Merge Sort, Quick Sort & Heap Sort

**UNIT - IV**

**Trees:** Introduction, Tree traversals, Binary Trees, Binary Search Trees, Balanced Binary search tree - AVL Trees and its operations.

**UNIT - V**

Graphs: Fundamentals, Representation of graphs, Graph Traversals: BFS, DFS.

Hashing: Hash Table, Hash Function, Collision resolution Techniques- separate Chaining, Open addressing, rehashing.

**TEXTBOOKS:**

1. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, Pearson Education, 2<sup>nd</sup> edition [1,2,3 units].
2. ReemaThareja, Data Structures using c, Oxford Publications [3,4,5].

**REFERENCE BOOKS:**

1. Langson, Augenstein&Tenenbaum, ‘Data Structures using C and C++’, 2nd Ed, PHI.
2. RobertL.Kruse, Leung and Tando, ‘Data Structures and Program Design in C’, 2ndedition, PHI.

B.Tech. (II Sem.)

20MC01 - CONSTITUTION OF INDIA

L	T	P	Cr.
2	0	0	0

**Pre-requisites:** Nil**Course Educational Objectives**

- To enable the student to understand the importance of constitution.
- To understand the structure of Executive, Legislature and Judiciary.
- To understand Philosophy of fundamental rights and duties.
- To understand the autonomous nature of constitution bodies like Supreme Court and High Court Controller and Auditor General of India and Election Commission of India.
- To understand the Central and State relation, financial and administrative.

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

- CO 1: Understand history and philosophy of constitution with reference to Preamble, Fundamental Rights and Duties (**Understand – L2**).
- CO 2: Understand the concept of Unitary and Federal Government along with the role of President, Prime Minister and Judicial System (**Understand – L2**).
- CO 3: Understand the structure of the state government, Secretariat, Governor and Chief Minister and their functions (**Understand – L2**).
- CO 4: learn local administration viz. Panchayat, Block, Municipality and Corporation (**Understand – L2**).
- CO 5: learn about Election Commission and the process and about SC, ST, OBC and women (**Understand – L2**).

**UNIT – I:**

Introduction to Indian Constitution: ‘Constitution’ meaning of the term, Indian Constitution – Sources and Constitutional History, Features – Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy.

**UNIT – II:**

Union Government and its Administration Structure of the Indian Union: Federalism Centre – State relationship, President: Role, Power and Position. Prime Minister (PM) and Council of Ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha. The Supreme Court and High Court: Powers and Functions.

**UNIT – III:**

State Government and its Administration Governor – Role and Position – Chief Minister (CM) and Council of Ministers. State Secretariat: Organization, Structure and Functions.

**UNIT – IV:**

A Local Administration -- Role and Importance, Municipalities – Mayor and Role of Elected Representative, Panchayati Raj: Functions of Panchayati Raj Institution, Zilla Panchayat, Elected Officials and their roles, Village level – Role of Elected and Appointed officials.

**UNIT – V:**

Election Commission: Election Commission – Role of Chief Election Commissioner and Election Commissionerate State Election Commission: Functions and Commissions for the welfare of SC/ST/OBC and Women.

**Reference Books**

1. Durga Das Basu, Introduction to the Constitution of India, Prentice – Hall of India Pvt.Ltd., New Delhi.
2. Subash Kashyap, Indian Constitution, National Book Trust.
3. J.A. Siwach, Dynamics of Indian Government and Politics.
4. D.C. Gupta, Indian Government and Politics.
5. H.M.Sreevai. Constitutional Law of India, 4th edition in 3 volumes (Universal LawPublication).
6. J.C. Johari, Indian Government and Politics Hans.
7. J.Raj, Indian Government and Politics.

8. M.V. Pylee, Indian Constitution, Durga Das Basu, Human Rights in Constitutional Law, Prentice – Hall of India Pvt. Ltd., New Delhi.
9. Noorani, A.G. (South Asia Human Rights Documentation Centre), Challenges to Civil Right). Challenges to Civil Rights Guarantees in India, Oxford University Press 2012.

**E-Resources:**

1. [nptel.ac.in/courses/109104074/8](https://nptel.ac.in/courses/109104074/8).
2. [nptel.ac.in/courses/109104045](https://nptel.ac.in/courses/109104045).
3. [nptel.ac.in/courses/101104065](https://nptel.ac.in/courses/101104065).
4. [www.hss.iitb.ac.in/en/lecture-details](http://www.hss.iitb.ac.in/en/lecture-details).
5. [www.iitb.ac.in/en/event/2nd-lecture-institute-lecture-series-indianconstitution](http://www.iitb.ac.in/en/event/2nd-lecture-institute-lecture-series-indianconstitution)

B.Tech. (II Sem.)

20FE53 - ENGINEERING CHEMISTRY LAB

L	T	P	Cr.
0	0	3	1.5

Pre-requisites: Nil

Course Educational Objectives: This course enables the students to analyze water samples and perform different types of volumetric titrations. It provides them with an overview of preparation of polymers and analytical techniques.

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

- CO1: Assess alkalinity of water based on the procedure given (Understand – L2).  
 CO2: Distinguish different types of titrations in volumetric analysis after performing the experiments listed in the syllabus (Understand – L2).  
 CO3: Acquire practical knowledge related to preparation of polymers. (Understand – L2).  
 CO4: Exhibit skills in performing experiments based on theoretical fundamentals. (Understand – L2).

List of Experiments

(Any of the 10 experiments are required to be conducted)

**Model Experiment**

1. Determination of  $\text{Na}_2\text{CO}_3$  using standard HCl solution.

**Water Analysis**

2. Determination of alkalinity of water sample.

**Complexometric Titrations**

3. Estimation of  $\text{Mg}^{+2}/\text{Zn}^{+2}/\text{Ca}^{+2}$  in given solution by using standard EDTA solution.

**Preparation of Polymers (only demonstration)**

4. Nylon Fibers
5. Bakelite

**Redox Titrations**

6. Estimation of Mohr's salt by using potassium permanganate.
7. Estimation of Mohr's salt by using potassium dichromate.
8. Estimation of copper (II) ion using standard hypo solution.

**Conductometric Measurements**

9. Estimation of amount of HCl conductometrically using standard NaOH solution.
10. Estimation of amount of HCl conductometrically using  $\text{NH}_4\text{OH}$  solution.

**Potentiometric Measurements**

11. Estimation of amount of HCl potentiometrically using NaOH solution.

**Estimations**

12. Measuring pH of the given sample solution using pH meter (demonstration only).
13. Estimation of Vitamin C in a given sample.

**Colorimetric Analysis**

14. Determination of Iron (III) by colorimetric method.

**REFERENCES**

1. LAB Manual

B.Tech. (II Sem.)

20CS54 - PYTHON PROGRAMMING LAB

L	T	P	Cr.
0	0	3	1.5

Pre-requisite: Basic Knowledge of Programming.

### Course Educational Objective:

The Objective of Python course is to lead the students from the basics of writing and running Python scripts in problem solving and to design and implement the modules and understands the working of classes and objects in python.

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

- CO 1:** Apply building blocks of Python in solving computational problems. (**Apply - L3**)
- CO 2:** Implement in-built data structures available in Python to solve computational problems. (**Apply - L3**)
- CO 3:** Implement modular programming, string manipulations and Object-oriented programming in python. (**Apply - L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

### Introduction: Language basics and example problems ( Two weeks)

- a) Implement Python Script for checking the given year is leap year or not.
- b) Implement Python Script for finding biggest number among 3 numbers.
- c) Implement Python Script for displaying reversal of a number.
- d) Implement Python Script to check given number is Armstrong or not.
- e) Implement Python Script to print sum of N natural numbers.
- f) Implement Python Script to check given number is palindrome or not.
- g) Implement Python script to print factorial of a number.
- h) Implement Python Script to print all prime numbers within the given range.
- i) Implement Python Script to calculate the series:  $S=1+x+x^2+x^3+\dots+x^n$
- j) Implement Python Script to print the following pattern:

```

      *
    *  *
  *   *   *

```



**Module 1: Exercise Programs on Lists.**

- a) Write a Python script to display elements of list in reverse order.
- b) Write a Python script to find the minimum and maximum elements without using built-in operations in the lists.
- c) Write a Python script to remove duplicates from a list.
- d) Write a Python script to append a list to the second list.
- e) Write a Python script to count the number of strings in a list where the string length is 2 or more.

**Module 2: Exercise Programs on Tuples.**

- a) Write a Python script to create a tuple with different data types.
- b) Write a Python script to find the repeated items of a tuple.
- c) Write a Python script to replace last value of tuples in a list.  
Sample list: [(10, 20, 40), (40, 50, 60), (70, 80, 90)]  
Expected Output: [(10, 20, 100), (40, 50, 100), (70, 80, 100)]
- d) Write a Python script to sort a tuple by its float element.  
Sample data: [('item1', '12.20'), ('item2', '15.10'), ('item3', '24.5')]  
Expected Output: [('item3', '24.5'), ('item2', '15.10'), ('item1', '12.20')]

**Module 3: Exercise Programs on Sets and Dictionaries.**

- a) Write a Python script to add member(s) in a set.
- b) Write a Python script to perform Union, Intersection, difference and symmetric difference of given two sets.
- c) Write Python script to test whether every element in S is in T and every element in T is in S.
- d) Write a Python script to sort (ascending and descending) a dictionary by value.
- e) Write a Python script to check whether a given key already exists or not in a dictionary.
- f) Write a Python script to concatenate following dictionaries to create a new one.
- g) Sample Dictionary : dic1={1:10, 2:20} dic2={3:30, 4:40} dic3={5:50,6:60}  
Expected Result : {1: 10, 2: 20, 3: 30, 4: 40, 5: 50, 6: 60}
- h) Write a Python script to print a dictionary where the keys are numbers between 1 and 15 (both included) and the values are square of keys.
- i) Write a Python program to map two lists into a dictionary.

**Module 4: Exercise Programs on functions and recursion.**

- a) Define a function max\_of\_three() that takes three numbers as arguments and returns the largest of them.
- b) Write a program which makes use of function to display all such numbers which are divisible by 7 but are not a multiple of 5, between given range X and Y.
- c) Define functions to find mean, median, mode for the given numbers in a list.
- d) Define a function which generates Fibonacci series up to n numbers.
- e) Implement a python script for factorial of number by using recursion.
- f) Implement a python script to find GCD of given two numbers using recursion.

**Module 5: Exercise programs on Date and Time Modules.**

- a) Write a Python script to get the current time in Python.
- b) Write a Python script to get current time in milliseconds in Python
- c) Write a Python script to print next 5 days starting from today.

**Module 6: Exercise programs on Exception Handling.**

- a) Write a Python script to handle simple errors by using exception handling mechanism.
- b) Write a Python script to handle multiple errors with one except statement.

**Module 7: Exercise programs on Strings**

- a) Implement Python Script to perform various operations on string using string libraries.
- b) Implement Python Script to check given string is palindrome or not.
- c) Implement python script to accept line of text and find the number of characters, number of vowels and number of blank spaces in it.
- d) Implement python script that takes a list of words and returns the length of the longest one.

**Module 8: Exercise programs on Regular Expressions**

- a) Write a Python script to check that a string contains only a certain set of characters (in this case a-z, A-Z and 0-9).
- b) Write a Python script to check whether password is valid or not.

Conditions for a valid password are:

Should have at least one number.

Should have at least one uppercase and one lowercase character.

Should have at least one special symbol.

Should be between 6 to 20 characters long.

**Module 9: Exercise programs on Object Oriented Programming**

- a) Write a Python script to create and access class variables and methods.
- b) Write a Python script to implement method overloading.
- c) Write a Python script to implement single inheritance.
- d) Write a Python script to implement method overriding.

**Module 10: Exercise programs on Python Libraries – Numpy , Pandas , Matplotlib**

- a) Write a NumPy program to generate a matrix product of two arrays.
- b) Write a NumPy program to create a random array with 1000 elements and compute the average, variance, standard deviation of the array elements.
- c) Demonstrate how to download dataset and how to create DataFrame
  - i. Write a Pandas program to get the first 3 rows of a DataFrame
  - ii. Write a Pandas program to select the specified columns and rows from a given data frame.
  - iii. Write a Pandas program to select the rows where the score is missing, i.e. is NaN.
  - iv. Write a Pandas program to insert a new column in existing DataFrame.
- d) Write a Python programming to display a bar chart using different color for each bar.
- e) Write a Python programming to create a pie chart with a title.

B.Tech. (II Sem.)

20CS53 - DATA STRUCTURES LAB

L	T	P	Cr.
0	0	3	1.5

Pre-requisite : Programming Language

### Course Educational Objectives:

The objective of this course is to make students familiar with writing algorithms to implement different data structures like stacks, queues, trees and graphs, and various sorting techniques.

**Course Outcomes (COs):** *At the end of this course, the student will be able to*

- CO 1: Implement Linear Data Structures using array and Linked list. (**Apply - L3**)
- CO 2: Implement Various Sorting Techniques. (**Apply - L3**)
- CO 3: Implement Non-Linear Data Structure such as Trees & Graphs. (**Apply - L3**)
- CO 4: Improve individual / teamwork skills, communication & report writing skills with ethical values.

### I) Exercise Programs on List ADT

- a) Implementation of List using Arrays.
- b) Implementation of List using Linked List.

### II) Exercise Programs on Stacks & Queue ADT

- a) Implementation of Stack Operations using Arrays.
- b) Implementation of Stack Operations using Linked List.
- c) Implementation of Queue Operations using Arrays.
- d) Implementation of Queue Operations using Linked List.

### III) Exercise Programs on Stack Applications

- a) Conversion of Infix Expression to postfix Expression.
- b) Conversion of Infix Expression to prefix Expression.
- c) Evaluation of Postfix Expression
- d) Implementation of Balancing Symbols.

### IV) Exercise Programs on Types of Queues

- a) Implementation of Circular Queues Linked List.
- b) Implementation of Double Ended Queue using Arrays.
- c) Implementation of Double Ended Queue using Linked List.

**V) Exercise Programs on Sorting Techniques.**

- a) Implementation of Insertion Sort and
- b) Implementation of Selection Sort.
- c) Implementation of Merge Sort.
- d) Implementation of Quick Sort.
- e) Implementation of Bubble Sort.
- f) Implementation of Heap Sort.

**VI) Exercise Programs on Trees**

- a) Implementation of Binary Tree Traversals.
- b) Implementation of Binary Search Tree Operations.

**VII) Exercise Programs on Graph Traversal Techniques.**

- a) Breadth First Search (BFS)
- b) Depth First Search (DFS)

L	T	P	Cr.
0	0	2	1

**Pre-requisites :** Nil

**Course Educational Objective:** The main objective of this course is to familiarize with the Unix/Linux command line and running simple commands and concept of environment variables and with the simple use of environment variables.

**Course Outcomes (COs):** *At the end of this course, the student will be able to*

- CO 1: Understand the basic unix/linux commands. (**Understand - L2**)
- CO 2: Learn importance of shell scripting. (**Understand - L2**)
- CO 3: Apply shell programming to various files. (**Apply - L3**)
- CO 4: Improve individual / teamwork skills, communication & report writing skills with ethical values.

### Module – I: Basic Linux Commands

Study of Unix/Linux general purpose utility command list obtained from (man, who, cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown) commands, study of vi editor, study of Unix/Linux file system

### Module – II: Introduction to Shell

Introduction to Shell, Shell responsibilities, running a shell script. Variables, passing arguments, Basic Operators, Basic String Operations, Decision Making, Loops, Arrays, Arrays – Comparison, Shell functions.

### Module – III: Advanced Shell

Special Variables, Bash trap command, File Testing, Input Parameter Parsing, Pipelines, Process Substitution, Regular Expressions, **Special Commands:** sed, awk, grep, sort

### Example Programs:

1. Use of Basic UNIX Shell Commands: ls, mkdir, rmdir, cd, cat, touch, file, wc, sort, cut, grep, dd, df, space, du, ulimit
2. Commands related to inode, I/O redirection and piping, process control commands, mails.
3. Shell Programming: Shell script exercises based on following:
  - (i) Interactive shell scripts (ii) Positional parameters (iii) Arithmetic (iv) if-then-fi, if-then- else-fi, nested if-else (v) Logical operators (vi) else + if equals elif, case structure (vii) while, until, for loops, use of break
4. Write a shell script to create a file. Follow the instructions

- (i) Input a page profile to yourself, copy it into other existing file
- (ii) Start printing file at certain line
- (iii) Print all the difference between two file, copy the two files.
- (iv) Print lines matching certain word pattern.

5. Write shell script for-

- (i) Showing the count of users logged in,
- (ii) Printing Column list of files in your home directory
- (iii) Listing your job with below normal priority
- (IV) Continue running your job after logging out.

6. Write a shell script to change data format. Show the time taken in execution of this script.

7. Write a shell script to print files names in a directory showing date of creation & serialnumber of the file.

8. Write a shell script to count lines, words, and characters in its input (do not use wc).

**Reference books:**

1. Learning the bash Shell, 3rd Edition by Cameron Newham, Publisher(s): O'Reilly Media, Inc., ISBN: 9780596009656
2. UNIX and Shell Programming by Behrouz A. Forouzan, Richard F. Gilberg Publisher: Thomson Press (India) Ltd, ISBN: 9788131503256, 9788131503256
3. Shell Scripting: Expert Recipes for Linux, Bash, and More by Steve Parker

L	T	P	Cr.
3	0	0	3

Pre-requisite : Nil

**Course Educational Objective:** The objective of this course is to provide students with the foundations and applications of probabilistic and statistical methods mainly used in varied applications in engineering and science.

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

- CO1:** Understand various probabilistic situations using the various laws of probability and random variables (**Understand - L2**)
- CO2:** Apply probability distributions like Binomial, Poisson, Normal and Exponential distributions in solving engineering problems (**Apply - L3**)
- CO3:** Calculate the standard error of sampling distribution and confidence intervals for parameters like mean and proportion based on the sample data. (**Apply - L3**)
- CO4:** Analyze the data scientifically with the appropriate statistical methodologies to apply the suitable test of hypothesis (**Analyze - L4**)
- CO5:** Construct the regression lines to predict the dependent variables and calculate the Correlation Coefficient for a bivariate statistical data. (**Apply - L3**)

## UNIT – I

### Probability and Random variables

Probability, Sample space and events, Additive Rule, Conditional probability, Multiplicative rule, Baye's theorem.

Random variables – Discrete and continuous Random Variables, distribution function. Mathematical Expectation of one-dimensional Random Variable.

## UNIT – II

### Probability Distributions

Binomial distribution , Poisson distribution , Poisson approximation to Binomial distribution, Exponential distribution, Normal distribution , Normal approximation to Binomial distribution

## UNIT – III

### Sampling distribution & Estimation

Population, sample, parameter, statistic, sampling distribution, Standard error, Types of sampling, Sampling distribution of means and sampling distribution of variance, Parameter estimations –point estimation and interval estimation for mean and proportions.

## UNIT – IV

### Tests of Hypothesis

Hypothesis, Null and Alternate Hypothesis, , Type I and Type II errors, level of significance. Z-test for means and proportions, t-test for single mean, difference of means, paired t-test, F-test for equality of population variances,  $\chi^2$  - test for goodness of fit and independence of attributes.



**UNIT – V**

**Correlation & Regression**

Karl Pearson's coefficient of correlation, linear Regression, Regression lines, Regression coefficients, Spearman's Rank correlation coefficient, Spearman's Rank correlation for repeated ranks.

**TEXTBOOKS:**

1. Jay L.Devore "Probability and Statistics for engineering and the sciences." , 8th edition, Cengage Learning india, 2012.
2. S.C.Gupta, V.K.Kapoor, "Fundamentals of Mathematical Statistics", 11thEdition, Sultan Chand and sons, New Delhi,2014

**REFERENCE BOOKS:**

1. Miller & Freund's "Probability and Statistics for Engineers",8th edition. PHI, New Delhi,2011.
2. B.V. Ramana, "Higher Engineering Mathematics", 1st Edition, TMH, New Delhi, 2010

B.Tech. (III Sem.)

**20CS04 - DISCRETE MATHEMATICAL  
STRUCTURES**

L	T	P	Cr.
3	0	0	3

Pre-requisite : Basic mathematical knowledge

**Course Educational Objective:** The objective of the course is to perform the operations associated with relations and functions. Relate practical examples to the functions and relations and interpret the associated operations and terminology used in the context. Use formal logic proofs and/or informal but rigorous logical reasoning to, for example, predict the behavior of software or to solve problems such as puzzles.

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

- CO1:** Construct mathematical arguments using logical connectives and quantifiers and verify them. **(Apply -L3)**
- CO2:** Demonstrate the basic terminology of functions, relations, lattices and their operations. **(Understand - L2)**
- CO3:** Apply the properties of graphs to solve the graph theory problems in Computer science. **(Apply- L3)**
- CO4:** Illustrate the basic principles/techniques to solve different algebraic structures & combinatorial problems. **(Understand- L2)**
- CO5:** Solve linear recurrence relations by recognizing homogeneity using constant coefficients, characteristic roots and Generating functions. **(Apply – L3)**

**UNIT – I: Mathematical Logic**

**Propositional logic and Predicate Calculus:** Statements and Notations, Connectives, Truth Tables, Tautologies, Equivalence of Formulas, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicative Logic, Statement Functions, Variables and Quantifiers.

**UNIT – II: Sets, Relations & Functions**

Introduction to Sets, representation of Sets, Operation on Sets, Properties of Binary Relations, Relation Matrix, Operations on Relations, Transitive Closure, Equivalence Relation, Compatibility and Partial Ordering Relations, Hasse Diagrams, Lattices: LUB, GLB.

**Functions:** Bijective Functions, Composition of Functions, Inverse Functions.

**UNIT – III: Graph Theory I & II**

Basic Concepts of Graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Eulerian and Hamiltonian Graphs.

**Graph Theory II:** Planar Graphs, Euler's Formula, Graph Coloring, Chromatic Number, Graph Traversals: BFS, DFS. Trees: Spanning Trees: Properties, Algorithms for Minimum cost Spanning Trees

**UNIT – IV: Algebraic Structures & Combinatorics**

Algebraic Systems with one Binary Operation, Properties of Binary operations, Semi groups and Monoids: Homomorphism of Semi groups and Monoids, Group, Abelian group, Sub Groups, Lagrange's Theorem

**Combinatorics:** Basic of Counting, Permutations, Combinations, Combinations with repetition Pigeonhole Principle and its Applications, Principle of inclusion-exclusion.

### **UNIT – V: Recurrence Relation**

Generating Function of Sequences, Calculating Coefficient of Generating Functions, Recurrence Relations, solving linear homogeneous recurrence Relations by substitution, generating functions and The Method of Characteristic Roots.

#### **TEXTBOOKS:**

1. Tremblay, Manohar, “Discrete Mathematical Structures with Applications to ComputerScience”, TMH Publications, 2008

#### **REFERENCE BOOKS:**

1. Chandrasekaran,Umaparvathi,DiscreteMathematics,PHI, 2010.
2. Ralph. P.Grimaldi, Ramana, Discrete and Combinational Mathematics,Pearson,5<sup>th</sup> edition.
3. <https://nptel.ac.in/courses/106/106/106106183/>

B.Tech. (III Sem.)

20CS07 - DATABASE MANAGEMENT SYSTEMS

L	T	P	Cr.
3	0	0	3

Pre-requisite : Data Structures

**Course Educational Objective:** The Objective of this course is to know about basic concepts of DBMS, Database Languages, Database Design, Normalization Process, Transaction Processing, Indexing, and Interfacing with NOSQL using MongoDB.

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

- CO1:** State the Basic Components of Database Management System and data modelling using Entity-Relationship Diagrams. **(Understand -L2)**
- CO2:** Examine the relational model using Structured Query Language (SQL). **(Apply- L3)**
- CO3:** Employ principles of normalization for effective database design. **(Apply- L3)**
- CO4:** Demonstrate the necessity of transaction processing, Concurrency control mechanisms and recovery strategies in DBMS. **(Understand- L2)**
- CO5:** Describe file organization, indexing techniques and the competency in selecting NoSQL Database. **(Understand- L2)**

## UNIT – I

**Introduction:** An overview of Database Management System, Database System Vs File System, Database System Concepts and Three Schema Architecture, Data Models, Database Schema and Instances, Data Independence, Database Languages, Database Structure.

**Data Modelling using the Entity Relationship Model:** ER model concepts, Notation for ER Diagram, Mapping Constraints, Keys, Concepts of Super Key, Candidate Key, Primary Key, Generalization, Aggregation, Reduction of an ER Diagrams to Tables, Relationships of Higher Degree.

## UNIT – II

**Relational Data Model and Language:** Relational Data Model Concepts, Integrity Constraints: Entity Integrity, Referential Integrity, Key Constraints, Domain Constraints, and Relational Algebra.

**Introduction to SQL:** Characteristics of SQL, Advantage of SQL. SQL Data types and Literals, Insert, Update and Delete Operations, Tables, Views and Indexes, Nested Queries, Aggregate Functions, Joins, Unions, Intersection, Minus, Cursors in SQL, Triggers in SQL.

## UNIT – III

**Normalization:** Functional Dependencies, Normal Forms - First, Second, Third Normal Forms, BCNF, Inclusion Dependencies, Loss Less Join Decompositions, Multi Valued Dependencies, Fourth Normal Form, Join Dependencies and Fifth Normal Form.

## UNIT – IV

**Transaction Processing Concepts:** Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializability, Recoverability, Deadlock Handling.

**Concurrency Control Techniques:** Concurrency Control, Locking Techniques for Concurrency Control, Time Stamping Protocols for Concurrency Control, Validation Based Protocol, Multiple Granularity, Recovery with Concurrent Transactions.

**Crash Recovery:** Log Based Recovery, Checkpoints, ARIES Algorithm

## **UNIT – V**

**Physical Database Design:** Storage and file structure, indexed files, hashed files, B+ trees, files with dense index; files with variable length records.

**Interfacing And Interacting With NoSQL:** Introduction to NoSQL, Storing and Accessing Data, Storing Data In and Accessing Data from MongoDB, Querying MongoDB.

### **TEXTBOOKS:**

1. Henry F. Korth, Abraham Silberschatz, S.Sudarshan, “Database System Concepts”, McGrawHill, 6th edition, 2009.
2. Shashank Tiwari, “ ProfessionalNoSql”, John Wiely& Sons, 2011.

### **REFERENCE BOOKS:**

1. Raghu Ramakrishnan, Johannes Gehrke, —Database Management Systemll, McGrawHill, 3rd edition, 2000.
2. Date C J, —An Introduction to Database System, Pearson Education, 8th edition, 2003.
3. RamezElmasri, ShamkanthB.Navathe, “Fundamentals of Database Systems”, AddisonWesley, 6th edition, 2010.

L	T	P	Cr.
3	0	0	3

Pre-requisite : IT Workshop

**Course Educational Objective:** The objective of the course is to learn about the functional blocks and data representation of computer system, and understands the design principles of processor and organization and management of memory and peripheral devices.

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

- CO1:** Evaluate digital number systems and use Boolean algebra theorems, Properties and Canonical forms for digital logic circuit design. (**Understand- L2**)
- CO2:** Design Combinational logic circuits and Sequential logic circuits. (**Apply- L3**)
- CO3:** Understand computer architecture and Data representation to perform computer arithmetic operations. (**Understand- L2**)
- CO4:** Illustrate the design principles of control unit and pipelining. (**Understand- L2**)
- CO5:** Analyze the memory hierarchy in a computer system. (**Understand- L2**)

## UNIT-I

### Number Systems, logic Gates and Boolean algebra

**Number Systems:** Binary, Octal, Decimal, Hexadecimal Number Systems. Conversion of Numbers from One, Radix to another Radix,  $r$ 's Complement and  $(r-1)$ 's Complements, Binary Codes, Basic Gates: NOT, AND, OR, Universal Gates: NAND, NOR, Special Gates: Ex-OR and Ex-NOR Gates

### Boolean algebra

Fundamental postulates of Boolean algebra, Basic theorems and properties, Complement and Dual of Logical Expressions, SOP, POS, Minimizations of Logic Functions Using Boolean Theorems, Karnaugh Map Method (K-Map): Minimization of Boolean Functions maximum up to Four Variables, Simplifications with Don't Care Conditions Using K-Map.

## UNIT-II

### Combinational Logic Circuits & Sequential Logic Circuits

Design of Half Adder, Full Adder, Half Subtractor, Full Subtractor, Ripple Carry Adder. Design of Decoders, Encoders, Multiplexers, De-multiplexers, Priority Encoder,

### Sequential Logic Circuits:

Latch and Flip-Flop, RS- Latch Using NAND and NOR Gates, RS, JK, T and D Flip-flops, Truth and Excitation Tables, Conversion of Flip Flops, Master-Slave Flip-flops. Registers and Counters.

## UNIT-III

**Functional Blocks of a Computer:** CPU, Memory, Input-Output Subsystems, Control Unit. Instruction Set Architecture of A CPU—Registers, Instruction Execution Cycle, RTL Interpretation of Instructions, Addressing Modes.

**Data Representation:** Fixed and Floating-Point Representations. Computer Arithmetic – Integer Addition and Subtraction, Multiplication – Shift-And Add, Booth Multiplier.

#### **UNIT-IV**

**CPU control unit design:** Hardwired and micro-programmed design approaches. Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards.

#### **UNIT-V**

**Memory system design:** Memory hierarchy, Cache Memory and Main Memory.

**Peripheral devices and their characteristics:** Input-output subsystems, I/O device interface, I/O transfers—program controlled, interrupt driven and DMA.

#### **TEXTBOOKS:**

1. Morris Mano, Michael D Ciletti ,Digital Design, 4/e, 2008 , PEA.
2. CarlHamacher, ZvonksVranesic, SafeaZaky, “Computer Organization”, TMH publications

#### **REFERENCE BOOKS:**

1. M.Morris Mano, “Computer Systems Architecture”, Pearson Education publishers.[units-1,2]
2. Leach, Malvino, Saha, “Digital Logic Design”, TMH,2006
3. R.P. Jain, “Modern Digital Electronics”, TMH ,2011
4. A. Anand Kumar, “Switching Theory and Logic Design”, Prentice-Hall Of India Pvt. Limited, 2010.
5. Kohavi, Jha, Cambridge,”Switching and Finite Automata Theory”, 3/e
6. William Stallings,“Computer Organization and Architecture”, Pearson/PHI publishers, Sixth Edition, 2015.

L	T	P	Cr.
3	0	0	3

Pre-requisite : Programming for Problem Solving using C.

**Course Educational Objective:** The objective of the course is to learn the constructs of the Java programming language along with built-in facilities to create different applications such as console & graphical user interfaces. In the process of learning the language, they will be applying knowledge of object-oriented programming; they will get the fundamental knowledge reason collection framework.

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

- CO 1 Demonstrate the fundamentals of object-oriented programming and basic building blocks of Java. (**Understand- L2**)
- CO 2 Apply object-oriented programming principles for the development of reusable applications. (**Apply - L3**)
- CO 3 Understand the importance of abstraction, user defined package creation and handling different exceptions. (**Understand- L2**)
- CO 4 Develop multitasking applications using JAVA multithreaded programming and perform different operations upon various data structures by using collection framework. (**Apply – L3**)
- CO 5 Develop GUI applications using AWT (Abstract Window Toolkit). (**Apply- L3**)

## UNIT-I

**Introduction to OOP:** Programming paradigms, procedural programming language versus object-oriented language, principles of OOP.

**Introduction to JAVA:** Data types, variables, keywords, operators, and control statements.

**Introduction to Classes and Object:** Class definition, variables, and methods. Declaring Objects, Constructors, and this keyword.

## UNIT-2

**Classes and objects:** overloading methods and constructors, parameter passing, returning objects, recursion. Access control, nested and inner classes, final and static keyword, variable and command-line arguments.

**Inheritance and polymorphism:** Inheritance, types of inheritance, super keyword, polymorphism (overloading & overriding), dynamic method dispatch, abstract class, using final with inheritance.

**String handling classes:** String, StringBuffer, StringTokenizer.

## UNIT-3

**Interfaces and packages:** Interface methods, inheritance in interfaces. API : The built-in JAVA packages and creating and managing user defined packages, importance of CLASSPATH.



**Exception Handling:** Exception hierarchy, importance of try, catch, throw, throws and finally. Block creation of user-defined exceptions, Assertions.

#### **UNIT-4**

**Multithreading:** Introduction, thread life cycle, creation of threads, naming a thread, joining a thread, thread priorities, daemon thread, thread pool, thread group, thread synchronization, Inter-thread communication.

**Collection Framework:** Introduction, generics, collection framework hierarchy, list, set, queue, and map.

#### **UNIT-5**

**AWT:** AWT hierarchy, components and containers, Button, Label, Text Field, Checkbox, Choice, List, Canvas, Scrollbar, Menu Item & Menu, Container class, Layout managers (Border Layout, Flow Layout, Grid Layout, Card Layout).

**Event handling:** Event delegation model, event classes (Action Event, Mouse Event, Key Event, Window Event), listener interfaces (Action Listener, Mouse Listener and MouseMotionListener, Key Listener, Window Listener), adapter classes, close AWT window.

#### **TEXTBOOKS:**

1. Herbert Schildt, "Java: The complete reference", TMH Publications, 7th edition, 2006.
2. Cay S. Horstmann, "Core Java Volume I – Fundamentals", Pearson, Eleventh edition, 2018.

#### **REFERENCE BOOKS:**

1. Dr.R.NageswaraRao, "Core JAVA: An Integrated Approach", Dreamtech Press, 1st Edition 2008.
2. E. Balaguruswamy, "Programming with JAVA", TMH Publications, 2nd Edition, 2000.
3. Patrick Niemeyer & Jonathan Knudsen, "Learning Java", O'REILLY Publications, 3rd Edition, 2005.
4. Benjamin J Evans & David Flanagan, "Java-in a Nutshell – A desktop quick reference", O'REILLY Publications, 6th Edition, 2014.



B.Tech. (III Sem.)

**20CS56 - DATABASE MANAGEMENT SYSTEMS  
LAB**

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Programming language, Discrete Mathematical Structures, and Data Structures.

**Course Educational Objective:** The objective of this lab is to provide a strong formal foundation in database concepts, technology, and practice to the participants to groom them into well-informed database application developers.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Create & manipulate the relational database using SQL. **(Apply- L3)**
- CO2:** Implement Views, procedures, triggers, and cursors on relational database. **(Apply- L3)**
- CO3:** Create Unstructured Databases using MongoDB. **(Apply- L3)**
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Introduction:** Language basics and example queries **(one or two weeks)**

1) Create a table STUDENT with appropriate data types and perform the following queries. Attributes are Roll number, student name, date of birth, branch and year of study.

- a) Insert 5 to 10 rows in a table?
- b) List all the students of all branches
- c) List student names whose name starts with 's'.
- d) List student names whose name contains 's' as third literal.
- e) List student names whose contains two 's' anywhere in the name
- f) List students whose branch is NULL.
- g) List students of CSE & ECE who born after 1980.
- h) List all students in reverse order of their names.
- i) Delete students of any branch whose name starts with 's'.
- j) Update the branch of CSE students to ECE.
- k) Display student name padded with '\*' after the name of all the students.

2) Create the following tables based on the above Schema Diagram with appropriate data types

and constraints and perform the following queries.

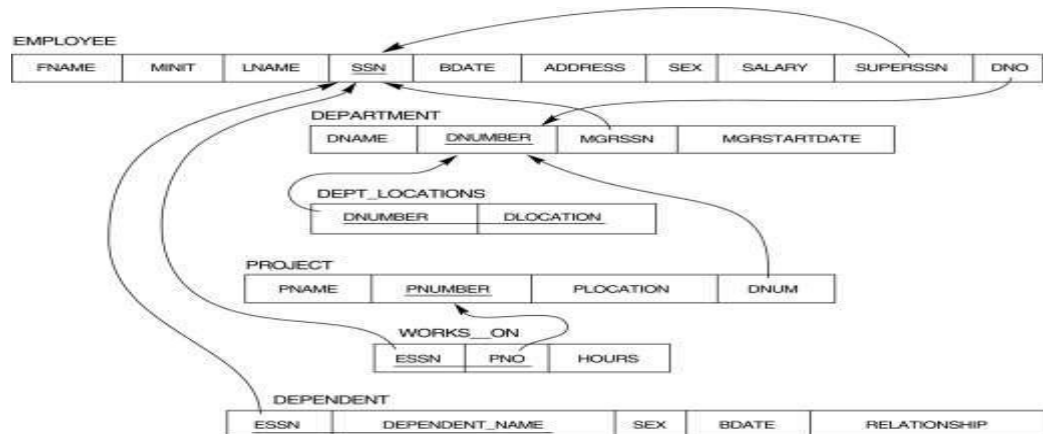
SAILORS (Sailid, Salname, Rating, Age)

RESERVES (Sailid, boatid, Day)

BOATS (Boatid, Boat-name, Color)

- a) Insert 5 to 10 rows in all tables?
- b) Find the name of sailors who reserved boat number 3.
- c) Find the name of sailors who reserved green boat.
- d) Find the colors of boats reserved by Ramesh.
- e) Find the names of sailors who have reserved at least one boat.

- f) Find the allsailid of sailors who have a rating of 10 or have reserved boated 104.
  - g) Find the Sailid's of sailors with age over 20 who have not registered a red boat.
  - h) Find the names of sailors who have reserved a red or green boat.
  - i) Find sailors whose rating is better than some sailor called Salvador.
  - j) Find the names of sailors who are older than the oldest sailor with a rating of 10.
- 3) Schema Diagram for the rest of the SQL and PLSQL Programs.



Create the following tables based on the above Schema Diagram with appropriate data types and constraints.

EMPLOYEE (Fname, Mname, Lname, SSN, Bdate, Address, Gender, Salary, SuperSSN,Dno)

DEPARTMENT (Dnumber, Dname, MgrSSN, Mgrstartdate)

DEPENDENT (ESSN, Dependent\_Name, Gender, Bdate, Relationship)

- a) Insert 5 to 10 rows into all the tables.
- b) Display all employees' names along with their department names.
- c) Display all employees' names along with their dependent details.
- d) Display name and address of all employees who work for Research department.
- e) List the names of all employees with two or more dependents.
- f) List the names of employee who have no dependents.
- g) List the names of employees who have at least one dependent.
- h) List the names of the employees along with names of their supervisors using aliases.
- i) Display name of the department and name of manager for all the departments.
- j) Display the name of each employee who has a dependent with the same first name and gender as the employee.

4) Create the following tables based on the above Schema Diagram with appropriate data types and constraints in addition to the tables in Experiment 2.

DEPT\_LOCATIONS (Dnumber, Dlocation)

PROJECT (Pname, Pnumber, Plocation,

Dnum) WORKS\_ON (ESSN, Pno, Hours)

- a) Insert 5 to 10 rows into all the tables.

- b) Find the names of the employees who work on all the projects controlled by the department Research.
- c) List the project number, name and no. Of employees who work on that project for all the projects.
- d) List the names of all the projects controlled by the departments department wise.
- e) Retrieve the names of employees who work on all projects that John works on.
- f) List the project numbers for projects that involve an employee either as worker or as a manager of the department that controls the project.
- g) List the names of all employees in one department who work more than 10 hours on one specific project.
- h) For each project, list the project name and total hours (by all employees) spent on that project.
- i) Retrieve the names of all employees who work on every project.
- j) Retrieve the names of all employees who do not work on any project.

5) Create a view that has project name, controlling department name, number of employees and total hours worked on the project for each project with more than one employee working on it.

- a) List the projects that are controlled by one department from this view.
- b) List the managers of the controlling departments for all the projects.
- c) Demonstrate one update operation on this view.
- d) List the Location of the controlling departments for all the projects.
- e) Retrieve the data from the view.

6) Create a view emp from employee such that it contains only emp\_no, emp\_name and department.

7) Create a view dept from department with only dept\_no and location.

8) Create a view that contains the details of employees who are managers only.

9) Write a procedure to check whether the given number is Armstrong or not.

10) Write a procedure which accept the account number of a customer and retrieve the balance.

11) Write a procedure which accepts the student number and displays the department in which he belongs to.

12) Create a cursor to modify the salary of all employees belonging to 'Research' department by 150%.

13) Consider the college database. Retrieve all students who have registered for a specific course and store their details into another table using Cursors.

14) Write an update trigger on Account table. The system should keep track of the records that are being updated.

15) Create NoSQL database for a sample application and perform CRUD operations

### **Design Database for any one of the following Case Studies**

#### **Case Study1 : Hospital Management System**

Aim: XYZ hospital is a multi-specialty hospital that includes a number of departments, rooms, doctors, nurses, compounders, and other staff working in the hospital. Patients having different kinds of ailments come to the hospital and get checkup done from the concerned doctors. If required they are admitted in the hospital and discharged after treatment. The aim of this case study is to design and develop a database for the hospital to maintain the records of various departments, rooms, and doctors in the hospital. It also maintains records of the regular patients, patients admitted in the hospital, the checkup of patients done by the doctors, the patients that have been operated, and patients discharged from the hospital.

Description: In hospital, there are many departments like Orthopedic, Pathology, Emergency, Dental, Gynecology, Anesthetics, I.C.U., Blood Bank, Operation Theater, Laboratory, M.R.I., Neurology, Cardiology, Cancer Department, Corpse, etc. There is an OPD where patients come and get a card (that is, entry card of the patient) for check up from the concerned doctor. After making entry in the card, they go to the concerned doctor's room and the doctor checks up their ailments. According to the ailments, the doctor either prescribes medicine or admits the patient in the concerned department. The patient may choose either private or general room according to his/her need. But before getting admission in the hospital, the patient has to fulfill certain formalities of the hospital like room charges, etc. After the treatment is completed, the doctor discharges the patient. Before discharging from the hospital, the patient again has to complete certain formalities of the hospital like balance charges, test charges, operation charges (if any), blood charges, doctors' charges, etc. Next we talk about the doctors of the hospital. There are two types of the doctors in the hospital, namely, regular doctors and call on doctors. Regular doctors are those doctors who come to the hospital daily. Calls on doctors are those doctors who are called by the hospital if the concerned doctor is not available.

#### Table Description:

Following are the tables along with constraints used in Hospital Management database

1. DEPARTMENT: This table consists of details about the various departments in the hospital. The information stored in this table includes department name, department location, and facilities available in that department.

Constraint: Department name will be unique for each department.

2. ALL\_DOCTORS: This table stores information about all the doctors working for the hospital and the departments they are associated with. Each doctor is given an identity number starting with DR or DC prefixes only.

Constraint: Identity number is unique for each doctor and the corresponding department should exist in DEPARTMENT table.

3. DOC\_REG: This table stores details of regular doctors working in the hospital. Doctors are referred to by their doctor number. This table also stores personal details of doctors like name, qualification, address, phone number, salary, date of joining, etc. Constraint: Doctor's number entered should contain DR only as a prefix and must exist in ALL\_DOCTORS table.

4. DOC\_ON\_CALL: This table stores details of doctors called by hospital when additional doctors are required. Doctors are referred to by their doctor number. Other personal details like name, qualification, fees per call, payment due, address, phone number, etc., are also stored.

Constraint: Doctor's number entered should contain DC only as a prefix and must exist in ALL\_DOCTORS table.

5. PAT\_ENTRY: The record in this table is created when any patient arrives in the hospital for a checkup. When patient arrives, a patient number is generated which acts as a primary key. Other details like name, age, sex, address, city, phone number, entry date, name of the doctor referred to, diagnosis, and department name are also stored. After storing the necessary details patient is sent to the doctor for checkup.

Constraint: Patient number should begin with prefix PT. Sex should be M or F only. Doctor's name and department referred must exist.

6. PAT\_CHKUP: This table stores the details about the patients who get treatment from the doctor referred to. Details like patient number from patient entry table, doctor number, date of checkup, diagnosis, and treatment are stored. One more field status is used to indicate whether patient is admitted, referred for operation or is a regular patient to the hospital. If patient is admitted, further details are stored in PAT\_ADMIT table. If patient is referred for operation, the further details are stored in PAT\_OPR table and if patient is a regular patient to the hospital, the further details are stored in PAT\_REG table.

Constraint: Patient number should exist in PAT\_ENTRY table and it should be unique.

7. PAT\_ADMIT: When patient is admitted, his/her related details are stored in this table. Information stored includes patient number, advance payment, mode of payment, room number, department, date of admission, initial condition, diagnosis, treatment, number of the doctor under whom treatment is done, attendant name, etc.

Constraint: Patient number should exist in PAT\_ENTRY table. Department, doctor number, room number must be valid.

8. PAT\_DIS: An entry is made in this table whenever a patient gets discharged from the hospital. Each entry includes details like patient number, treatment given, treatment advice, payment made, mode of payment, date of discharge, etc.

Constraint: Patient number should exist in PAT\_ENTRY table.

9. PAT\_REG: Details of regular patients are stored in this table. Information stored includes date of visit, diagnosis, treatment, medicine recommended, status of treatment, etc.

Constraint: Patient number should exist in patient entry table. There can be multiple entries of one patient as patient might be visiting hospital repeatedly for checkup and there will be entry for patient's each visit.

10. PAT\_OPR: If patient is operated in the hospital, his/her details are stored in this table. Information stored includes patient number, date of admission, date of operation, number of the doctor who conducted the operation, number of the operation theater in which operation was carried out, type of operation, patient's condition before and after operation, treatment advice, etc.

Constraint: Patient number should exist in PAT\_ENTRY table. Department, doctor number should exist or should be valid.

11. ROOM\_DETAILS: It contains details of all rooms in the hospital. The details stored in this table include room number, room type (general or private), status (whether occupied or not), if occupied, then patient number, patient name, charges per day, etc.

Constraint: Room number should be unique. Room type can only be G or P and status can only be Y or N.

### **Case Study2 :Railway Reservation**

**Aim:** The railway reservation system facilitates the passengers to enquire about the trains available on the basis of source and destination, booking and cancellation of tickets, enquire about the status of the booked ticket, etc. The aim of case study is to design and develop a database maintaining the records of different trains, train status, and passengers. The record of train includes its number, name, source, destination, and days on which it is available, whereas record of train status includes dates for which tickets can be booked, total number of seats available, and number of seats already booked. The database has been developed and tested on the Oracle.

#### Description:

Passengers can book their tickets for the train in which seats are available. For this, passenger must provide the desired train number and the date for which ticket is to be booked. Before booking a ticket for a passenger, the validity of train number and booking date is checked. Once the train number and booking date are validated, it is checked whether the seat is available. If yes, the ticket is booked with confirm status and corresponding ticket ID is generated which is stored along with other details of the passenger. After all the available tickets are booked, certain numbers of tickets are booked with waiting status. If waiting lot is also finished, then tickets are not booked and a message of non-availability of seats is displayed. The ticket once booked can be cancelled at any time. For this, the passenger must provide the ticket ID (the unique key). The ticket ID is searched, and the corresponding record is deleted. With this, the first ticket with waiting status also gets confirmed.

#### List of Assumption

Since the reservation system is very large in reality, it is not feasible to develop the case study to that extent and prepare documentation at that level. Therefore, a small sample case study has been created to demonstrate the working of the reservation system. To implement this sample case study, some assumptions have been made, which are as follows:

1. The number of trains has been restricted to 5.
2. The booking is open only for next seven days from the current date.
3. Only two categories of tickets can be booked, namely, AC and General.
4. The total number of tickets that can be booked in each category (AC and General) is 10.
5. The total number of tickets that can be given the status of waiting is 2.
6. The in-between stoppage stations and their bookings are not considered.

#### Description of Tables and Procedures

Tables and procedures that will be created are as follows:

1. **TrainList:** This table consists of details about all the available trains. The information stored in this table includes train number, train name, source, destination, fair for AC ticket, fair for general ticket, and weekdays on which train is available.

**Constraint:** The train number is unique.

2. **Train\_Status:** This table consists of details about the dates on which ticket can be booked for a train and the status of the availability of tickets. The information stored in this table includes



train number, train date, total number of AC seats, total number of general seats, number of AC seats booked, and number of general seats booked.

Constraint: Train number should exist in TrainList table.

3. Passenger: This table consists of details about the booked tickets. The information stored in this table includes ticket ID, train number, date for which ticket is booked, name, age, sex and address of the passenger, status of reservation (either confirmed or waiting), and category for which ticket is booked.

Constraint: Ticket ID is unique and the train number should exist in TrainList table.

4.Booking: In this procedure, the train number, train date, and category is read from the passenger. On the basis of the values provided by the passenger, corresponding record is retrieved from the Train\_Status table. If the desired category is AC, then total number of AC seats and number of booked AC seats are compared in order to find whether ticket can be booked or not. Similarly, it can be checked for the general category. If ticket can be booked, then passenger details are read and stored in the Passenger table.

5.Cancel: In this procedure, ticket ID is read from the passenger and corresponding record is searched in the Passenger table. If the record exists, it is deleted from the table. After deleting the record (if it is confirmed), first record with waiting status for the same train and same category are searched from the Passenger table and its status is changed to confirm.

### **Case Study3** :Painting Hire Business

#### **System Description:**

A local businesswoman has decided to start her own Internet business, called Masterpieces Ltd, hiring paintings to private individuals and commercial companies. Because of your reputation as a database designer, she has called upon your services to design and implement a database to support her new business. At the initial planning meeting, to discuss the design, the following user requirements were requested. The system must be able to manage the details of customers, paintings and those paintings currently on hire to customers. Customers are categorized as B (bronze), S (silver), G (gold) or P (platinum). These categories entitle a customer to a discount of 0%, 5%, 10% or 15% respectively.

Customers often request paintings by a particular artist or theme (e.g. animal, landscape, seascape, naval, still-life, etc). Over time a customer may hire the same painting more than once.

Each painting is allocated a customer monthly rental price defined by the owner. The owner of the painting is then paid 10% of that customer rental price. Any paintings that are not hired within six months are returned to the owner. However, after three months, an owner may resubmit a returned painting. Each painting can only have one artist associated with it. Several reports are required from the system. Three main ones are:

1. For each customer, a report showing an overview of all the paintings they have hired or are currently hiring
  2. For each artist, a report of all paintings submitted for hire
  3. For each artist, a returns report for those paintings not hired over the past six months
- remember to identify key attributes and any foreign key attributes.

B.Tech. (III Sem.)

**20CS57 - OBJECT ORIENTED PROGRAMMING  
LAB**

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite :** Programming for Problem Solving using C and Data structures

**Course Educational Objective:** The objective of the course is to apply the constructs of Java programming language along with built-in facilities to create different applications such as console & graphical user interfaces. They will be applying knowledge of object-oriented programming, collection framework to perform all operations on data.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Solve basic mathematical problems using fundamentals of Java and its object-oriented principles. (**Apply – L3**)
- CO2:** Implement multithreading and exception handling mechanisms. (**Apply – L3**)
- CO3:** Develop GUI applications and basic data structures using collection framework. (**Apply – L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Introduction:** Language basics and example problems (**one or two Modules**)

- Write a C++ program to display default value of all primitive data type of Java.
- Write a C++ program that display the roots of a quadratic equation  $ax^2+bx=0$ . Calculate the discriminate D and basing on value of D, describe the nature of root.
- Write a C++ Program to find the factorial of list of numbers reading input as command line argument.
- Write a C++ Program to display all prime numbers between two limits.
- Five bikers compete in a race such that they drive at a constant speed, which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all five racers. Take as input the speed of each racer and print back the speed of qualifying racers.
- Write a C++ program to implement Rhombus pattern reading the limit from the user.
- Write a C++ program to calculate gross salary & net salary taking the following data.  
**Input:** empno, empname, basic salary  
**Process:** DA=50% of basic, HRA=25% of basic, CCA=Rs240/-, PF=10% of basic, PT=Rs100/-.
- Write a C++ program to search for an element in a given list of elements using binary search mechanism.

**Module 1:**

- a. Develop a java program to create class, which contains data & methods, create an object to access those members.
- b. Develop a java program, which implements all types of java variables (local, class level: static, instance).
- c. Develop a java program to calculate the sum of diagonal elements of given  $n \times n$  matrix.

**Module 2:**

- a. Develop a java program, which contains both static and non-static methods.
- b. Develop a java program to find area of geometrical figures using method.
- c. Develop a java program to initialize instance variables by using constructors.
- d. Develop a java program, which implements constructor overloading by passing different number of parameters of different types.

**Module 3:**

- a. Develop a java program to count the words, characters in the given line of text.
- b. Develop a java program for sorting a given list of names in ascending order.
- c. Develop a java program that reads a line of integers separated by commas and then displays each integer, find the sum of the integers (using StringTokenizer).
- d. Develop a java program to implement multi-level inheritance.

**Module 4:**

- a. Develop a java program to create and access user-defined package.
- b. Develop a java program to identify the accessibility of a variable by means of different access specifier within and outside the package.
- c. Develop a java program to implement the concept of method overloading.
- d. Develop a java program to implement the concept of method overriding.

**Module 5:**

- a. Develop a java program for abstract class to find areas of different shapes.
- b. Develop a java program to achieve multiple inheritance using interfaces.
- c. Develop a java program to create an interface named Vehicle which contains two abstract methods (Specifications (), Display ()). Provide two classes named Two-wheeler, Four-wheeler that is implemented by that interface.

**Module 6:**

- a. Develop a java program that implements a multi-threaded program, which has three threads. First thread generates a random integer for every 1 second, if the generated integer is even the second thread computes the square of the number and print it. If the generated integer is odd the third thread will print the value of cube of the number.
- b. Develop a java program to identify the use of synchronized blocks, synchronized methods and static synchronized methods in threads concept.
- c. Develop a java program to illustrate the concept of inter thread communication.

**Module 7:**

- a. Develop a java program that creates a user interface to perform integer divisions with possible validations (Divide by Zero, NumberFormatException).
- b. Develop a java program to implement mouse events like mouse pressed, mouse released, and mouse moved by means of adapter classes.

**Module 8:**

- a. Develop a java program that works as a simple calculator. Use a Grid Layout to arrange Buttons for digits and for the + - \* % operations. Add a text field to display the result. Handle any possible exceptions like divide by zero.
- b. Develop a java program to simulate a traffic light, user can select any one of the three buttons with: red, yellow, and green color. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear with the selected button color.

**Module 9:**

- a. Develop a java program to print the collection data by using the following ways  
i) for loop   ii) for-each loop   iii) Iterator   iv) ListIterator
- b. Develop a java program to perform all the operations in Collection interface.

**Module 10:**

- a. Develop a java program to implement and perform all the operations in List, Set Interface.
- b. Develop a java program to implement and perform all the operations in Map interface.

B.Tech. (III Sem.)

20IT53 – R PROGRAMMING LAB

L	T	P	Cr.
0	0	3	1.5

---

**Pre-requisite** : Nil

**Course Educational 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:** At the end of this course,

**CO1:** Implement basic concepts of R programming, and its different module that includes conditional, looping, lists, Strings, Functions, Frames, Arrays, and File programming. (**Understand - L2**)

**CO2:** Implement the concepts of R Script to extract the data from data frames and file operations. (**Apply – L3**)

**CO3:** Implement the various statistical techniques with visualization using R add-on packages. (**Apply – L3**)

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values

**Week 1:**

- Installing R and RStudio
- Basic functionality of R, variable, data types in R

**Week 2:**

- Implement R script to show the usage of various operators available in R language.
- Implement R script to read person's age from keyboard and display whether he is eligible for voting or not.
- Implement R script to find biggest number between two numbers.
- Implement R script to check the given year is leap year or not.

**Week 3:**

- Implement R Script to create a list.
- Implement R Script to access elements in the list.
- Implement R Script to merge two or more lists.
- Implement R Script to perform matrix operation

**Week 4:**

Implement R script to perform following operations:

- various operations on vectors
- Finding the sum and average of given numbers using arrays.
- To display elements of list in reverse order.
- Finding the minimum and maximum elements in the array.

**Week 5:**

- Implement R Script to perform various operations on matrices
- Implement R Script to extract the data from dataframes.
- Write R script to display file contents.
- Write R script to copy file contents from one file to another

**Week 6:**

- a) Write an R script to find basic descriptive statistics using summary(), str(), quartile() function on mtcars datasets.
- b) Write an R script to find subset of dataset by using subset (), aggregate () functions on iris dataset

**Week 7:**

- a) Reading different types of data sets (.txt, .csv) from Web or disk and writing in file in specific disk location.
- b) Reading Excel data sheet in R.
- c) Reading XML dataset in R

**Week 8:**

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

**Week 9:**

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

**Week 10:**

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

**Week 11:**

Introduction to 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 Google Spreadsheets, API and web scraping examples

**REFERENCE BOOKS:**

1. R Cookbook Paperback – 2011 by Teetor Paul O Reilly Publications
2. Beginning R: The Statistical Programming Language by Dr. Mark Gardener, Wiley Publications.
3. R Programming for Dummies by Joris Meys Andrie de Vries, Wiley Publications
4. Hands-On Programming with R by Grolemond, O Reilly Publications
5. Statistical Programming in R by KG Srinivas G.M. Siddesh, ChetanShetty&Sowmya B.J. – 2017 edition
6. R Fundamentals and Programming Techniques, Thomas Lumely.
7. R for Everyone Advanced Analytics and Graphics, Jared P. Lander- Addison Wesley Series.
8. The Art of R Programming, Norman Matloff, Cengage Learning.
9. Maria Dolores Ugarte, Ana F.Militino, AlanT.Arnholt—Probability and Statistics with R, 2<sup>nd</sup> Edition, CRC Press,2016.
10. R-programming for Data science, Roger D.Peng.
11. An Introduction to statistical learning-with applications in R, Trevor Hastie and Rob Tibshirani.

**WEB LINKS:**

1. URL: <https://cran.r-project.org/doc/manuals/r-release/R-intro.pdf> ( Online Resources)
2. <http://nptel.ac.in/courses/106104135/48>
3. <http://nptel.ac.in/courses/110106064/>

**SOFTWARE REQUIREMENTS:**

1. The R statistical software program. Available from: <https://www.r-project.org/>
2. RStudio an Integrated Development Environment (IDE) for R. Available from: <https://www.rstudio.com/>

**B.Tech. (III Sem.)**      **20CSS1–WEB APPLICATION DEVELOPMENT  
USING FULL STACK – MODULE - I  
(FRONT END DEVELOPMENT)**

L	T	P	Cr.
1	0	2	2

**Pre-requisite** : Knowledge of basic Computer hardware & software.

**Course Educational Objective:** The objective of the course is to understand the design of HTML web pages, Styling of HTML pages using CSS, web forms validation using JavaScript and developing responsive web page using JQuery.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Understand the basic design and styling of web pages. (**Understand- L2**)
- CO2:** Understand the DOM of web design, markup language and client-side scripting. (**Understand- L2**)
- CO3:** Understand the responsive web design using DHTML. (**Understand- L2**)
- CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

## Modules

1. Design of web pages (Basic Structure, text formatting tags, heading tags, image tag, marquee tag, hyperlink, paragraph, font, line breaking, types of lists).
2. Creating Web applications using HTML5 (Table tags, Form elements, Frames concept)
3. Styling the web pages using CSS (Types of CSS, CSS Selectors, Properties: Text, backgrounds, font, Links, Borders, Margins, Cell padding, Layouts).
4. Validating the web forms using JavaScript (JavaScript Introduction, JavaScript DOM, Form validation).
5. Handling and validating semi-structured data
  - a. XML document validation using DTD & Schema
  - b. JSON
6. jQuery
  - a. selectors
  - b. validation
  - c. hide
  - d. slide
  - e. fades
  - f. animation

## REFERENCE BOOKS:

1. HTML & CSS: The Complete Reference, 5<sup>th</sup> Edition by Thomas Powell, McGrawHill, 2017.
2. Beginning HTML, XHTML, CSS, and JavaScript by Jon Duckett, Wiley India, 2010.
3. jQuery Cookbook by Cody Lindley, O'Reilly Media, 2009
4. HTML, XHTML, and CSS Bible, 5th Edition by Steven M. Schafer, Wiley India, 2011.
5. Web Development with jQuery by Richard York, Wiley India, 2015.
6. Head first HTML & CSS 2nd Edition by Elisabeth Robson, Eric Freeman, O'Reilly.



B.Tech. (IV Sem.)

**20CS06 - DESIGN AND ANALYSIS OF ALGORITHMS**

L	T	P	Cr.
3	0	0	3

Pre-requisite: Discrete Mathematical Structures and DataStructures.

**Course Educational Objective:** The Objective of the course is to learn various algorithm design techniques and analyze the computing resources of the algorithms, and motivate the students to design new algorithms for various problems

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

- CO1:** Identify the characteristics of an algorithm and analyze its time and space complexity. **(Understand- L2)**
- CO2:** Apply the divide-and-conquer method for solving problems like searching and sorting. **(Apply- L3)**
- CO3:** Design Greedy algorithms for the optimization problems like knapsack problem, minimum cost spanning tree, single source shortest path problem. **(Apply - L3)**
- CO4:** Apply dynamic programming paradigm to solve optimization problems like travelling salesperson problem, 0/1 knapsack problem, Optimal binary search tree. **(Apply - L3)**
- CO5:** Analyze the backtracking and branch-and-bound search methods on optimization problems like N-queens, sum of subsets, 0/1 knapsack, Hamiltonian circuit and so on. **(Apply - L3)**

**UNIT – I**

**Introduction:** Algorithm definition, Specifications, Performance Analysis- Time Complexity, Space Complexity. Asymptotic Notations-Big-Oh, Omega, Theta.

**Divide and Conquer:** General Method, Binary Search, Finding Maximum and Minimum, Merge Sort, Quick sort, closest pair of points.

**UNIT – II**

**The Greedy Method** – General Method, Knapsack Problem, Job sequencing with deadlines, Minimum-cost spanning trees, Optimal storage on tapes, Single source shortest paths, Huffman coding.

**UNIT – III**

**Dynamic Programming** - General method, Multistage graph, All pairs shortest path, Single Source Shortest path, Optimal Binary search trees, 0/1 Knapsack, Reliability design, the travelling salesman problem.

**UNIT - IV**

**Back tracking** - The General Method, The 8-Queens Problem, Sum of subsets, Graph Coloring, Hamiltonian cycles.

## UNIT-V

**Branch and Bound** – General method, Job sequencing with deadlines –LC Branch and Bound, FIFO Branch and Bound and LIFO Branch and Bound, 0/1 Knapsack problem - LC Branch and Bound solution, FIFO Branch and Bound solution, Travelling salesperson Problem – LC Branch and Bound solution

### **TEXTBOOK(S):**

1. Ellis Horowitz, Sartaj Sahni, SRajasekaran , “Fundamentals of Computer Algorithms”, University press, 2<sup>nd</sup> edition, 2012.

### **REFERENCE BOOKS:**

1. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C++”, Pearson, 3rd edition , 2007.
2. Aho, Hopcroft & Ullman, “The Design and Analysis of Computer Algorithms”, Addison Wesley publications, 2008.
3. Thomas H.Cormen et al, “Introduction to Algorithms”, PHI, 3rd edition, 2008
4. Anany Levitin, “Introduction to the Design and Analysis of Algorithms”, PEA,
5. P. H. Dave, H. B. Dave, “Design and Analysis of Algorithms”, Pearson Education”, 2008.

B.Tech. (IV Sem.)

**20CS10 -DATA WAREHOUSING AND DATA MINING**

L	T	P	Cr.
3	0	0	3

Pre-requisite : DBMS and Probability and Statistics

**Course Educational Objective:** The Objective of the course is to introduce the concepts of data warehouse and data mining, which gives a complete description about the principles, used, architectures, applications, design and implementation of data mining and data ware housing concepts.

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

**CO1:** Summarize the architecture of data warehouse.(Understand- L2)

**CO2:** Apply different preprocessing methods, Similarity, Dissimilarity measures for any given raw data.(Apply – L3)

**CO3:** Construct a decision tree and resolve the problem of model overfitting.(Analyze– L4)

**CO4:** Compare Apriori and FP-growth association rule mining algorithms for frequent itemset generation.(Apply - L3)

**CO5:** Apply suitable clustering algorithm for the given data set.(Apply - L3)

**UNIT-I**

**Data Warehouse and OLAP Technology:** An Overview: Data Warehouse, A Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to Data Mining. (Han &Kamber)

**UNIT-2**

**Data Mining:** Introduction, Data Mining, Motivating challenges, The origins of Data Mining, Data Mining Tasks, Types of Data, Data Quality.

Data Preprocessing: Aggregation, Sampling, Dimensionality Reduction, Feature Subset Selection, Feature creation, Discretization and Binarization, Variable Transformation, Measures of Similarity and Dissimilarity. (Tan &Vipin)

**UNIT-3**

**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.

**Model Overfitting:** Due to presence of noise, due to lack of representation samples, evaluating the performance of classifier: holdout method, random sub sampling, cross-validation, bootstrap. Bayes Theorem, Naïve Bayes Classifier (Tan &Vipin)

**UNIT-4**

**Association Analysis:** Basic Concepts and Algorithms: Problem Definition, Frequent Item Set Generation, Apriori Principle, Apriori Algorithm, Rule Generation, Compact Representation of Frequent Itemsets, FPGrowth Algorithm. (Tan &Vipin)

## UNIT-5

**Cluster Analysis:** Basic Concepts and Algorithms: Overview, 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, Strengths and Weaknesses; Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. (Tan & Vipin)

### **TEXTBOOKS:**

1. Introduction to Data Mining : Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Fifth Impression, Pearson, 2015.
2. Data Mining concepts and Techniques, 3rd Edition, Jiawei Han, Michel Kamber, Elsevier, 2011

### **REFERENCE BOOKS:**

1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning, 2010.
2. Data Mining : Introductory and Advanced topics : Dunham, First Edition, Pearson, 2020
3. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH, 2008.
4. Data Mining Techniques, Arun K Pujari, Universities Press, 2001

B.Tech. (IV Sem.)

20CS11 - OPERATING SYSTEMS

L	T	P	Cr.
3	0	0	3

Pre-requisite : Knowledge of Computer fundamentals & Data structures & Algorithms

**Course Educational Objective:** The objective of the course is to provide basic knowledge of computer operating system structure and functioning, understand how Operating Systems evolved with advent of computer architecture, and comprehend the different CPU scheduling algorithms, page replacement algorithms, disk scheduling and identify best one.

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

- CO1:** Demonstrate the underlying principles and techniques of operating system(**Understand-L2**)
- CO2:** Interpret scheduling and communication methods of processes handled by operating systems(**Understand-L2**).
- CO3:** Distinguish the process synchronization methods and deadlock handling approaches employed in operating systems(**Understand-L2**).
- CO4:** Classify memory management techniques and virtual memory mechanisms(**Understand-L2**).
- CO5:** Interpret the strategies of disk scheduling algorithms and file system architecture (**Understand-L2**).

### Unit-1: Introduction to Operating System

**Operating System Structures:** Operating-System Services, User Operating-System Interface, System Calls, Types of System Calls, System Programs, Operating-System Design and Implementation, Operating-System Structure, Virtual Machines, Operating-System Generation, System Boot.

### Unit-2: Process Management

**Processes:** Process concept, Inter-process Communication, Communication in Client-Server Systems. **Threads:** Overview, Multithreading Models

**Process Scheduling:** Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, PRIORITY, ROUNDROBIN)

### Unit-3: Synchronization and Deadlocks

**Synchronization:** The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors.

**Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention. Deadlock Avoidance, Deadlock Detection, Recovery from deadlock.

### Unit-4: Memory Management

**Memory Management Strategies:** Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.

**Virtual Memory Management:** Demand Paging, Page Replacement, Allocation of Frames, Thrashing.

**Unit-5: File System Management**

**Mass-Storage Structure:** Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management.

**Implementing File System:** File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery.

**TEXTBOOKS:**

1. Silberschatz & Galvin, “Operating System Concepts”, Wiley, 7th edition, 2007.

**REFERENCE BOOKS:**

1. William Stallings, “Operating Systems”, PHI, 5th Edition, 2004.
2. B.A. Forouzan & R.F. Giberg, —Unix and shell Programming, Thomson, First Edition, New Delhi, 2003.
3. <http://codex.cs.yale.edu/avi/os-book/OS9/slide-dir/index.html>
4. [https://swayam.gov.in/nd1\\_noc19\\_cs50/preview](https://swayam.gov.in/nd1_noc19_cs50/preview)

B.Tech. (IV Sem.)

20IT01 - SOFTWARE ENGINEERING

L	T	P	Cr.
3	0	0	3

Pre-requisite : Object Oriented Programming

**Course Educational Objective:** The objective of the course is to provide understanding of different s/w process models and how to choose one among them by gathering the requirements from a client and specifying them. Using those requirements in the design of the software architecture based on the choices with the help of modules and interfaces. To enable s/w development, by using different testing techniques like unit, integration and functional testing, the quality assurance can be achieved.

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

- CO 1 Understand the fundamentals of software engineering concepts and software process models. **(Understand-L2)**
- CO 2 Apply the requirement elicitation techniques for preparing SRS and design engineering. **(Apply-L3)**
- CO 3 Understanding the basic building blocks of UML, Class and object diagrams. **(Understand-L2)**
- CO 4 Apply the behavioral models for real world applications. **(Apply-L3)**
- CO 5 Demonstrate different software testing approaches for testing the real time applications. **(Understand-L2)**

#### UNIT – I:

**Software and software Engineering:** The evolving role of Software, Characteristics of Software, Importance of software Engineering, Changing nature of software, Legacy Software, Software Myths.

**Software Process and Process Models:** Layered technology, Process frame work, The process and Product, software process models, the water fall model, incremental model, the spiral and V Model, Component based s/w development, Unified process model,

#### UNIT – II:

**Requirements Analysis and Software design:** Requirements gathering and analysis, software requirements specifications (SRS).

**Design Engineering:** overview of design process, Design Concepts, Architectural Concepts

#### UNIT – III:

**Design Using UML:** Building Blocks of UML, Defining things, relationships and diagrams, Common Mechanism in UML, Class and Object Diagrams

#### UNIT – IV:

**Behavioral Modeling: Interactions,** Interaction diagrams, use cases, Use case Diagrams, Activity Diagrams, Events and signals, state machines, processes and Threads, time and space, state chart diagrams

**Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams.

**UNIT – V:**

**Testing Techniques:** Software testing fundamentals, Unit testing, Integration testing, Black box testing, white box testing, Debugging, System testing.

**TEXTBOOKS:**

1. Roger S. Pressman, “Software engineering- A practitioner ‘s Approach”, TMH International Edition, 6<sup>th</sup> edition, 2005.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modelling Language User Guide”, PEARSON ,4<sup>th</sup> Impression, 2012.

**REFERENCE BOOKS:**

1. Software Engineering - Concepts and practices: Ugrasen Suman, Cengage learning
2. Object-oriented analysis and design using UML”, Mahesh P. Matha, PHI
3. Fundamentals of Software Engineering, Rajib Mall, Third Edition ,PHI
4. [https://onlinecourses.nptel.ac.in/noc20\\_cs68](https://onlinecourses.nptel.ac.in/noc20_cs68) [1,2,3,4,5]



B.Tech. (IV Sem.)

**20HS01 – UNIVERSAL HUMAN VALUES 2:  
UNDERSTANDING HARMONY**

L	T	P	Cr.
3	0	0	3

**Pre-requisite** : Nil

**Course Educational Objective:** The objective of the course is to become more aware of themselves, and their surroundings (family, society, nature); they would become more responsible in life, and in handling problems with sustainable solutions, while keeping human relationships and human nature in mind.

**Course Outcomes:** At the end of this course,

**CO1:** Apply the value inputs in life and profession (**Apply – L3**)

**CO2:** Distinguish between values and skills, happiness and accumulation of physical facilities, the self, and the Body (**Understand – L2**)

**CO3:** Understand the role of a human being in ensuring harmony in society (**Understand – L2**).

**CO4:** Understand the role of a human being in ensuring harmony in the nature and existence. (**Understand – L2**)

**CO5:** Distinguish between ethical and unethical practices (**Apply – L3**)

**UNIT – I: Need, Basic Guidelines, Content and Process for Value Education**

‘Natural Acceptance’ and Experiential Validation- as the process for self-exploration; Continuous Happiness and Prosperity- A look at basic Human Aspirations; Right understanding, Relationship and Physical Facility, Understanding Happiness and Prosperity

**UNIT – II: Understanding Harmony in the Human Being - Harmony in Myself!**

Understanding human being as a co-existence of the sentient ‘I’ and the material ‘Body’; Understanding the needs of Self (‘I’) and ‘Body’ - happiness and physical facility; Understanding the Body as an instrument of ‘I’ (I am being the doer, seer and enjoyer);

Understanding the characteristics and activities of ‘I’ and harmony in ‘I’; Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail

**UNIT – III: Understanding Harmony in the Family and Society- Harmony in Human-Human Relationship**

Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfilment to ensure mutual happiness; Trust and Respect as the foundational values of relationship.

Understanding the harmony in the society: Resolution, Prosperity, fearlessness and co-existence as comprehensive Human Goals; Visualizing a universal harmonious order in society- Undivided

Society, Universal Order- from family to world family, Gratitude as a universal value in relationships.

**UNIT – IV: Understanding Harmony in the Nature and Existence - Whole existence as Coexistence**

Understanding the harmony in the Nature; Interconnectedness and mutual fulfilment among the four orders of nature- recyclability and self-regulation in nature; Understanding Existence as Co-existence of mutually interacting units in all-pervasive space; Holistic perception of harmony at all levels of existence.

**UNIT – V: Implications of the above Holistic Understanding of Harmony on Professional Ethics**

Natural acceptance of human values; Definitiveness of Ethical Human Conduct; Basis for Humanistic Education, Humanistic Constitution and Humanistic Universal Order; Competence in professional ethics, Strategy for transition from the present state to Universal Human Order

**TEXTBOOK(S):**

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

**REFERENCE BOOKS:**

1. Jeevan Vidya: EkParichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi

B.Tech. (IV Sem.)

20MC02 – ENVIRONMENTAL SCIENCE

L	T	P	Cr.
2	0	0	0

**Pre-requisite** : Nil

**Course Educational Objective:** The objective of this course is to understand Environmental issues like over population, human health etc., related to local, regional, and global levels. The necessity of resources, their exploitation and sustainable management. The interactions of human and ecosystems and their role in the food web in the natural world. The global biodiversity, threats to biodiversity and its conservation. Environmental problems like pollution, disasters and possible solutions. The importance of environmental decision making in organizations through audits.

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

- CO1:** Identify environmental problems arising due to engineering and technological activities that help to be the part of sustainable solutions. (**Remember– L1**)
- CO2:** Evaluate local, regional, and global environmental issues related to resources and their sustainable management (**Understand - L2**)
- CO3:** Realize the importance of ecosystem and biodiversity for maintaining ecological balance. (**Understand - L2**)
- CO4:** Acknowledge and prevent the problems related to pollution of air, water, and soil. (**Apply– L3**)
- CO5:** Identify the significance of implementing environmental laws and abatement devices for environmental management. (**Understand - L2**)

### UNIT – I :Nature and scope of Environmental Problems

- Introduction to Environment Science.
- Population explosion, variations among nations
- Resettlement and Rehabilitation - Issues and possible solutions
- Environmental hazards – causes and solutions. Biological hazards – AIDS, Malaria, Chemical hazards- BPA, PCB, Phthalates, Mercury, Nuclear hazards- Risk and evaluation of hazards.
- Role of Information Technology in environmental management and human health

### UNIT – II :Natural Resources and Conservation

Introduction and classification of Natural Resources

- Forest resources: Use and over-exploitation, deforestation, Timber extraction, mining, dams and their effects on forests and tribal people
- Water resources: Use and over-utilization of surface and ground water, conflicts over water, interlinking of rivers, dams-benefits and problems, Rainwater harvesting
- Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources
- Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, soil salinity
- Energy resources: Growing energy needs renewable, non-renewable and alternate energy resources

**UNIT – III :Ecology and Biodiversity**

- Structure and functions of an Ecosystem, Food chains and Food webs, Ecological succession, Ecological pyramids, Biogeochemical cycles
- Biodiversity, Values of biodiversity, Bio geographical classification of India. Endangered and endemic species of India, Threats to biodiversity; Man and wild life conflicts, Conservation of biodiversity: In-situ and Ex-situ conservation methods

**UNIT – IV : Environmental Pollution**

Introduction to Environmental Pollution Causes, effects and control measures of:

Air pollution, Water pollution, Noise pollution, Solid Waste Management – Sources, Classification, effects and control measures of Municipal solid waste, Biomedical waste & Hazardous and e-waste, Disaster Management.

**UNIT – V : Environmental Management**

- Sustainable development and unsustainability
- Climate disruption, Greenhouse effect, Ozone layer depletion and Acid rain.
- Stockholm and Rio Summit
- Environmental Impact Assessment (EIA)
- Green building
- Environmental Law- Air, Water, Wildlife, Forest, and Environmental protection act

**TEXTBOOKS:**

1. Anubha Kaushik, C.P.Kaushik, “*Perspectives in Environmental Studies*”, 5<sup>nd</sup> edition, New age international publishers, Delhi, 2016.
2. G. Tyler Miller, Scott Spoolman, “*Introduction to Environmental Studies*”, 13<sup>th</sup> Edition, Cengage Learning, New Delhi, 2009.

**REFERENCE BOOKS:**

1. M. Anji Reddy, “Textbook of Environmental Sciences and Technology”, 2nd Edition, BS Publications, Delhi 2011.
2. Deeshita Dave, P. Udaya Bhaskar, “Environmental Studies”, 2nd Edition, Cengage Learning, New Delhi, 2012.
3. S.Deswal, A. Deswal, “A Basic course in Environmental Studies”, 2nd Edition, Educational & Technical Publishers, Delhi, 2014.
4. R. Rajagopalan, “Environmental Studies (From Crisis to Cure)”, 3rd Edition, Oxford University Press, New Delhi, 2012.
5. De, A.K, “Environmental Chemistry”, 5th Edition, New Age International (P) Limited, New Delhi, 2003.
6. Dr.K.V.S.G. Murali Krishna, “Environmental Studies”, 1st Edition, VGS Techno Series, Vijayawada, 2010.
7. Mahua Basu, S.Xavier, “Fundamentals of Environmental Studies”, 1st edition, Cambridge University Press, Delhi, 2016.

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite :** Python Programming.

**Course Educational Objective:** The objective of this lab is to Practical exposure on implementation of well-known data mining algorithms and Learning performance evaluation of data mining algorithms in a supervised and an unsupervised setting.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Apply preprocessing techniques on real world datasets.(**Apply-L3**)
- CO2:** Apply apriori algorithm to generate frequent itemsets.(**Apply L3**)
- CO3:** Apply Classification and clustering algorithms on different datasets.(**Apply L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Use python library scikit-learn wherever necessary

1. Demonstrate the following data preprocessing tasks using python libraries.
  - a) Loading the dataset
  - b) Identifying the dependent and independent variables.
  - c) Dealing with missing data
2. Demonstrate the following data preprocessing tasks using python libraries.
  - a) Dealing with categorical data.
  - b) Scaling the features.
  - c) Splitting dataset into Training and Testing Sets
3. Demonstrate the following Similarity and Dissimilarity Measures using python
  - a) Pearson's Correlation
  - b) Cosine Similarity
  - c) Jaccard Similarity
  - d) Euclidean Distance
  - e) Manhattan Distance
4. Build a model using linear regression algorithm on any dataset.
5. Build a classification model using Decision Tree algorithm on iris dataset
6. Apply Naïve Bayes Classification algorithm on any dataset
7. Generate frequent itemsets using Apriori Algorithm in python and also generate association rules for any market basket data.
8. Apply K- Means clustering algorithm on any dataset.
9. Apply Hierarchical Clustering algorithm on any dataset.
10. Apply DBSCAN clustering algorithm on any dataset.

**Web Resources:**

1. <https://analyticsindiamag.com/data-pre-processing-in-python/>
2. <https://towardsdatascience.com/decision-tree-in-python-b433ae57fb93>

3. <https://towardsdatascience.com/calculate-similarity-the-most-relevant-metrics-in-a-nutshell-9a43564f533e>
4. <https://www.springboard.com/blog/data-mining-python-tutorial/>
5. <https://medium.com/analytics-vidhya/association-analysis-in-python-2b955d0180c>
6. <https://www.datacamp.com/community/tutorials/naive-bayes-scikit-learn>
7. <https://www.analyticsvidhya.com/blog/2019/05/beginners-guide-hierarchical-clustering/>
8. <https://towardsdatascience.com/dbscan-algorithm-complete-guide-and-application-with-python-scikit-learn-d690cdae4c5d>

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Shell Scripting and Object-Oriented Programming

**Course Educational Objective:** The objective of this lab is to provide the various UNIX/Linux operating system commands, importance of System calls, Scheduling algorithms and Memory Management techniques.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Experiment with Unix commands and shell programming (**Understand- L2**)
- CO2:** Implement CPU scheduling algorithms and memory management techniques(**Apply- L3**).
- CO3:** Simulate process synchronization and file system management using system calls(**Apply –L3**).0
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

#### Cycle-1:

Execute various UNIX system calls

1. Process Management
2. File Management
3. Input/Output System Calls

#### Cycle-2:

Simulate the following CPU scheduling algorithms.

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

#### Cycle-3:

Simulate the file allocation strategies:

a) Sequential b) Indexed c) Linked

#### Cycle-4:

Simulate MVT and MFT

Simulate contiguous memory allocation techniques

a) Worst-fit b) Best fit c) First fit

#### Cycle-5:

Simulate all File Organization techniques

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

#### Cycle-6:

Simulate Bankers Algorithm for Deadlock Avoidance

Simulate Bankers algorithm for Deadlock Prevention

**Cycle-7:**

Simulate disk scheduling algorithms.

a) FCFS b) SCAN c) C-SCAN

**Cycle-8:**

Programs on process creation and synchronization, inter process communication including shared memory, pipes, and messages. (Dinning - Philosopher problem).

**REFERENCES:**

1. <https://github.com/brenns10/lsh/tree/407938170e8b40d231781576e05282a41634848c>
2. <https://github.com/MaaSTaaR/SSFS>



L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Object Oriented Programming

**Course Educational Objective:** The main objective of this course is that a student will be familiar with principles behind the Object-Oriented Design and able to apply those principles in a project setting. Students will analyze applications and know how to take a pragmatic approach to software design and development.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Analyze Software Requirements for the given Real-World Application using Use Cases. **(Analyze-L4)**
- CO2:** Develop the UML Diagrams to view Software System in static aspects. **(Analyze-L4)**
- CO3:** Develop the UML Diagrams to view Software System in dynamic aspects. **(Analyze-L4)**
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

Consider the following Case Studies:

- 1) Automated Teller Machine (ATM)
- 2) Library Management System
- 3) Railway Ticket Reservation System
- 4) Point-of-Sale Terminal
- 5) Customer Support Service Operations
- 6) Cab Booking Service

**Cycle-1: Analyze the Requirements for the following Case Studies.**

- 1) Automated Teller Machine (ATM)
- 2) Library Management System
- 3) Railway Ticket Reservation System

**Cycle-2: Analyze the Requirements for the following Case Studies.**

- 1) Point-of-Sale Terminal
- 2) Customer Support Service Operations
- 3) Cab Booking Service

**Cycle-3: Basics of UML**

- 1) Introduction to UML.
- 2) Familiarization with any one of the Software such as Rational Rose or Umbrello or Gliffy Diagram etc.

**Cycle-4: For each case study given earlier, Construct Use Case Diagram for following:**

- 1) Identify and Analyze the Actors.
- 2) Identify the Actions.
- 3) Analyze the Relationships between Actors and Actions.
- 4) Sketch the Use Case Diagram.

**Cycle-5 and Cycle-6:For each case study given earlier, Construct Class Diagram in the following manner:**

- 1) Identify and Analyze the Classes related to your problem.
- 2) Analyze the Attributes and Operations
- 3) Analyze the Relationships between Classes
- 4) Sketch the Class Diagram

**Cycle-7:For each case study given earlier, Construct Interaction Diagrams in the following manner:**

- 1) Identify the Objects participating in Communication.
- 2) Identify the Messages between the objects.
- 3) Give numbering to messages.
- 4) Use Flat Sequencing or Procedural Sequencing for numbering.

**Cycle-8:For each case study given earlier, Construct Activity Diagram in the following manner:**

- 1) Identify activities in your case study.
- 2) Identify relationships among activities.
- 3) Use Fork or Join, if necessary.
- 4) Sketch the diagram.

**Cycle-9:For each case study given earlier, Construct State Chart Diagram in the following manner:**

- 1) Identify the different states in your case study.
- 2) List out the different sub-states present in the state.
- 3) Identify relationships among the state to state.
- 4) Sketch the diagram.

**Cycle-10:For each case study given earlier, Construct Component Diagram in the following manner:**

- 1) Identify the different components in your case study.
- 2) Create a visual for each of the component.
- 3) Describe the organization and relationships between components using interfaces, ports etc.
- 4) Sketch the diagram.

**Cycle-11:For each case study given earlier, Construct Deployment Diagram in the following manner:**

- 1) Identify the nodes.
- 2) Identify the relationships among the nodes.
- 3) Sketch the Diagram.

**REFERENCE BOOK(S):**

1. Roger S. Pressman, “Software engineering- A practitioner’s Approach”, TMH International Edition, 6th edition, 2005.
2. Grady Booch, James Rumbaugh, Ivar Jacobson, “The Unified Modeling Language UserGuide”, Pearson Education, 2<sup>nd</sup> Edition, ISBN: 0-201-57168-4, 1998.
3. Meilir Page-Jones, “Fundamentals of Object-Oriented Design in UML”, Pearson Education, 1<sup>st</sup> Edition, ISBN: 9788177586770, 8177586777, and 2007.
4. Robert B. Jackson, Stephen D. Burd, John W. Satzinger, “Object-Oriented Analysis and Design with the Unified Process”, Cengage Learning, 1<sup>st</sup> Edition, ISBN: 9788131502693, 8131502694, 2007.
5. Craig Larman, “Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development”, PHI, 3<sup>rd</sup> Edition, ISBN: 978-0131489066, 2004.

**B.Tech. (IV Sem.)**

**20CSS2–WEB APPLICATION DEVELOPMENT  
USING FULL STACK – Module-II  
(Backend Development)**

L	T	P	Cr.
1	0	2	2

**Pre-requisite** : Object Oriented Programming and Data Base Management Systems.

**Course Educational Objective:** The objective of this course is to learn the importance of client-server architecture in the web application development and able to develop dynamic data driven web applications by using advanced java technologies (Servlets, JSP, Struts2 and Hibernate framework).

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Apply JDBC concepts to establish the communication between Java Applications and database. (**Apply-L3**)
- CO2:** Develop Static and Dynamic Web Applications by using Servlets and Java Server Pages (JSP). (**Apply L3**)
- CO3:** Develop Dynamic Data Driven Web Applications by using Struts2 and Hibernate frameworks. (**Apply L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Module 1:**

- a) Develop a java program to connect oracle database by using JDBC Connection string.
- b) Develop a java program to prepare the Statement to execute the query to insert the number of records into oracle database.
- c) Develop a java program to fetch the records from the oracle database table by using ResultSet Interface.

**Module 2:**

- a) Develop a java program to prevent the SQL Injection attacks by using PreparedStatement.
- b) Develop a java program to execute stored procedures and sub functions by using Callable Statement.
- c) Develop a java program to Scroll the ResultSet data in both forward and backward directions by using ScrollableResultSet.
- d) Develop a java program to print the database table meta data by using ResultSetMetaData.

**Module 3:**

- a) Develop a Servlet program to demonstrate the role of Servlet Life Cycle methods in the web application.
- b) Develop a Servlet program to access Init parameter values from web.xml to Servlet program by using ServletConfig interface.
- c) Develop a Servlet program to navigate from one Servlet page to another Servlet page by using RequestDispatcher interface.

**Module 4:**

- a) Develop a Servlet program to AUTHENTICATE User details from oracle database by using JDBC connectivity. (Dynamic Login Checking for the registered users in the oracle database)
- b) Develop a Servlet program to implement Session Management concept by using HttpSession.

**Module 5:**

- a) Develop a JSP program to demonstrate the importance of Scripting elements in JSP.
- b) Develop a JSP program to illustrate the importance of implicit objects in JSP.
- c) Develop a JSP program to access JavaBean class by using jsp:useBean, jsp:setProperty and jsp:getProperty.

**Module 6:**

- a) Develop a JSP program to insert and retrieve the records from the oracle database.
- b) Develop a JSP program to demonstrate the session management in between login and logout period of the end user.
- c) Develop a JSP program to illustrate the process of handling errors and exceptions in JSP pages.

**Module 7:**

- a) Develop a Struts2 web application to perform Insert, Update and Delete operations in the oracle database table through JDBC.
- b) Develop a Struts2 web application to fetch all the records of the table created in the above from oracle database.

**Module 8:**

- a) Develop a Program how to code a basic Java Hibernate program by using eclipse IDE.
- b) Develop a web application to insert the record of the user in the oracle database by using Hibernate.

**REFERENCE BOOK(S):**

- 1. Herbert Schildt, “Java: The complete reference”, TMH Publications, 7th edition, 2006.
- 2. Kathy Sierra & Bert Bates, “Head First Servlets and JSP: Passing the Sun Certified Web Component Developer Exam”, O'Reilly Publications Second Edition.
- 3. Budi Kurniawan, “Struts 2 Design and Programming: A Tutorial”, BrainySoftware, 2<sup>nd</sup> Edition, 2008.
- 4. Christian Bauer, Gavin King, Gary Gregory “Java Persistence with Hibernate: Revised Edition of Hibernate in Action Paperback”, Manning Publication, 2nd Edition, 2006.
- 5. Santosh Kumar K, “JDBC 4.2, Servlet 3.1, and JSP 2.3 Includes JSF 2.2 and Design Patterns, Black Book”, Dreamtech publication, 2<sup>nd</sup> Edition.
- 6. Mahmoud Parsian, “JDBC Recipes: A Problem-Solution Approach”, Apress.
- 7. Madhusudhan Konda, “Just Hibernate, A Lightweight Introduction to the Hibernate Framework”, O'Reilly Media.
- 8. Chuck Cavaness, “Programming Jakarta Struts”, O'Reilly Media, 2nd Edition.

B.Tech. (V Sem.)

20CS12-COMPUTER NETWORKS

L	T	P	Cr.
3	0	0	3

**Pre-requisite** : Data Structures and Operating Systems

**Course Educational Objective:** The Objective of the course is to provide a foundation to understand computer networks using layered architectures. It also helps students to understand the various network models, addressing concept, routing protocols and design aspects of computer networks.

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

- CO1:** Demonstrate the modern network architectures from a design perspective (**Understand- L2**)
- CO2:** Apply various Data Link layer design issues and error detection & correction techniques to solve collisions problems. (**Apply- L3**)
- CO3:** Demonstrate the network Layer functionalities (**Understand- L2**)
- CO4:** Outline the functions of transport layer protocols (**Understand- L2**)
- CO5:** Examine different application layer protocols. (**Understand- L2**)

**UNIT-I**

**Data Communication Components:** Representation of data and its flow Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN.

**UNIT-II**

**Data Link Layer and Medium Access Sub Layer:** Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA, CSMA/CD,CDMA/CA

**UNIT-III**

**Network Layer:** Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.

**UNIT-IV**

**Transport Layer:** Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

**UNIT-V**

Application Layer: Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls,

**TEXT BOOKS:**

1. Behrouz A. Forouzan, , “Data Communication and Networking”, McGraw-Hill, 4th Edition, 2011.
2. Andrew S. Tanenbaum, “Computer Networks”, Pearson New International Edition, 8th Edition, 2013.

**REFERENCE BOOKS:**

1. William Stallings, “Data and Computer Communication”, Pearson Prentice Hall India, 8<sup>th</sup> Edition.
2. Douglas Comer, Internetworking with TCP/IP, Prentice Hall of India, Volume 1, 6<sup>th</sup> Edition, 2009.
3. Richard Stevens, “TCP/IP Illustrated” , Addison-Wesley, Volume 1, 2001.
4. <http://www.cse.iitk.ac.in/users/dheeraj/cs425/>
5. [http://www.tcpipguide.com/free/t\\_OSIReferenceModelLayers.htm](http://www.tcpipguide.com/free/t_OSIReferenceModelLayers.htm)



B.Tech. (V Sem.)

20AD04-MACHINE LEARNING

L	T	P	Cr.
3	0	0	3

**Pre-requisite :** Probability and Statistics, Data Warehousing and Data Mining

**Course Educational Objective:** The objective of the course provides the basic concepts and techniques of Machine Learning and helps to use recent machine learning software for solving practical problems. It enables students to gain experience by doing independent study and research.

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

<b>CO1:</b>	Identify the characteristics of machine learning. ( <b>Understand- L2</b> )
<b>CO2:</b>	Understand the Model building and evaluation approaches ( <b>Understand- L2</b> )
<b>CO3:</b>	Apply regression algorithms for real-world Problems. ( <b>Apply- L3</b> )
<b>CO4:</b>	Handle classification problems via supervised learning algorithms. ( <b>Apply- L3</b> )
<b>CO5:</b>	Learn advanced learning techniques to deal with complex data ( <b>Apply- L3</b> )

### **UNIT-I: Introduction to Machine Learning and Preparing to Model**

**Introduction to Machine Learning** - Introduction, Types of Machine Learning, Applications of Machine Learning, Issues in Machine Learning.

**Preparing to Model**- Introduction, Machine Learning Activities, Basic Types of Data in Machine Learning, Exploring Structure of Data, Data Quality and Remediation, Data Pre-Processing

### **UNIT-II: Modelling & Evaluation, Basics of Feature Engineering**

**Modelling & Evaluation**- Introduction, selecting a Model, training a Model (for Supervised Learning), Model Representation and Interpretability, Evaluating Performance of a Model.

**Basics of Feature Engineering**- Introduction, Feature Transformation – Feature Construction, Feature Extraction, Principal Component Analysis (PCA), Singular Value Decomposition (SVD), Linear Discriminant Analysis (LDA), Feature Subset Selection

### **UNIT-III: Regression**

Introduction to regression analysis, Simple linear regression, Multiple linear regression, Assumptions in Regression Analysis, Main Problems in Regression Analysis, Improving Accuracy of the linear regression model, Polynomial Regression Model, Logistic Regression, Regularization, Regularized Linear Regression, Regularized Logistic Regression.

### **UNIT-IV: Supervised Learning: Classification**

**Classification**- Introduction, Example of Supervised Learning, Classification Model, Classification Learning Steps.

**Common Classification Algorithms** - k-Nearest Neighbour (kNN), Support vector Machines (SVM), Random Forest model.

### **UNIT-V: Other Types of Learning**

**Ensemble Learning**- Bagging, Boosting, Stacking and its impact on bias and variance, AdaBoost, Gradient Boosting Machines, XGBoost.

**Reinforcement Learning** - Introduction, Q Learning.

### **TEXT BOOKS:**

1. Subramanian Chandramouli, Saikat Dutt, Amit Kumar Das, “Machine Learning”, Pearson Education India ,1<sup>st</sup> edition,2015.
2. Tom M. Mitchell, “Machine Learning”, MGH, 1997.

### **REFERENCE BOOKS:**

1. Shai Shalev-Shwartz, Shai Ben David, “Understanding Machine Learning: From Theory to Algorithms”, Cambridge.
2. Peter Harington, “Machine Learning in Action” , Cengage, 1<sup>st</sup> edition, 2012.
3. Peter Flach, “Machine Learning: The art and science of algorithms that make sense of data”, Cambridge university press,2012.
4. Jason Brownlee, “Machine Learning Mastery with Python Understand Your Data, Create Accurate Models and Work Projects End-To-End”, Edition: v1.4, 2011.

B.Tech. (V Sem.)

20CS13-THEORY OF COMPUTATION

L	T	P	Cr.
3	0	0	3

Pre-requisite : Discrete Mathematical Structures

**Course Educational Objective:** The objective of the course is to provide a formal connection between algorithmic problem solving and the theory of Automata and languages, and develop them into a mathematical view towards algorithmic design and in general computation itself.

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

- CO1:** Construct finite automata for regular languages and prove it's equivalence. **(Apply-L3)**
- CO2:** Construct regular expression for regular languages and prove the equivalence of regular expression and Finite Automata. **(Apply-L3)**
- CO3:** Design Pushdown automata for the context-free languages. **(Understand-L2)**
- CO4:** Design Turing machine to model computational problems. **( Apply-L3)**
- CO5:** Distinguish decidable and undecidable problems with the help of Turing machine. **(Understand – L2)**

**UNIT-1 : Finite Automata**

**Introduction:** Alphabet, Strings and Languages and grammars, Classification of Automata, definitions, and its applications.

**Finite Automata:** Deterministic finite automata (DFA), nondeterministic finite automata (NFA). Equivalence of NFA and DFA. Equivalence of NFA with epsilon and NFA without epsilon, and minimization of finite automata. Finite Automata with output: Mealy and Moore Machine and its equivalence.

**UNIT-II : Regular Expression and Regular languages**

**Regular Expression :** Equivalence of regular expression and finite automata.

**Regular Grammar :** Definition of grammar, derivation and parse tree, equivalence of regular grammar and finite automata, closure properties of regular languages, pumping lemma for regular languages.

**UNIT- III : CFL and Pushdown Automata**

**Context-Free Languages :** Context-Free Grammars (CFG), ambiguity in CFG, Chomsky and Greibach normal forms.

**Pushdown automata (PDA):** Definition of PDA, Deterministic and Non deterministic PDA, equivalence of PDA and CFG, pumping lemma for context-free languages, closure properties of CFLs.

**UNIT – IV : Turing Machine**

**Turing Machine:** The basic model for Turing Machine (TM), Turing-recognizable (recursively enumerable) and Turing-decidable (recursive) languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators.

**UNIT – V : Undecidability**

**Undecidability:** Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice's theorem, Post's correspondence problem, undecidable problems about languages.

**TEXT BOOKS:**

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, "Introduction to Automata Theory, Languages, and Computation", Pearson Education Asia, 1997.

**REFERENCE BOOKS:**

1. Harry R. Lewis and Christos H. Papadimitriou, "Elements of the Theory of Computation", Pearson Education Asia, 2000.
2. Dexter C. Kozen, "Automata and Computability", Springer, 2011.
3. Michael Sipser, "Introduction to the Theory of Computation", PWS Publishing, 2005.
4. John Martin, Introduction to Languages and The Theory of Computation, Tata McGraw Hill, 2<sup>nd</sup> Edition, 2003.

B.Tech. (V Sem.)

20CS14-COMPUTER GRAPHICS

L	T	P	Cr.
3	0	0	3

**Pre-requisite :** Knowledge of coordinate geometry, and Programming for Problem solving using C .

**Course Educational Objective** The objective of the course is to have an appreciation of the history and evolution of computer graphics, both hardware and software. Students will understand 2D graphics and algorithms which include line drawing, polygon filling, clipping, and transformations. Students will understand the concepts & techniques used in 3D computer graphics, including viewing transformations, hierarchical modeling, colour, lighting and texture mapping.

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

- CO 1 Summarize the working principle of display devices, interactive input devices and graphic applications **(L1-Remember)**
- CO 2 Analyze line, circle, ellipse and character generation algorithms. **(L2-Understand)**
- CO 3 Demonstrate 2-Dimensional geometrical transformations. **(L3-Apply)**
- CO 4 Apply the clipping algorithms on points, lines , and polygons. **(L3- Apply)**
- CO 5 Apply transformation techniques on 3-Dimensional objects. **(L3-Apply)**

**UNIT – I: Introduction to Computer Graphics and their Applications**

**A Survey of Computer Graphics:** Presentation Graphics, Computer Aided Design, Computer Art, Entertainment, Education and Training, Visualization, Image Processing, Graphical User Interfaces.

**Overview of Graphics systems:** Video Display Devices, Raster Scan systems, random scan systems, Graphics monitors and workstations, Input devices.

**UNIT – II: Output primitives**

**Output primitives:** Points and Lines, Line Drawing Algorithms, Loading the Frame buffer, Line function, Circle, Generating Algorithms, Ellipse Generating Algorithms, Other Curves, Parallel Curve Algorithms, Curve Functions, Pixel Addressing, Filled Area Primitives, Filled Area Functions.

**UNIT – III: Two Dimensional Geometric Transformations**

**Two Dimensional Geometric Transformations:** Basic Transformations, Matrix Representations, Homogeneous Coordinates, Composite Transformations, Other Transformations, Transformations between Coordinate Systems, Affine Transformations, Transformation Functions, Raster methods for Transformation.

**UNIT – IV: Two-Dimensional Viewing**

**Two-Dimensional Viewing:** The viewing Pipeline, Viewing Coordinate Reference Frame, Window-to-Viewport Coordinate Transformation, Two Dimensional Viewing Functions, Clipping Operations: Point Clipping, Line Clipping, Polygon Clipping.

**UNIT – V: Three Dimensional Concepts, Object Representations, Geometric and modelling Transformations**

Three Dimensional Concepts and Object representations: 3D display methods, 3DGraphics-Polygon Surfaces, Curved Lines and Surfaces, Quadratic Surfaces Three Dimensional Geometric and Modelling Transformations: Translation, Rotation, Scaling, Other Transformations, Composite Transformations, 3D Transformation Functions, Modelling and Coordinate Transformations.

**TEXT BOOK**

1. Donald Hearn & M. Pauline Baker, “Computer Graphics C Version”, Pearson Education, New Delhi, 2009 (Chapters 1 to 12 except 10-9 to 10-22 of the Text book)

**REFERENCE BOOKS:**

1. David F. Rogers “Procedural Elements for Computer Graphics” TMH Publications.
2. J. D. Foley, S. K Feiner, A Van Dam F. H John; “Computer Graphics: Principles & Practice in C” Pearson.
3. Francis S Hill Jr “Computer Graphics using open GL”; Pearson Education, 2004.
4. NPTEL : Computer Graphics , Prof. Sukhendu Das, IIT Madras.
5. NPTEL : Introduction to computer graphics by Prof. Prem K Kalra Department of Computer Science and Engineering, IIT Delhi url : <https://nptel.ac.in/courses/106/102/106102065/>

B.Tech. ( V Sem.)

## 20CS15-PRINCIPLES OF PROGRAMMING LANGUAGES

L	T	P	Cr.
3	0	0	3

Pre-requisite : Object Oriented Programming.

**Course Educational Objective:** Introduce the Programming Language Categories and Implementation Methods for learning Various Concepts of Imperative Programming Languages that include Exception and Event Handling.

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

- CO1:** Understand the Preliminary Concepts and Implementation Methods for Programming Languages. (**Understand- L2**)
- CO2:** Summarize Semantic issues of Variables and design issues of the Data Types in different Programming Languages. (**Understand- L2**)
- CO3:** Demonstrate Statement-Level Control Structures and concepts of Subprograms. (**Understand- L2**)
- CO4:** Illustrate the process of Exception and Event handling in different Programming Languages. (**Understand- L2**)
- CO5:** Describe the features of Functional Programming Languages and Scripting Languages. (**Remember- L1**)

### UNIT – I

**Preliminary Concepts:** Reasons for studying concepts of Programming Languages, Language Categories - Imperative, Functional, Logic and Object Oriented. Programming Language Implementation Methods.

**Describing Syntax:** General Problem of describing Syntax, Formal methods of describing syntax - BNF, EBNF for common programming languages features, Parse Trees, Ambiguous Grammars, Attribute Grammars.

### UNIT – II

**Names, Bindings, Type Checking and Scopes:** Names, Variables, The Concept of Binding, Type Checking, Strong Typing, Type Compatibility, Scope and Lifetime.

**Data types:** Introduction, Primitive Data Types, Character String Type, User Defined Ordinal Types, Array Types, Associative Arrays, Record Types, Union Types, Pointer and Reference Types.

### UNIT – III

**Statement-Level Control Structures:** Selection Statements, Iterative Statements, Unconditional Branching, Guarded Commands.

**Subprograms:** Fundamentals of Subprograms, Design issues of Subprograms, Local Referencing Environments, Parameter Passing Methods, Parameters that are Subprogram

Names, Overloaded Subprograms, Design Issues for Functions, User defined Overloaded Operators, Co-routines.

#### **UNIT – IV**

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

#### **UNIT-V**

**Functional Programming Languages:** Introduction, Fundamentals of FPL, LISP, Applications of Functional Programming Languages and Comparison of Functional and Imperative Languages. **Scripting Language:** Pragmatics, Key Concepts, Case Study: Python – Values and Types, Variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library.

#### **TEXT BOOK:**

1. Robert. Sebesta, “Concepts of Programming Languages”, Pearson Education, 10<sup>th</sup> edition, 2013.

#### **REFERENCE BOOKS:**

1. Ghezzi, “Programming Languages”, John Wiley, 3<sup>rd</sup> edition.
2. Pratt and Zelkowitz, “Programming Languages Design and Implementation”, PHI/Pearson Education, 4<sup>th</sup> edition.
3. A B Tucker, R E Noonan, “Programming Languages”, TMH, 2<sup>nd</sup> edition.
4. K C Loudon, “Programming Languages”, Thomson, 2003, 2<sup>nd</sup> edition.



B.Tech. (V Sem.)

**20CS16-PRINCIPLES OF ARTIFICIAL INTELLIGENCE**

L	T	P	Cr.
3	0	0	3

**Pre-requisite** : Basic Engineering Mathematics Knowledge

**Course Educational Objective:** The objective of the course is to present an overview of artificial intelligence (AI) principles and approaches. Develop a basic understanding of the building blocks of AI as presented in terms of intelligent agents: Search, Knowledge representation, inference, logic, reasoning, and learning. Students will implement a small AI system in a team environment. The knowledge of artificial intelligence plays a considerable role in some applications students develop for courses in the program.

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

- CO1:** Understand the fundamentals of Artificial Intelligence types of AI agents and their structures to solve engineering problems. **(Understand - L2)**
- CO2:** Identify different searching algorithms to find and optimize the solution for the given problem. **(Understand - L2)**
- CO3:** Apply different gaming algorithms and identify the importance of knowledge representation in Artificial Intelligence. **(Apply- L3)**
- CO4:** Make use of predicate logic and rule-based system to represent the knowledge in AI domain. **(Understand - L2)**
- CO5:** Interpret the forms of learning in the AI domain as well as present efficient techniques to remove uncertainty in knowledge domain. **(Understand - L2)**

## UNIT - I

**Introduction:** AI introduction, applications of AI, history of AI, types of AI.

**Intelligent Agents:** Agents and rationality, structure of agents, agent environment and nature of the environment, types of agents - simple reflex agents, model-based agents, goal-based agents, utility-based agents, learning agents.

**Problem Solving:** Problems, search spaces, defining the problem as state space search, production system, problem characteristics, issues in the design of search programs.

## UNIT - II

**Search Algorithms:** Problem solving agents, search algorithms terminologies, properties of search algorithms, types of search algorithms.

**Uniformed/Blind Search Algorithms:** Breadth-first Search, Depth-first Search, Depth-limited Search, Iterative deepening depth-first search, Uniform cost search, Bidirectional Search.

**Informed/ Heuristic Search algorithms:** Greedy best-first search algorithm, A\* search algorithm, Hill climbing algorithm, Constraint satisfaction problem, Means-Ends Analysis.

## UNIT - III

**Adversarial Search/ Game Playing:** Introduction, Minimax algorithm, Alpha-Beta pruning.

**Knowledge Representation:** Representations and mappings, approaches of knowledge representation, issues in knowledge representation.

## UNIT - IV

**Knowledge Representation Using Predicate Logic:** Representing simple facts in logic, representing instance and Isa relationships, computable functions and predicates, resolution, natural deduction.

**Representing Knowledge using Rules:** Procedural verses declarative knowledge, logic programming, forward verses backward reasoning, matching, control knowledge.

## UNIT - V

**Uncertain Knowledge and Reasoning:** Probability and Bayes' theorem, certainty factors and rule-based systems, Bayesian networks, Dempster-Shafer theory, Fuzzy logic.

**Learning:** Overview of different forms of learning, Learning Decision Trees, Neural Networks.

## TEXT BOOKS:

1. Stuart Russell, Peter Norvig, "Artificial Intelligence: A Modern Approach", 3rd edition, Prentice Hall, 2009. Can also use 2nd Ed., Pearson Education International, 2003.
2. Rich & Knight, Artificial Intelligence, Tata McGraw Hill, second edition, 2007

**REFERENCE BOOKS:**

1. Nils Nilsson, “Artificial Intelligence: A New Synthesis”, Morgan Kaufmann, 1998.
2. David Poole, Alan Mackworth, “Artificial Intelligence: Foundations for Computational Agents”, Cambridge Univ. Press, 2010.
3. Ronald Brachman, “Knowledge Representation and Reasoning”, Morgan Kaufmann, 2004.
4. Frank van Harmelen, Vladimir Lifschitz, Bruce Porter (Eds), “Handbook of Knowledge representation”, Elsevier, 2008.
5. Ivan Bratko, “Prolog Programming for Artificial Intelligence”, 4th Ed., Addison-Wesley, 2011.

B.Tech. (V Sem.)

20CS60-COMPUTER NETWORKS LAB

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Data Structures, and Operating Systems

**Course Educational Objective:** The Objective of the course is to prepare students to write programs to illustrate communication in networks, configure different networks (LAN, WAN) and prepare students to differentiate various protocols and their performance.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

**CO1:** Implement Network layer functionalities using NS3 simulator. (**Apply-L3**)

**CO2:** Demonstrate Transport Layer functionalities. (**Understand- L2**)

**CO3:** Analyze Application layer protocols using Wireshark. (**Analyze – L4**)

**CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

1. To gain familiarity with the basic network commands & utilities available in the Linux OS.
2. To learn about network layer tools and analyze captures for congestion.
3. To learn about queue management techniques, and global routing in ns3.
4. To learn about broadcasting, multicasting, and bridging in a Local Area Network using ns3.
5. To learn about Wifi and Mobile Adhoc topologies with ns3.
6. To introduce Socket Programming in TCP and UDP.
7. Observations of Transmission Control Protocol (TCP) Connection states, Flags and Flow control.
8. To learn Transmission Control Protocol (TCP) Flow Control, Error Control, and Congestion.
9. To introduce Wireshark & tcpdump, and observation of packets in a LAN network.
10. To analyze HTTP packets using Wireshark tool, and understand the records returned by a DNS server.

B.Tech. (V Sem.)

20AD54-MACHINE LEARNING LAB

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Probability and Statistics, Programming Knowledge

**Course Educational Objective:** The objective of this lab is to Make use of Data sets in implementing the machine learning algorithms in any suitable language of choice.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

**CO1:** Apply the appropriate pre-processing techniques on data set. (**Apply – L3**)

**CO2:** Implement supervised Machine Learning algorithms. (**Apply – L3**)

**CO3:** Implement advanced Machine Learning algorithms (**Apply – L3**)

**CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

### Experiments:

1. Basic statistical functions for data exploration
2. Data Visualization: Box plot, scatter plot, histogram
3. Data Pre-processing: Handling missing values, outliers, normalization, Scaling
4. Principal Component Analysis (PCA)
5. Singular Value Decomposition (SVD)
6. Linear Discriminant Analysis (LDA)
7. Regression Analysis: Linear regression, Logistic regression, Polynomial regression
8. Regularized Regression
9. K-Nearest Neighbour (kNN) Classifier
10. Support Vector Machines (SVMs)
11. Random Forest model
12. AdaBoost Classifier and XGBoost

L	T	P	Cr.
1	0	2	2

Pre-requisite

: Full Stack Development

**Course Outcomes:**

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

**CO1:** Develop professional web pages of an application using HTML elements like lists, navigations, tables, various form elements, embedded media which includes images, audio, video and CSS Styles ( **Apply-L3**)

**CO2:** Build a basic web server using Node.js , Express.js and also working with Node Package Manager (NPM) (**Apply-L3**)

**CO3:** Make use of Typescript to optimize JavaScript code by using the concept of strict type checking. (**Apply-L3**)

**CO4:** Improve individual / teamwork skills, communication & report writing skills with ethical values

**List of Exercises**

1.a	<b>Course Name:</b> HTML5 - The Language
	<b>Module Name:</b> Case-insensitivity, Platform-independency, DOCTYPE Declaration, Types of Elements, HTML Elements - Attributes, Metadata Element
	Include the Metadata element in Homepage.html for providing description as "IEKart's is an online shopping website that sells goods in retail. This company deals with various categories like Electronics, Clothing, Accessories etc.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28320667711144660000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28320667711144660000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a>
1.b	<b>Course Name:</b> HTML5 - The Language
	<b>Module Name:</b> Sectioning Elements
	Enhance the Homepage.html of IEKart's Shopping Application by adding appropriate sectioning elements.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_63722913471108570000_shared?collectionId=lex_177397328348408100000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_63722913471108570000_shared?collectionId=lex_177397328348408100000_shared&amp;collectionType=Course</a>
1.c	<b>Course Name:</b> HTML5 - The Language
	<b>Module Name:</b> Paragraph Element, Division and Span Elements, List Element
	Make use of appropriate grouping elements such as list items to "About Us" page of IEKart's Shopping Application
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_32785192040894940000_shared?collectionId=lex_177397328348408100000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_32785192040894940000_shared?collectionId=lex_177397328348408100000_shared&amp;collectionType=Course</a>
1.d	<b>Course Name:</b> HTML5 - The Language
	<b>Module Name:</b> Link Element
	Link "Login", "SignUp" and "Track order" to "Login.html", "SignUp.html" and "Track.html" page respectively. Bookmark each category to its details of IEKart's Shopping application.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_15515105953273338000_shared?collectionId=lex_177397328348408100000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_15515105953273338000_shared?collectionId=lex_177397328348408100000_shared&amp;collectionType=Course</a>
1.e	<b>Course Name:</b> HTML5 - The Language
	<b>Module Name:</b> Character Entities

	<p>Add the © symbol in the Home page footer of IEKart's Shopping application.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_547667376938096260_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_547667376938096260_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a></p>
1.f	<p><b>Course Name:</b> HTML5 - The Language</p> <p><b>Module Name:</b> HTML5 Global Attributes</p> <p>Add the global attributes such as contenteditable, spellcheck, id etc. to enhance the Signup Page functionality of IEKart's Shopping application.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28723566050321920000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28723566050321920000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a></p>
2.a	<p><b>Course Name:</b> HTML5 - The Language</p> <p><b>Module Name:</b> Creating Table Elements, Table Elements : Colspan/Rowspan Attributes, border, cellspacing, cellpadding attributes</p> <p>Enhance the details page of IEKart's Shopping application by adding a table element to display the available mobile/any inventories.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013168035284033536113_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013168035284033536113_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a></p>
2.b	<p><b>Course Name:</b> HTML5 - The Language</p> <p><b>Module Name:</b> Creating Form Elements, Color and Date Pickers, Select and Datalist Elements</p> <p>Using the form elements create Signup page for IEKart's Shopping application.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13975270903118459000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13975270903118459000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a></p>
2.c	<p><b>Course Name:</b> HTML5 - The Language</p> <p><b>Module Name:</b> Input Elements – Attributes</p> <p>Enhance Signup page functionality of IEKart's Shopping application by adding attributes to input elements.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_14048414537062347000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_14048414537062347000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a></p>
2.d	<p><b>Course Name:</b> HTML5 - The Language</p> <p><b>Module Name:</b> Media, Iframe</p> <p>Add media content in a frame using audio, video, iframe elements to the Home page of IEKart's Shopping application.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_30738402225794945000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_30738402225794945000_shared?collectionId=lex_17739732834840810000_shared&amp;collectionType=Course</a></p>
3.a	<p><b>Course Name:</b> Javascript</p> <p><b>Module Name:</b> Type of Identifiers</p> <p>Write a JavaScript program to find the area of a circle using radius (var and let - reassign and observe the difference with var and let) and PI (const)</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013053264414818304732_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013053264414818304732_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a></p>
3.b	<p><b>Course Name:</b> Javascript</p> <p><b>Module Name:</b> Primitive and Non Primitive Data Types</p> <p>Write JavaScript code to display the movie details such as movie name, starring, language, and ratings. Initialize the variables with values of appropriate types. Use template literals wherever necessary.</p> <p><a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-">https://infyspringboard.onwingspan.com/web/en/viewer/web-</a></p>

	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_21528322245232402000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">module/lex_21528322245232402000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
3.c	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Operators and Types of Operators
	Write JavaScript code to book movie tickets online and calculate the total price, considering the number of tickets and price per ticket as Rs. 150. Also, apply a festive season discount of 10% and calculate the discounted amount.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13808338384481720000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13808338384481720000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
3.d	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Types of Statements, Non - Conditional Statements, Types of Conditional Statements, if Statements, switch Statements
	Write a JavaScript code to book movie tickets online and calculate the total price based on the 3 conditions: (a) If seats to be booked are not more than 2, the cost per ticket remains Rs. 150. (b) If seats are 6 or more, booking is not allowed. (c) If se
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_16257498471333610000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_16257498471333610000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
3.e	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Types of Loops
	Write a JavaScript code to book movie tickets online and calculate the total price based on the 3 conditions: (a) If seats to be booked are not more than 2, the cost per ticket remains Rs. 150. (b) If seats are 6 or more, booking is not allowed. (c) If
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_6238536888292970000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_6238536888292970000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
4.a	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Types of Functions, Declaring and Invoking Function, Arrow Function, Function Parameters, Nested Function, Built-in Functions, Variable Scope in Functions
	Write a JavaScript code to book movie tickets online and calculate the total price based on the 3 conditions: (a) If seats to be booked are not more than 2, the cost per ticket remains Rs. 150. (b) If seats are 6 or more, booking is not allowed. (c) If
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_15455199570613326000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_15455199570613326000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
4.b	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Working With Classes, Creating and Inheriting Classes
	Create an Employee class extending from a base class Person. Hints: (i) Create a class Person with name and age as attributes. (ii) Add a constructor to initialize the values (iii) Create a class Employee extending Person with additional attributes role
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_012599811117760512458_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_012599811117760512458_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
4.c	<b>Course Name:</b> Javascript
	<b>Module Name:</b> In-built Events and Handlers
	Write a JavaScript code to book movie tickets online and calculate the total price based on the 3 conditions: (a) If seats to be booked are not more than 2, the cost per ticket remains Rs. 150. (b) If seats are 6 or more, booking is not allowed. (c) If se
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_4192188372573027000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_4192188372573027000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>



4.d	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Working with Objects, Types of Objects, Creating Objects, Combining and cloning Objects using Spread operator, Destructuring Objects, Browser Object Model, Document Object Model
	If a user clicks on the given link, they should see an empty cone, a different heading, and a different message and a different background color. If user clicks again, they should see a re-filled cone, a different heading, a different message, and a diffe
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13197025862804100000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13197025862804100000_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
5.a	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Creating Arrays, Destructuring Arrays, Accessing Arrays, Array Methods
	Create an array of objects having movie details. The object should include the movie name, starring, language, and ratings. Render the details of movies on the page using the array.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013053270191734784711_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013053270191734784711_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
5.b	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Introduction to Asynchronous Programming, Callbacks, Promises, Async and Await, Executing Network Requests using Fetch API
	Simulate a periodic stock price change and display on the console. Hints: (i) Create a method which returns a random number - use Math.random, floor and other methods to return a rounded value. (ii) Invoke the method for every three seconds and stop when
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_012599811633905664460_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_012599811633905664460_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
5.c	<b>Course Name:</b> Javascript
	<b>Module Name:</b> Creating Modules, Consuming Modules
	Validate the user by creating a login module. Hints: (i) Create a file login.js with a User class. (ii) Create a validate method with username and password as arguments. (iii) If the username and password are equal it will return "Login Successful" else w
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013052857053585408667_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013052857053585408667_shared?collectionId=lex_18109698366332810000_shared&amp;collectionType=Course</a>
6.a	<b>Course Name:</b> Node.js
	<b>Module Name:</b> How to use Node.js
	Verify how to execute different functions successfully in the Node.js platform.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_19002830632103186000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_19002830632103186000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
6.b	<b>Course Name:</b> Node.js
	<b>Module Name:</b> Create a web server in Node.js
	Write a program to show the workflow of JavaScript code executable by creating web server in Node.js.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28177338996267815000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28177338996267815000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
6.c	<b>Course Name:</b> Node.js
	<b>Module Name:</b> Modular programming in Node.js
	Write a Node.js module to show the workflow of Modularization of Node application.

	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28865394191004004000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28865394191004004000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
6.d	<b>Course Name:</b> Node.js
	<b>Module Name:</b> Restarting Node Application
	Write a program to show the workflow of restarting a Node application.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_9174073856000159000_shared?collectionId=lex_32407835671946760000_0_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_9174073856000159000_shared?collectionId=lex_32407835671946760000_0_shared&amp;collectionType=Course</a>
6.e	<b>Course Name:</b> Node.js
	<b>Module Name:</b> File Operations
	Create a text file src.txt and add the following data to it. Mongo, Express, Angular, Node.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_33376440180246100000_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_33376440180246100000_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course</a>
7.a	<b>Course Name:</b> Express.js
	<b>Module Name:</b> Defining a route, Handling Routes, Route Parameters, Query Parameters
	Implement routing for the AdventureTrails application by embedding the necessary code in the routes/route.js file.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_29394215542149950000_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_29394215542149950000_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course</a>
7.b	<b>Course Name:</b> Express.js
	<b>Module Name:</b> How Middleware works, Chaining of Middlewares, Types of Middlewares
	In myNotes application: (i) we want to handle POST submissions. (ii) display customized error messages. (iii) perform logging.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13930661312009580000_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_13930661312009580000_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course</a>
7.c	<b>Course Name:</b> Express.js
	<b>Module Name:</b> Connecting to MongoDB with Mongoose, Validation Types and Defaults
	Write a Mongoose schema to connect with MongoDB.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035588775485440691_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035588775485440691_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course</a>
7.d	<b>Course Name:</b> Express.js
	<b>Module Name:</b> Models
	Write a program to wrap the Schema into a Model object.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035593896869888662_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035593896869888662_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course</a>
8.a	<b>Course Name:</b> Express.js
	<b>Module Name:</b> CRUD Operations
	Write a program to perform various CRUD (Create-Read-Update-Delete) operations using Mongoose library functions.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035684270129152696_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035684270129152696_shared?collectionId=lex_32407835671946760000_00_shared&amp;collectionType=Course</a>
8.b	<b>Course Name:</b> Express.js

	<b>Module Name:</b> API Development In the myNotes application, include APIs based on the requirements provided. (i) API should fetch the details of the notes based on a notesID which is provided in the URL. Test URL - <a href="http://localhost:3000/notes/7555">http://localhost:3000/notes/7555</a> (ii) API should update the details bas <a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035745250975744755_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_auth_013035745250975744755_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
8.c	<b>Course Name:</b> Express.js <b>Module Name:</b> Why Session management, Cookies Write a program to explain session management using cookies. <a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24299316914857090000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24299316914857090000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
8.d	<b>Course Name:</b> Express.js <b>Module Name:</b> Sessions Write a program to explain session management using sessions. <a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_905413034723449100_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_905413034723449100_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
8.e	<b>Course Name:</b> Express.js <b>Module Name:</b> Why and What Security, Helmet Middleware Implement security features in myNotes application <a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_31677453061177940000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_31677453061177940000_shared?collectionId=lex_32407835671946760000_shared&amp;collectionType=Course</a>
9.a	<b>Course Name:</b> Typescript <b>Module Name:</b> Basics of TypeScript On the page, display the price of the mobile-based in three different colors. Instead of using the number in our code, represent them by string values like GoldPlatinum, PinkGold, SilverTitanium. <a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28910354929502245000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_28910354929502245000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
9.b	<b>Course Name:</b> Typescript <b>Module Name:</b> Function Define an arrow function inside the event handler to filter the product array with the selected product object using the productId received by the function. Pass the selected product object to the next screen. <a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_10783156469383723000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_10783156469383723000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
9.c	<b>Course Name:</b> Typescript <b>Module Name:</b> Parameter Types and Return Types Consider that developer needs to declare a function - getMobileByVendor which accepts string as input parameter and returns the list of mobiles. <a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712912427057152901_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712912427057152901_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
9.d	<b>Course Name:</b> Typescript <b>Module Name:</b> Arrow Function Consider that developer needs to declare a manufacturer's array holding 4 objects with id and price as a parameter and needs to implement an arrow function - myfunction to populate the id parameter of manufacturers array whose price is greater than or equ

	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712910875500544904_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712910875500544904_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
9.e	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Optional and Default Parameters
	Declare a function - getMobileByManufacturer with two parameters namely manufacturer and id, where manufacturer value should be passed as Samsung and id parameter should be optional while invoking the function, if id is passed as 101 then this function should
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712914940641280906_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712914940641280906_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
10.a	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Rest Parameter
	Implement business logic for adding multiple Product values into a cart variable which is type of string array.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712921860915200909_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712921860915200909_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
10.b	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Creating an Interface
	Declare an interface named - Product with two properties like productId and productName with a number and string datatype and need to implement logic to populate the Product details.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712925244276736910_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712925244276736910_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
10.c	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Duck Typing
	Declare an interface named - Product with two properties like productId and productName with the number and string datatype and need to implement logic to populate the Product details.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712925995458560912_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712925995458560912_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
10.d	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Function Types
	Declare an interface with function type and access its value.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712948945346560918_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712948945346560918_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
11.a	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Extending Interfaces
	Declare a productList interface which extends properties from two other declared interfaces like Category, Product as well as implementation to create a variable of this interface type.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712951652139008920_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712951652139008920_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
11.b	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Classes
	Consider the Mobile Cart application, Create objects of the Product class and place them into the productList array.

	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_3705824317381604400_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_3705824317381604400_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
11.c	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Constructor
	Declare a class named - Product with the below-mentioned declarations: (i) productId as number property (ii) Constructor to initialize this value (iii) getProductId method to return the message "Product id is <<id value>>".
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712954616782848927_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712954616782848927_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
11.d	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Access Modifiers
	Create a Product class with 4 properties namely productId, productName, productPrice, productCategory with private, public, static, and protected access modifiers and accessing them through Gadget class and its methods.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712953517170688931_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/hands-on/lex_auth_012712953517170688931_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
12.a	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Properties and Methods
	Create a Product class with 4 properties namely productId and methods to setProductId() and getProductId().
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_9356738095572543000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_9356738095572543000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
12.b	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Creating and using Namespaces
	Create a namespace called ProductUtility and place the Product class definition in it. Import the Product class inside productlist file and use it.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_20787271128051925000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_20787271128051925000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
12.c	<b>Course Name:</b> Typescript
	<b>Module Name:</b> Creating and using Modules
	Consider the Mobile Cart application which is designed as part of the functions in a module to calculate the total price of the product using the quantity and price values and assign it to a totalPrice variable.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24788158187785620000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_24788158187785620000_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>
12.d	<b>Course Name:</b> Typescript
	<b>Module Name:</b> What is Generics, What are Type Parameters, Generic Functions, Generic Constraints
	Create a generic array and function to sort numbers as well as string values.
	<a href="https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_446287045482942800_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course">https://infyspringboard.onwingspan.com/web/en/viewer/web-module/lex_446287045482942800_shared?collectionId=lex_9436233116512678000_shared&amp;collectionType=Course</a>

## Software configuration and installation

### 1. HTML & Javascript

- Simple editors such as Notepad or go for IDEs like Visual Studio Code(recommended), Eclipse etc. which makes coding easier.
- And, to execute application, you can use any commonly used browser such as Google Chrome(recommended), Mozilla Firefox etc
- Setup details: [Environmental Setup for HTML5 - Viewer Page | Infosys Springboard \(onwingspan.com\)](#)
- [Environment Setup: Internal - Viewer Page | Infosys Springboard \(onwingspan.com\)](#)

2. Node JS

Download **Node.js** from the official site

Setup details : [How to use Node.js - Viewer Page | Infosys Springboard \(onwingspan.com\)](#)

3. Typescript

[Installing TypeScript - Internal - Viewer Page | Infosys Springboard \(onwingspan.com\)](#)

**Text Books:**

1. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson.
2. Pro Mean Stack Development, 1st Edition, ELadElrom, Apress O'Reilly.
3. Full Stack JavaScript Development with MEAN, Colin J Ihrig, Adam Bretz, 1st edition, SitePoint, SitePoint Pty. Ltd., O'Reilly Media.

**Reference Books:**

1. Web Technologies, HTML, JavaScript, PHP, Java, JSP, XML and AJAX, Black book, 1st Edition, Dream Tech.
2. An Introduction to Web Design, Programming, 1st Edition, Paul S Wang, Sanda S Katila, Cengage Learning.

**Web Links:**

1. [https://infyspringboard.onwingspan.com/en/app/toc/lex\\_17739732834840810000\\_shared/overview](https://infyspringboard.onwingspan.com/en/app/toc/lex_17739732834840810000_shared/overview) (HTML5)
2. [https://infyspringboard.onwingspan.com/en/app/toc/lex\\_18109698366332810000\\_shared/overview](https://infyspringboard.onwingspan.com/en/app/toc/lex_18109698366332810000_shared/overview) (Javascript)
3. [https://infyspringboard.onwingspan.com/en/app/toc/lex\\_32407835671946760000\\_shared/overview](https://infyspringboard.onwingspan.com/en/app/toc/lex_32407835671946760000_shared/overview) (Node.js & Express.js)
4. [https://infyspringboard.onwingspan.com/en/app/toc/lex\\_9436233116512678000\\_shared/overview](https://infyspringboard.onwingspan.com/en/app/toc/lex_9436233116512678000_shared/overview) (Typescript)



B.Tech. (VI Sem.)

**20CS17-INFORMATION SECURITY**

L	T	P	Cr.
3	0	0	3

**Pre-requisite** : Computer Networks, Number theory and programming language

**Course Educational Objective:** The Objective of the course is to course elevates the security aspects and provides the knowledge to understand the basic concept of Cryptography and Network Security principles. It antilight 's different types of cipher mechanisms and various symmetric and asymmetric algorithms. Also provides the knowledge on digital signatures, different threats, viruses, intruders, and firewalls.

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

**CO1:** Summarize encryption algorithms to achieve data confidentiality. (**Understand-L2**)

**CO2:** Apply Secure hash functions for attaining data integrity. (**Apply-L3**)

**CO3:** Illustrate the email security mechanisms to achieve authentication. (**Understand- L2**)

**CO4:** Demonstrate the techniques of web security. (**Understand-L2**)

**CO5:** Examine the threats and remedial measures for system security. (**Apply-L3**)

**UNIT - I: INTRODUCTION:**

Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, Access Control and Availability) and Mechanisms. A Model for Internetwork security, Conventional Encryption Principles, Conventional Encryption Algorithms (DES, Triple DES and AES), Cipher Block Modes of Operations (CBC, CFB only), Stream Ciphers and RC4, Location of Encryption Devices, Key Distribution.

**UNIT - II: PUBLIC -KEY CRYPTOGRAPHY:**

Approaches of Message Authentication, MD5, Secure Hash Functions (SHA-512) and HMAC Algorithm, Public Key Cryptography principles, Public Key Cryptography Algorithms, Digital Signatures, Public Key Infrastructure, Digital Certificates, Certificate Authority, Key Management, X.509 certificate.

**UNIT - III: EMAIL PRIVACY:**

Email privacy, Pretty Good Privacy (PGP), PGP and S/MIME, IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange, cryptographic Suites.

**UNIT - IV: WEB SECURITY:**

Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET) and HTTPs.

**UNIT - V: INTRUDERS:**

Intruders: Intruder Behavior Patterns, Intrusion Techniques, Statistical Anomaly Detection, Rule-Based Intrusion Detection, Honeypot, Malicious Software: Backdoor, Logic Bomb, Trojan Horses, Mobile Code, Viruses: The Nature of Viruses, Viruses Classification, Antivirus Approaches, Distributed Denial of Service Attacks, DDoS Countermeasures Firewall Design principles, Trust Management System, Introduction to digital forensics.

**TEXTBOOKS:**

1. William Stallings, "Network Security Essentials (Applications and Standards)", Pearson Education, 1999.
2. William Stallings, "Cryptography and Network Security", PHI/Pearson, fourth edition, 2000.

**REFERENCE BOOKS:**

1. Whitman, "Principles of Information Security", Thomson, PHI, 2000
2. Robert Bragg, Mark Rhodes, Network Security: The complete reference, TMH
3. Buchmann, Introduction to Cryptography, Springer, 2010.
4. Jeetendra pande, Ajay, "Digital Foresnsics", ISBN: 978-93-84813-94-9
5. <https://www.mat.unical.it/ianni/storage/Intro-2013.ppt>
6. <http://www.inf.ufsc.br/~bosco.sobral/ensino/ine5680/material-cripto-seg/2014>
7. <http://www.williamstallings.com/netsec/netsec4e.html>
8. <https://www.cs.kau.se/cs/dvgc19/ch14.pdf>
9. <https://www.ntut.edu.tw/~jhwang/IS>



B.Tech. (VI Sem.)

20CS18-COMPILER DESIGN

L	T	P	Cr.
3	0	0	3

**Pre-requisite :** Programming language, and Automata theory and formal languages

**Course Educational Objective:** The objective of the course is to understand and list the different stages in the process of compilation, identify different methods of lexical analysis, design top-down and bottom-up parsers, develop syntax directed translation schemes, and develop algorithms to generate code for a target machine.

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

- CO1:** Design the lexical analyzer using LEX tool. (**Apply- L3**)
- CO2:** Design a parser using top-down and bottom-up parser design methods. (**Apply- L3**)
- CO3:** Construct syntax-directed translator for semantic checking and intermediate code generation Using YACC tool (**Apply –L3**)
- CO4:** Demonstrate the machine dependent and machine independent code optimization techniques. (**Understand-L2**)
- CO5:** Understand the design issues of the code generator and run-time environment of the program. (**Understand-L2**)

**UNIT - I**

**Introduction to Compiler:** Language Processing System, Compiler vs Interpreter, The Phases of Compiler, Compiler Construction Tools, and Boot Strapping.

**Lexical Analysis:** The role of lexical analyzer, Input buffering, Specification of tokens, Recognition of Tokens, A language for specifying lexical analyzer (LEX).

**UNIT - II**

**Syntax Analyzer:** The role of parser, writing a Grammar-Elimination of Left recursion, Left factoring.

**Top-Down Parsing** –Recursive descent parsing, Predictive parsing, Pre-processing steps required for predictive parsing, LL (1) grammar, Error recovery in predictive parsing, Backtracking.

**UNIT - III**

**Bottom-Up parsing:** Handle Pruning, Shift-Reduce parsing, and Operator Precedence Parsing. LR parsers-SLR, CLR and LALR, Handling ambiguous grammars, Error recovery in LR parsing, YACC – automatic parser generator.

**UNIT - IV**

**Syntax Directed Translation:** Syntax-directed Definitions , Evaluation order of SDD's, Applications of SDD and Syntax directed Translation Schemes.

**Intermediate code generation:** - Syntax tree, polish notation and three address code, static single assignment. Translation of expressions and control flow statements, and Boolean Expressions.

**Run time storage:** Storage organization, storage allocation strategies, and parameter passing techniques.

**UNIT - V**

**Code Optimization:** Basic Blocks and flow graphs, principal sources of optimization, loop optimization, Introduction to Data-flow analysis.

**Code generation:** Design issues, object code forms, optimization of Basic blocks- DAG representation of basic blocks and code generation using DAG, A simple code generator, peephole optimization, and Register allocation and assignment.

**TEXTBOOK:**

1. Alfred V.Aho, Jeffrey Ullman, Ravi sethi, “Compilers Principles, Techniques and Tools”, Pearson Education, 2nd Edition, 2008.

**REFERENCE BOOKS:**

1. ParagH.Dave, HimanshuB.Dave ,“Compilers Principles and Practice” , Person Education, First Edition, 2012.
2. Andrew W.appel ,“Modern compiler implementation in C” ,Cambridge, Revised Edition, 2010.
3. <http://nptel.ac.in/courses/106108052/> (Video lectures for Compiler design)

L	T	P	Cr.
3	0	0	3

**Pre-requisite:** Database Management Systems, Data Warehousing and Data Mining

**Course Educational Objective:** Understanding the process of distributed data (Structured, Semi-Structured and Unstructured) that process the Terabytes of data using Hadoop Eco System Tools.

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

**CO1:** Identify Big Data and its Business Implications. (**Understand - L2**)

**CO2:** Process of distributed file system using Hadoop(**Apply - L3**)

**CO3:** Illustrate the MapReduce mechanism (**Apply - L3**)

**CO4:** Develop structured data processing tools (**Apply- L3**)

**CO5:** Develop semi/unstructured data processing tools (**Apply – L3**)

### UNIT – I: Introduction to Big data

Types of Digital Data, Classification of Digital Data, Characteristics of Data, Evolution of Big Data, Definition of Big Data, Challenges with Big Data, What is Big Data?, Other Characteristics of Data Which are not Definitional Traits of Big Data, Why Big Data?, analyzing Data with Unix tools, Analyzing Data with Hadoop, Hadoop Streaming, Hadoop Echo System.

### UNIT – II: Hadoop Distributed File System

The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingestion with Sqoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures.

### UNIT – III: MapReduce Technique

How MapReduce works?, Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features.

### UNIT – IV: Structured Data Processing Tools

**Hive:** Installation, Running Hive, HiveQL, Tables, Querying Data, User Defined functions

**Sqoop:** Introduction, generate code, Database import, working with imported data, Importing large objects, performing an exports

### UNIT – V: Semi-structured and unstructured Data Processing Tools

**Pig:** Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators.

**HBase:** Basics, Concepts, Clients, Example, HBase Versus RDBMS..

### TEXT BOOKS:

1. Tom White "Hadoop: The Definitive Guide" Third Edit, O'reily Media, 2012.
2. Big Data and Analytics, 2ed Seema Acharya, Subhashini Chellappan, Wiley 2015.

**REFERENCE BOOKS:**

1. Michael Berthold, David J. Hand, "Intelligent Data Analysis", Springer, 2007.
2. Jay Liebowitz, "Big Data and Business Analytics" Auerbach Publications, CRC press (2013)
3. Tom Plunkett, Mark Hornick, "Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop", McGraw-Hill/Osborne Media (2013), Oracle press.
4. Anand Rajaraman and Jeffrey David Ulman, "Mining of Massive Datasets", Cambridge University Press, 2012.
5. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", John Wiley & sons, 2012.
6. Glen J. Myat, "Making Sense of Data", John Wiley & Sons, 2007
7. Pete Warden, "Big Data Glossary", O'Reily, 2011.
8. Michael Mineli, Michele Chambers, Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", Wiley Publications, 2013.
9. ArvindSathi, "BigDataAnalytics: Disruptive Technologies for Changing the Game", MC Press, 2012
10. Paul Zikopoulos ,Dirk DeRoos , Krishnan Parasuraman , Thomas Deutsch , James Giles, David Corigan, "Harness the Power of Big Data The IBM Big Data Platform", Tata McGraw Hill Publications, 2012.

B.Tech. ( VI Sem.)

20CS20-PATTERN RECOGNITION

L	T	P	Cr.
3	0	0	3

Pre-requisite : Basic knowledge of probability & statistics, data mining.

**Course Educational Objective:** The main objective of the course is that the concept of a pattern and the fundamentals of pattern recognition and its relevance to classical and modern problems and to be able to identify where, when and how pattern recognition can be applied.

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

**CO1:** Understand the primitives of Pattern Recognition. (Understand- L2)

**CO2:** Understand the fundamental concepts of Bayesian approach (Understand – L2)

**CO3:** Understand Bayesian classifier with respect to its parameter estimation. (Understand- L2)

**CO4:** Gain knowledge of Mixture densities and clustering techniques. (Understand- L2)

**CO5:** Develop a statistical framework for Speech Recognition. (Apply - L3)

**UNIT-I**

**Introduction:** Machine perception, pattern recognition example, pattern recognition systems, the Design cycle, learning and adaptation

**Bayesian Decision Theory:** Introduction, continuous features – two categories classifications, minimum error-rate classification-zero-one loss function, classifiers, discriminant functions, and decision surface

**UNIT- II**

**Normal density:** Univariate and multivariate density, discriminant functions for the normal Density different cases, Bayes decision theory – discrete features, compound Bayesian decision theory and context

**UNIT- III**

**Maximum likelihood and Bayesian parameter estimation:** Introduction, maximum likelihood Estimation, Bayesian estimation, Bayesian parameter estimation–Gaussian case.

**UNIT- IV**

**Un-supervised learning and clustering:** Introduction, mixture densities and identifiability, maximum likelihood estimates, application to normal mixtures, K-means clustering. Data description and clustering – similarity measures, criteria function for clustering.

**UNIT- V**

**Pattern recognition using discrete hidden Markov models:** Discrete-time Markov process, Extensions to hidden Markov models, three basic problems of HMMs, Types of HMMs

**TEXT BOOKS:**

1. Pattern classifications, Richard O. Duda, Peter E. Hart, David G. Stroke. Wiley's student edition, Second Edition, 2004.
2. Pattern Recognition, an Introduction, V Subshell Devi, M Narsimha Murthy, Universiy Press, 2010.
3. Fundamentals of speech Recognition, Lawrence Rabiner, Biing – Hwang Juang Pearson education.

**REFERENCE BOOKS:**

1. R.C Gonzalez and R.E. Woods, "Digital Image Processing", Addison Wesley, 1992.
2. Earl Gose, Richard John baugh, Steve Jos, "Pattern Recognition and Image Analysis", PHI 2004.
3. SergiosTheodoridis, Konstantinos Koutroumbas, "Pattern Recognition", Academic Press, Elsevier, 4ed.

**Online Courses and Video Lectures:**

1. <http://freevideolectures.com/Course/3194/Pattern-Recognition>
2. <http://nptel.ac.in/courses/117105101/>
3. <http://freevideolectures.com/Course/3530/Pattern-Recognition-I>

B.Tech. (VI Sem.)

20CS21-INFORMATION RETRIEVAL SYSTEMS

L	T	P	Cr.
3	0	0	3

**Pre-requisite** : DBMS, Data Structures, and DW&DM

**Course Educational Objective:** The main objective of this course is to present the basic concepts in information retrieval and the significance of various indexing and searching techniques for information retrieval.

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

- CO 1 Identify the basic concepts of Information retrieval system.. (**Remember – L1**)
- CO 2 Describe the taxonomy of different information retrieval models.(**Understand-L2**)
- CO 3 Demonstrate and evaluate automatic indexing, document & term clustering techniques. (**Understand-L2**)
- CO 4 Demonstrate and evaluate various searching techniques. (**Understand- L2**)
- CO 5 Evaluate text processing techniques and operations in information retrieval system. (**Apply- L3**)

**UNIT I**

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

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

**UNIT II**

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

**Cataloguing and Indexing:** Objectives, Indexing Process, Automatic Indexing, and Information Extraction.

**UNIT III**

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

**Kowalski, Gerald, Mark T Maybury — Information Storage & Retrieval Systems: Theory and Implementation**, Kluwer Academic Press, 2nd edition, 2002.

**Document and Term Clustering:** Introduction, Thesaurus generation, Item clustering, Hierarchy of clusters.

**UNIT IV**

**User Search Techniques:** Search statements and binding, Similarity measures and ranking, Relevance feedback, Selective dissemination of information search, weighted searches of Boolean systems, Searching the Internet and hypertext.

**Information Visualization:** Introduction, Cognition and perception, Information visualization technologies.

**UNIT V**

**Text Search Algorithms:** Introduction, Software text search algorithms, Hardware text search systems.

**Information System Evaluation:** Introduction, Measures used in system evaluation, Measurement example – TREC results.

**TEXT BOOK**

1. Kowalski, Gerald, Mark T Maybury – Information Storage & Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 2<sup>nd</sup> edition, 2002.

**REFERENCE BOOKS**

1. C. D. Manning, P. Raghavan, and H. Schutze, An Introduction to Information Retrieval, Cambridge University Press, 2009.
2. William B Frakes, Ricardo Baeza-Yates, “Information Retrieval Data Structures and Algorithms”, Pearson Education, 1992.
3. <https://epdf.tips/queue/information-storage-and-retrieval-systems-theory-and-implementation-the-informat.html>
4. Robert Korthagen, “Information Storage & Retrieval”, John Wiley & Sons, 3<sup>rd</sup> Edition, 2011.



**B.Tech.(VI Sem) 20CS22- DISTRIBUTED OPERATING SYSTEMS**

L	T	P	Cr.
3	0	0	3

**Pre-requisites:** Knowledge of Operating systems, Computer Networks

**Course Educational Objectives:**

This course enables the students to know about a comprehensive introduction to understand the underlying principles, techniques and approaches which constitute a coherent body of knowledge in operating systems. In particular, the course will consider inherent functionality and processing of program execution. The emphasis of the course will be placed on understanding how the various elements that underlie operating system interact and provides services for execution of application software.

**Course Outcomes (COs):** At the end of this course, the student will be able to

- CO1:** Identify the hardware and software components to design the communication model in Distributed System. **(L2-Understand)**
- CO2:** Illustrate the processor allocation and process scheduling algorithms in Distributed Systems. **(L2-Understand)**
- CO3:** Apply the Clock Synchronization protocols and Deadlock handling mechanisms in Distributed system. **(L3-Apply)**
- CO4:** Analyze the implementation of Distributed Shared memory for real world problems. **(L2-Understand)**
- CO5:** Demonstrate the Implementation of Distributed file system and CHROUS, MACH distributed operating systems. **(L3-Apply)**

**Unit-1: Introduction to Distributed Systems**

Definition, Goals, Hardware Concepts, Software Concepts, Design issues. Communication in Distributed Systems: Layered Protocols, ATM networks, Client-Server model, Remote Procedure Call, Group Communication.

**Unit-2: Processes and Processors in Distributed Systems**

Threads, System Models, Processor Allocation, Scheduling in Distributed Systems, Fault tolerance, Real time Distributed Systems.

**Unit-3: Synchronization in Distributed Systems**

Clock Synchronization, Mutual Exclusion, Election Algorithms, Atomic Transactions, Deadlocks in Distributed Systems.

**Unit-4: Distributed Shared Memory**

Introduction, Consistency Models, Page-based Distributed shared memory, Shared-variable Distributed shared memory, Object-based Distributed shared memory.

**Unit-5: Distributed File Systems**

Distributed File System design, Distributed File System implementation, Trends in Distributed File Systems.

**Case Studies:** MACH, CHORUS

**TEXT BOOKS**

1. T Andrew S Tanenbaum, “Distributed Operating Systems” , 3rd edition, Pearson publication, 2007
2. George Coulouris, Jean Dollimore and Tim Kindberg, “Distributed Systems- Concepts and Design” , 5<sup>th</sup> Edition, Pearson Publications, 2009

**REFERENCE BOOKS**

1. Ajay D. Kshemkalyani and MukeshSinghal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2008.
2. SunitaMahajan, Seema Shan, Distributed Computing, Oxford University Press, 2015
3. Web link: <https://dinus.ac.id/repository/docs/ajar/George-Coulouris-Distributed-Systems-Concepts-and-Design-5th-Edition.pdf>

B.Tech. (VI Sem.)

20CS61-INFORMATION SECURITY LAB

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Computer Programming, Mathematics, Computer Networks

**Course Educational Objective:** The Objective of the course is to provide practical level training that enables the usage of security algorithms for handling security projects. It also provides foundation level training for network analysis to find out the intruders in a network traffic.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Demonstrate the security concepts, type of attacks and network security algorithms. (**Understand - L2**)
- CO2:** Apply symmetric and asymmetric cryptography technique to encrypt and decrypt text. (**Apply - L3**)
- CO3:** Apply network-based tools for network analysis. (**Apply - L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

## List of Experiments

1. Implement any two Substitution Techniques.
2. Implement any two Transposition Techniques
3. Implement any two Symmetric algorithms.
4. Implement any two Private -Key based algorithms.
5. Explore any four network diagnosis tools.
6. Study about Wireshark packet sniffer tool in promiscuous and non-promiscuous mode.
7. Download and install nmap. Use it with different options to scan open ports, do a ping scan, tcp port scan, udp port scan.
8. Iptables in linux.
9. Demonstrate intrusion detection system (ids) using any tool (snort or any other s/w).

## TEXT BOOKS

1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education, Second Edition.
2. **Snort 2.1 Intrusion Detection** by Jay Beale, Andrew R.Baker , Second Edition.
3. **Wireshark Network Analysis** by Laura Chappell, Second Edition.

## REFERENCES

1. William Stallings, "Cryptography and Network security", Pearson Education, Fourth Edition.
2. <https://www.snort.org/documents/snort-3-1-0-0-on-ubuntu-18-20>.
3. <https://www.wireshark.org/download/docs/user-guide.pdf>

B.Tech. (VI Sem.)

**20CS62- DATA ANALYTICS AND  
VISUALIZATION LAB**

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : Object Oriented programming & Database Management Systems

**Course Educational Objective:** The Objective of the course is to provide practical, foundation level training that enables immediate and effective participation in Big Data and other Analytics projects using Hadoop and Data Visualization using Tableau.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

**CO1:** Demonstrate the installation of Bigdata analytic tools. (**Understand – L2**)

**CO2:** Apply data modelling techniques to large data sets. (**Apply – L3**)

**CO3:** Conduct exploratory data analysis using visualization. (**Understand – L2**)

**CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

## **List of Experiments**

1. Refreshing Linux Commands and Installation of Hadoop
2. Implementation of Run a basic Word Count Map Reduce program
3. Implementation of Matrix Multiplication with Hadoop Map Reduce.
4. Implementation of Weather mining by taking weather data set using Map Reduce.
5. Installation of Hive along with practice examples.
6. Installation of Sqoop along with Practice examples.
7. Downloading and installing Tableau Understanding about importing data, saving, opening, and sharing workbooks.
8. Data Preparation with Tableau
9. Charts: Bar Charts, Legends, Filters, and Hierarchies, Step Charts, Line Charts.
10. Maps: Symbol Maps, Filled Maps, Density Maps, Maps with Pie Charts
11. Interactive Dashboards

## **TEXTBOOKS:**

1. Seema Acharya, Subhasini Chellappan, "**Big Data Analytics**" Wiley 2015
2. Alexander Loth, "**Visual Analytics with Tableau**", ISBN: 978-1-119-56020-3, Wiley 2019

## **REFERENCES:**

1. Tom White, "**Hadoop: The Definitive Guide**", Third Edit on, O'reily Media, 2012
2. Michael Berthold, David J. Hand, "**Intelligent Data Analysis**" Springer, 2007.
3. Jay Liebowitz, "**Big Data and Business Analytics**" Auerbach Publications, CRC press (2013).
4. AnandRajaraman and Jeffrey David Ulman, "**Mining of Massive Datasets**", Cambridge University Press, 2012.
5. ArvindSathi, ". **BigDataAnalytics: Disruptive Technologies for Changing the Game**", MC Press, 2012, 2001.
6. <http://nptel.ac.in/courses/106104135/48>
7. <http://nptel.ac.in/courses/110106064/>

B.Tech. (VI Sem.)

20CS63-SERVER-SIDE SCRIPTING LAB

L	T	P	Cr.
0	0	3	1.5

**Pre-requisite** : JAVA Programming & Web Technologies

**Course Educational Objective:** Students will learn the importance of client-server architecture in the web application development and able to develop dynamic data driven (server-side) web applications by using advanced technologies (AJAX, PHP)

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

- CO1:** Understand the differences between server-side and client-side script, develop simple server-server-side web applications. (**Understand - L2**)
- CO2:** Identify the importance of AJAX, PHP programming constructs to design server-side web applications. (**Remember - L1**)
- CO3:** Develop Dynamic Data Driven (Server-side) Web Applications by using AJAX, PHP. (**Apply - L3**)
- CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Module 1:**

- a) Develop a static web application and deploy it in any one of the web servers (WAMP/Apache Tomcat/IIS).
- b) Develop a JavaScript program to validate the client-side user input data (Example: username, password, email, phone number).

**Module 2:**

- a) Create a XMLHttpRequest with a callback function and retrieve data from a TXT file by using AJAX.
- b) Create an XMLHttpRequest to retrieve data from an XML file and display the data in an HTML table by using AJAX.

**Module 3:**

- a) Develop AJAX application to demonstrate how a web page can communicate with a web server while a user type characters in an input field.
- b) Develop a simple web application to demonstrate how a web page can fetch information from a database with AJAX.

**Module 4:**

- a) Develop a PHP program to illustrate the PHP Form handling by using GET and POST methods.
- b) Develop a PHP program to demonstrate the importance of include() and require() functions.

**Module 5:**

- a) Develop a PHP program to manage the state information about the user by using PHP Cookie.
- b) Develop a PHP program to manage the state information about the user by using PHP Session.

**Module 6:**

- a) Develop a PHP program to perform various file handling operations.
- b) Develop a PHP program to demonstrate the process of uploading and downloading the file.

**Module 7:**

- a) Develop a PHP program to describe the importance of mail function to send the email.
- b) Develop a PHP and AJAX program which demonstrates how a web page can communicate with a web server while a user types characters in an input field.

**Module 8:**

- a) Develop a PHP web application which demonstrates the process of storing HTML form data into MySQL database.
- b) Develop a PHP web application which demonstrates the process of retrieving table data from the MySQL database and display it in the HTML table.



**Module 9:**

- a) Develop a PHP web application which demonstrates the process of deleting a particular record in MySQL database table.
- b) Develop a PHP web application to illustrate the process of updating MySQL database table data.

**Module 10:**

- a) Develop a PHP web application which demonstrates how to upload image into database and display it.
- b) Develop a PHP web application which demonstrates the process of encode and decode JSON by the help of json\_encode() and json\_decode() functions.

**REFERENCE BOOKS:**

1. Steven Holzner, “PHP: The Complete Reference”, McGraw-Hill Education, 2007.
2. Kevin Tatroe, Peter MacIntyre, RasmusLerdorf, “Programming in PHP”, O'Reilly, 3<sup>rd</sup> Edition, 2013.
3. Luke Welling, Laura Thomson, “PHP and MySQL Web Development”, Pearson Education, 5<sup>th</sup> Edition, 2011
4. Lynn Beighley, Michael Morrison, “Head First PHP & MySQL: A Brain-Friendly Guide”, O'Reilly, 1<sup>st</sup> Edition, 2007

L	T	P	C
1	0	2	2

**Course Educational Objectives:**

The Soft Skills Laboratory course equips students with required behavioral, interpersonal & Intrapersonal skills, communication skills, leadership skills etc. It aims at training undergraduate students on soft skills leading to enhanced self-confidence, esteem, and acceptability in professional circles.

**Course Outcomes (COs):** At the end of the course, student will be able

**CO1:** To Develop self-awareness and personality traits for professional growth  
(Understand – L2)

**CO2:** Work effectively in multi-disciplinary and heterogeneous teams through knowledge of teamwork, Inter-personal relationships, conflict management and leadership quality.  
(Apply – L3)

**CO3:** Communicate through verbal/oral communication with good listening skills and empathy  
(Apply – L3)

**CO4:** Apply skills required to qualify in recruitment tests, Interviews & other professional assignments (Apply – L3)

**Personality Development Skills**

Role of language in Personality – How language reflects, impacts Personality – Using gender-neutral language in MNCs – being culturally-sensitive-Personality Traits - Grooming & Dress code

Activities: Group Discussion/Role play/Presentations (authentic materials: News papers, pamphlets and news clippings)

**Impactful Communication**

Activities : Extempore / Story Telling/ Group Discussion ( Case studies/Current affairs etc.)/ Elocution on Interpretation of given quotes/ Critical Appreciation and Textual Analysis/ Writing reviews on short story/videos/book/Social Media profiling/ Pronunciation Practice

**Professional Skills:**

Career Planning- job vs. career- goal setting- SWOT analysis-Time management – self-management – stress-management.

Activities: SWOT analysis of the self/Goal setting-Presentation/Writing Report/Listening exercises/Effective Resume-Writing and presentation/ Interview Skills: Mock interviews/Video samples.

**REFERENCES :**

1. Edward Holffman, “Ace the Corporate Personality”, McGraw Hill,2001
2. Adrian Furnham, Personality and Intelligence at Work, Psyc 2. hology Press, 2008.
3. M.Ashraf Rizvi, “Effective Technical Communication”, 1 st edition, Tata McGraw Hill, 2005
4. Ace of Soft skills Gopalaswamy Ramesh, Pearson Education India, 2018
5. Soft Skills for the Workplace, Goodheart-Willcox Publisher · 2020.
6. How to Win Friends and Influence People, Dale Carnegie · 2020

**B.Tech. (VII Sem.) 20CS23-SOFTWARE TESTING METHODOLOGIES**

L	T	P	Cr.
3	0	0	3

Pre-requisite : Software Engineering and UML

**Course Educational Objective:** To know the importance of automation testing compared with manual testing by identifying the risks associated with bugs, using different testing approaches like path, Transaction flow, Domain, logic-based testing, and finally converting flow graphs into graph matrices with associated operations for the node reduction.

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

- CO1:** Analyze the purpose, model of testing and consequences due to bugs. **(Understand-L2)**
- CO2:** Demonstrate control flow graph with complete Path testing and Transaction flow testing for detecting anomalies. **(Understand-L2)**
- CO3:** Illustrate domain testing strategies for different domains. **(Understand-L2)**
- CO4:** Apply reduction procedure to control flow graphs and path expressions. **(Apply-L3)**
- CO5:** Demonstrate Transition testing in state graphs and graph matrices for node reduction. **(Understand-L2)**

**UNIT – I**

**Introduction:** Purpose of Testing, Dichotomies, model for testing, consequences of bugs, Taxonomy of bugs.

**UNIT – II**

**FlowGraphs and Path testing:** Basic concepts, Predicates, Path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

**Transaction flow testing:** Transaction flows, transaction flow testing techniques. Data flow testing: Basics of Data flow testing, strategies in dataflow testing, application of dataflow testing.

**UNIT – III**

**Domain Testing:** Domains and paths, Nice and ugly domains, domain testing, domains and interfaces testing, domains and testability.

**UNIT – IV**

**Paths, path products and Regular expressions:** Path products & Path expression, reduction procedure, applications, regular expressions and 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 and 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. A Boris Biezer; "Software Testing Techniques"; International Thomson computer Press, 2<sup>nd</sup> Edition, 2006.

**REFERENCE BOOKS:**

1. Brain Marick; "The Craft of Software Testing", Prentice Hall Series in innovative technology, 2007.
2. RenuRajani, Pradeep Oak, "Software Testing, Effective methods, Tools and Techniques", TMHI, 2<sup>nd</sup> Edition, 2009.
3. Dr.K.V.K.K.Prasad, "Software Testing Tools", Dreamtech Software Testing in the Real World, Edward Kit, Pearson, 2014
4. Perry, "Effective methods of Software Testing", John Wiley, 3<sup>rd</sup> Edition, 2011.
5. <https://www.youtube.com/watch?v=gPE9emPFrwo>
6. <https://freevideolectures.com> > Computer Science > IIT Bombay

B.Tech. (VII Sem.)

20CS24-CLOUD COMPUTING

L	T	P	Cr.
3	0	0	3

**Pre-requisite: Computer networks, and Operating Systems.**

**Course Objectives:**

- To explain the evolving computer model and cloud computing.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.
- To motivate students to do programming and experiment with the various cloud computing environments.

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

- Illustrate the key dimensions of the challenge of Cloud Computing (**Understand – L2**)
- Classify the Levels of Virtualization and mechanism of tools. (**Understand – L2**)
- Analyze Cloud infrastructure including Google Cloud and Amazon Cloud. (**Analyze – L4**)
- Create Combinatorial Auctions for cloud resource and design scheduling algorithms for computing cloud. (**Apply – L3**)
- Assess control storage systems and cloud security, the risks involved its impact and develop cloud application (**Analyze – L4**)

**UNIT I:**

**Systems Modeling, Clustering and Virtualization:** Scalable Computing over the Internet-The Age of Internet Computing, Scalable computing over the internet, Technologies for Network Based Systems, System models for Distributed and Cloud Computing, Performance, Security and Energy Efficiency.

**UNIT II:**

**Virtual Machines and Virtualization of Clusters and Data Centers:** Implementation Levels of Virtualization, Virtualization Structures/ Tools and Mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation.

**UNIT III:**

**Cloud Platform Architecture:** Cloud Computing and Service Models, Public Cloud Platforms, Service Oriented Architecture, Programming on Amazon AWS, and Microsoft Azure

**UNIT IV:**

**Cloud Resource Management and Scheduling:** Policies and Mechanisms for Resource Management, Applications of Control Theory to Task Scheduling on a Cloud, Stability of a Two-Level Resource Allocation Architecture, Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds-Fair Queuing, Start Time Fair Queuing.

**UNIT V:**

**Storage Systems:** Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system.

**Text Books:**

1. Distributed and Cloud Computing, Kai Hwang, Geoffry C. Fox, Jack J. Dongarra MK Elsevier.
2. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.

**Reference Books:**

1. Cloud Computing, A Hands on approach, Arshadeep Bahga, Vijay Madiseti, University Press
2. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH
3. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammaraiselvi, TMH

B.Tech- (VII SEM)

20AD07 - DEEP LEARNING

L	T	P	Cr.
3	0	0	3

**Pre-requisite :** Probability and Statistics, LATT, Machine Learning

**Course Educational Objective:** The objective of the course is to make students learn the frameworks of deep learning and their application

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

**CO1:** Apply the fundamentals of linear algebra to machine learning algorithms. (**Apply- L3**)

**CO2:** Understand the fundamental building blocks of deep learning(**Understand- L2**)

**CO3:** Apply the concepts of Convolutional Neural Networks to computer vision applications. (**Apply- L3**)

**CO4:** Apply the concepts of Recurrent Neural Networks to Natural Language Processing. (**Apply- L3**)

**CO5:** Apply the regularization techniques to improve the model performance. (**Apply- L3**)

**UNIT-I: Mathematical foundations of Deep Learning**

Scalars, Vectors, Matrices and Tensors, Multiplying Matrices and Vectors, Identity and Inverse Matrices, Linear dependence and span, Norms, Special kinds of matrices and vectors, Trace operations, Eigen value decomposition

**UNIT-II: Fundamentals of Deep Learning**

Anatomy of Neural Networks: Layers, Models, Loss functions and optimizers  
 Training Deep Networks: Cost Functions, Optimizers  
 Types of Deep Neural Networks

**UNIT-III: Convolutional Neural Networks**

Motivation, Convolution Operation, Types of layers, Pooling, LENET5 Architecture

**UNIT-IV: Recurrent Neural Networks**

Architecture of traditional RNN, Types and applications of RNN, Variants of RNNs, Word Embedding using Word2vec

**UNIT-V: Regularization and Autoencoders**

Regularization for Deep Learning: L1 and L2, Dropout, Data Augmentation, Early Stopping, Case study on MNIST data  
 Autoencoders: Architecture, Implementation, Denoising Autoencoders, Sparse Autoencoders, Use cases

**TEXTBOOKS:**

1. Deep Learning, Ian Goodfellow, YoshuaBengio and Aaron Courville, MIT Press, 2016
2. Deep Learning with Python, Francois Chollet, Manning Publications, Released December 2017.
3. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence – Jon Krohn, Grant Beyleveld, AglaéBassens, Released September 2019, Publisher(s): Addison-Wesley Professional, ISBN: 9780135116821
4. Deep Learning from Scratch - Seth Weidman, Released September 2019, Publisher(s): O'Reilly Media, Inc., ISBN: 9781492041412

**Reference Books:**

1. Artificial Neural Networks, Yegnanarayana, B., PHI Learning Pvt. Ltd, 2009.
2. Matrix Computations, Golub, G.,H., and Van Loan,C.,F, JHU Press,2013.
3. Neural Networks: A Classroom Approach, Satish Kumar, Tata McGraw-HillEducation, 2004.

**Web Link:**

Swayam NPTEL: [https://onlinecourses.nptel.ac.in/noc22\\_cs22/preview](https://onlinecourses.nptel.ac.in/noc22_cs22/preview) B.Tech.(Artificial Intelligence and Data Science) Deep Learning:



B.Tech. (VII Sem.)

20CS25-SOFTWARE PROJECT MANAGEMENT

L	T	P	Cr.
3	0	0	3

Pre-requisite :Software Engineering, Software Testing Methodologies, Object oriented Analysis and Design.

**Course Educational Objective:** This course is centered on unique aspects of software project management at three levels: Organizational management, Infrastructure management and project management and measurement of the Project, and how these are applied to actual software projects.

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

- CO1:** Identify the process of ConventionalSoftwareManagement the Evolution and Improvement ofSoftwareEconomics. (**Remember-L1**)
- CO2:** Describe the basic s/w processes, Cost estimation and improvement in s/w Economics. (**Remember-L1**)
- CO3:** Summarize Life cycle phases and Artifactsoftheprocess in Software projectmanagement. (**Understand-L2**)
- CO4:** Apply Workflows and checkpoints in IterativeProcess planning.(**Apply-L3**)
- CO5:** Illustrate ProjectOrganizations, process automation building blocks and metrics in assessing Software Quality. (**Understand-L2**)

**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, The principles of conventional software Engineering, principles of modern software management, transitioning to an iterative process.

**UNIT-II**

**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, A Management perspective and technical perspective.

**UNIT-III**

**Work Flows of the process:** Software process workflows, Iteration workflows.

**Checkpoints of the process:** Major milestones, Minor Milestones, Periodic status assessments.

**Iterative Process Planning:** Work breakdown structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning.

**UNIT-IV**

**Project Organizations and Responsibilities:** Line-of-Business Organizations, Project Organizations, evolution of Organizations.

**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.

**UNIT-V**

**Tailoring the Process:** Process discriminates

**Future Software Project Management:** Modern Project Profiles, Next generation Software economics, modern process transitions.

**Case Study:** The command Center Processing and Display system Replacement (CCPDS)

**TEXTBOOK:**

1. Walker Royce, "Software Project Management", Pearson Education, 2015.

**REFERENCE BOOKS:**

1. Robert K. Wysocki, "Effective Software Project Management", Wiley Publication, 2011.
2. Walker Royce, "Software Project Management", Addison-Wesley, 1998.
3. Bob Hughes and Mike Cotterell, "Software Project Management", Tata McGraw-Hill Edition, 2000.

B.Tech. – (VII SEM)

20AD09- NATURAL LANGUAGE PROCESSING

L	T	P	Cr.
3	0	0	3

Pre-requisite : Nil

**Course Educational Objective:** The Objective of the course is to make learn the basic elements of C programming, control structures, derived data types, Modular programming, user defined structures, basics of files and its I/O operations.

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

- CO1:** Familiar with the basic components of NLP. (**Understand - L2**)
- CO2:** Applying N-gram models to predict a sequence of text. (**Apply - L3**)
- CO3:** Build a basic language understanding system using preliminary concepts of NLTK library. (**Apply - L3**)
- CO4:** Exposure on advanced techniques for understanding patterns in text (**Apply- L3**)
- CO5:** Understand the semantics of linguistic components in a natural dialogue (**Understand - L2**)

## UNIT – I

### Introduction

Knowledge in Speech and Language Processing; Ambiguity; Models and Algorithms; Language, Thought and Understanding;

### Regular Expressions

Regular Expression; Words; Corpora; Text Normalization; Minimum Edit Distance

## UNIT – II

### N-gram Language Models

N-Grams; Evaluating Language Models.Generalization and Zeros.

**Smoothing:** Laplace Smoothing; Add-k Smoothing; Backoff and Interpolation; Kneser-Ney Smoothing

## UNIT – III

### Natural language processing tools in Python (NLTK Package)

**Part-I:** Introduction to NLTK; Tokenizing; Filtering Stop words; Stemming; Tagging parts of speech; Lemmatizing; Chunking; Chinking

### Part-II:

Using Named Entity Recognition (NER); Getting Text to Analyze; Using a Concordance; Making a Dispersion Plot;

## UNIT – IV

**Information Extraction:** Relation Extraction Algorithms; Using Patterns to extract relations; Relation extraction via supervised learning; Semi supervised relation extraction via bootstrapping; Distant Supervision for Relation Extraction; Evaluation of Relation Extraction; Extracting Times; Extracting Events and their Times; Template Filling

## UNIT – V

Word Senses and WordNet

- Defining Word Senses; How many senses do words have?
- Relations between senses

WordNet: Sense relations in WordNet; Word Sense Disambiguation; Alternate WSD algorithms and Tasks

### TEXTBOOKS:

1. Daniel Jurafsky, James H. Martin ,”Speech and Language Processing” , Third Edition, PHI, 2020.
2. <https://realpython.com/nltk-nlp-python/#getting-text-to-analyze>

### REFERENCE BOOKS:

1. Natural Language Processing with Python: Analysing Text with the Natural Language Toolkit, Steven Bird, Ewan Klein, 2011
2. Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning, Benjamin Bengfort, Rebecca Bilbro, 2018
3. Speech and Language Processing, 2nd Edition, Daniel Jurafsky, James H. Martin, 2009.
- 4.

B.Tech. ( VII Sem.)

**20CS26-SOFT COMPUTING**

L	T	P	Cr.
3	0	0	3

Pre-requisite : Linear Algebra and Python.

**Course Educational Objective:** Introduce soft computing concepts to students and foster their abilities in designing and implementing soft computing-based solutions for real-world and engineering problems. The fundamental concepts of fuzzy systems, fuzzy logic, and its applications will be introduced to the students.

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

- CO1:** Describe the preliminaries of Artificial intelligence and Neural networks. **(Understand - L2)**
- CO2:** Map the issues to AI-based solutions. **(Apply - L3)**
- CO3:** Apply Soft computing techniques to solve real world problems. **(Apply – L3)**
- CO4:** Implement systems based on fuzzy logic. **(Apply – L3)**
- CO5:** Use Genetic algorithms to develop evolutionary approaches for solving real-world problems. **(Apply – L3)**

**UNIT -I**

**Introduction & Architecture:** Neuron, Nerve structure and synapse, Artificial Neuron and its model, activation functions, Neural network architecture: single layer and multi-layer feed forward networks, recurrent networks. Various learning techniques; perception and convergence rule, Auto-associative and hetero-associative memory.

**UNIT- II**

Perceptron model, solution, single layer artificial neural network, multi-layer perception model; back propagation learning methods, effect of learning rule co-efficient; back propagation algorithm, factors affecting back propagation training, applications.

**UNIT- III**

**Fuzzy Logic-I:** Basic concepts of fuzzy logic, Fuzzy sets and Crisp sets, Fuzzy set theory and operations, Properties of fuzzy sets, Fuzzy and Crisp relations, Fuzzy to Crisp conversion.

Fuzzy relations, rules, propositions, implications, and inferences. Defuzzification techniques. Some applications of Fuzzy logic.

**Unit -IV**

**Fuzzy Logic – II:** Membership functions, interference in fuzzy logic, fuzzy if-then rules, Fuzzy implications and Fuzzy algorithms, Fuzzifications & Defuzzification. Python Libraries for Fuzzy Logic: Fuzzywuzzy Python library, String Pattern Matching using Levenstein Algorithm

**Unit -V**

**Genetic Algorithms:** Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques, Basic GA framework and different GA architectures.

GA operators: Encoding, Crossover, Selection, Mutation, etc. Solving single-objective optimization problems using GAs.

**TEXTBOOKS**

1. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", John Wiley & Sons, 3<sup>rd</sup> Edition 2010.
2. S. Rajasekaran & G.A. Vijayalakshmi Pai, "Neural Networks, Fuzzy Logic & Genetic Algorithms, Synthesis & applications", PHI Publication, 1st Edition, 2009.
3. David E. Goldberg, "Genetic Algorithms", Pearson Education India, 2006.

**REFERENCE BOOKS**

1. Laurene Fauseett, "Fundamentals of Neural Networks", Prentice Hall India, New Delhi, 1994.
2. B. Yagna Narayana, "Artificial Neural Networks", PHI, 3<sup>rd</sup> Edition, 2009
3. Simon O. Haykin, "Neural Networks and Learning Machines", Prentice Hall, 3<sup>rd</sup> Edition, 2009.
4. <https://pypi.org/project/fuzzywuzzy/>

B.Tech. ( VII Sem.)

**20CS27-SOCIAL NETWORK ANALYSIS**

L	T	P	Cr.
3	0	0	3

Pre-requisite : Graph Theory, Computer Networks.

**Course Educational Objective:** The learning objective of the course «Social Network Analysis is to provide students with essential knowledge of network analysis applicable to real world data, with examples from today's most popular social networks.

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

- CO1:** Develop semantic web applications. (**Apply – L3**)
- CO2:** Represent social network data using the principles of Ontology. (**Apply - L3**)
- CO3:** Implement community mining algorithms to extract knowledge from Social Network Applications. (**Apply – L3**)
- CO4:** Predict human behavior in social web and related communities (**Apply – L3**)
- CO5:** Use visualization tools to represent the knowledge of social network data. (**Apply – L3**)

**UNIT - I: INTRODUCTION:**

Introduction to Semantic Web: Limitations of current Web - Development of Semantic Web - Emergence of the Social Web - Social Network analysis: Development of Social Network Analysis - Key concepts and measures in network analysis - Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities - Web-based networks - Applications of Social Network Analysis.

**UNIT II : MODELLING, AGGREGATING AND KNOWLEDGE REPRESENTATION:**

Ontology and their role in the Semantic Web: Ontology-based knowledge Representation - Ontology languages for the Semantic Web: Resource Description Framework - Web Ontology Language - Modelling and aggregating social network data: State-of-the-art in network data representation - Ontological representation of social individuals - Ontological representation of social relationships - Aggregating and reasoning with social network data - Advanced representations.

**UNIT - III: EXTRACTION AND MINING COMMUNITIES IN WEB SOCIAL NETWORKS:**

Extracting evolution of Web Community from a Series of Web Archive - Detecting communities in social networks - Definition of community - Evaluating communities - Methods for community detection and mining - Applications of community mining algorithms - Tools for detecting communities social network infrastructures and communities - Decentralized online social networks - Multi- Relational characterization of dynamic social network communities.

**UNIT - IV : PREDICTING HUMAN BEHAVIOUR AND PRIVACY ISSUES**

Understanding and predicting human behaviour for social communities - User data management - Inference and Distribution - Enabling new human experiences - Reality mining - Context - Awareness - Privacy in online social networks - Trust in online environment - Trust models based on subjective logic - Trust network analysis - Trust transitivity analysis - Combining trust and reputation - Trust derivation based on trust comparisons - Attack spectrum and countermeasures.

**UNIT - V :VISUALIZATION AND APPLICATIONS OF SOCIAL NETWORKS**

Graph theory - Centrality - Clustering - Node-Edge Diagrams - Matrix representation - Visualizing online social networks, Visualizing social networks with matrix-based representations - Matrix and Node-Link Diagrams - Hybrid representations - Applications - Cover networks - Community welfare - Collaboration networks - Co-Citation networks.

**TEXT BOOKS:**

1. Peter Mika, "Social Networks and the Semantic Web", 1<sup>st</sup> Edition, Springer 2007.
2. Borko Furht, "Handbook of Social Network Technologies and Applications", 3<sup>rd</sup> Edition, Springer, 2015.

**REFERENCE BOOKS:**

1. Guandong Xu, Yanchun Zhang and Lin Li, "Web Mining and Social Networking – Techniques and applications", First Edition Springer, 2011.



2. Dion Goh and Schubert Foo, “Social information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively”, IGI Global Snippet, 2008.
3. Max Chevalier, Christine Julien and Chantal Soulé-Dupuy, “Collaborative and Social Information Retrieval and Access: Techniques for Improved user Modelling”, IGI Global Snippet, 2009.
4. John G. Breslin, Alexander Passant and Stefan Decker, “The Social Semantic Web”, Springer, 2009.

B.Tech. ( VII Sem.)

20CS28-COMPUTER VISION

L	T	P	Cr.
3	0	0	3

Pre-requisite : Linear Algebra, Probability and Statistics, Computer Graphics.

**Course Educational Objective:** 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; and to provide the student with programming experience from implementing computer vision and object recognition applications.

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

- CO1:** Understand image processing techniques required for computer vision. (**Understand – L2**)
- CO2:** Describe image formation models. (**Understand – L2**)
- CO3:** Understand feature extraction and motion estimation techniques. (**Understand- L2**)
- CO4:** Apply segmentation algorithms for shape analysis. (**Apply – L3**)
- CO5:** Implement image classification Applications using object recognition methods. (**Apply – L3**)

**UNIT-I**

**Introduction:** Image Processing, Computer Vision and Computer Graphics , What is Computer Vision -Low-level, Mid-level, High-level , Overview of Diverse Computer Vision Applications: Document Image Analysis, Biometrics, Object Recognition, Tracking, Medical Image Analysis, Content-Based Image Retrieval, Video Data Processing, Multimedia, Virtual Reality and Augmented Reality.

**UNIT-II**

**Image Formation Models:** Monocular imaging system , Radiosity: The ‘Physics’ of Image Formation, Radiance, Irradiance, BRDF, color etc, Orthographic & Perspective Projection,• Camera model and Camera calibration, Binocular imaging systems, Multiple views geometry, Structure determination, shape from shading, Photometric Stereo, Depth from Defocus , Construction of 3D model from images.

**UNIT-III**

**Image Processing and Feature Extraction:** Image preprocessing, Image representations (continuous and discrete) , Edge detection.

**Motion Estimation:** Regularization theory, Optical computation, Stereo Vision, Motion estimation, Structure from motion.

**UNIT-IV**

**Shape Representation and Segmentation:**Contour based representation, Region based representation, Deformable curves and surfaces, Snakes and active contours, Level set representations, Fourier and wavelet descriptors, Medial representations, and Multiresolution analysis.

**UNIT-V**

**Object recognition:** Hough transforms and other simple object recognition methods, Shape correspondence and shape matching, Principal Component analysis, Shape priors for recognition.

**TEXT BOOKS:**

1. D. Forsyth and J. Ponce , “Computer Vision -A modern approach,”Pearson, 2<sup>nd</sup> Edition ,2015.
2. E. Trucco and A. Verri, “Introductory Techniques for 3D Computer Vision”, Prentice Hall,3<sup>rd</sup> Edition, 2016

**REFERENCE BOOKS:**

1. R. C. Gonzalez, R. E. Woods. “Digital Image Processing”, Addison Wesley Longman, Inc., 2<sup>nd</sup> Edition 1992.
2. Richard Szeliski, “Computer Vision: Algorithms and Applications (CVAA)”, Springer, 2<sup>nd</sup> Edition,2010.
3. Sonka,Hlavac, and Boyle. “Image Processing, Analysis, and Machine Vision”. Thomson, 4<sup>th</sup> Edition,2011.
4. E. R. Davies, “Computer and Machine Vision”, Academic Press,4<sup>th</sup>Edition,2012.

5. Simon J. D. Prince, “Computer Vision: Models, Learning, and Inference”, Cambridge University Press, 2012.
6. Mark Nixon and Alberto S.Aquado, “Feature Extraction& Image Processing for Computer Vision” , Academic Press, 3<sup>rd</sup> Edition,2012.

B.Tech. (VII Sem.)

20CS29-BLOCK CHAIN TECHNOLOGY

L	T	P	Cr.
3	0	0	3

**Pre-requisite:** Information Security

**Course Educational Objective:** To understand block chain technology and Crypto currency works

**Course Outcomes:** After the completion of the course, student will be able to

**CO1:** Demonstrate the block chain basics, Crypto currency (**Understand-L2**)

**CO2:** Compare and contrast the use of different private vs. public block chain and use cases (**Understand-L2**)

**CO3:** Design an innovative Bit Coin, Block chain and scripts, Block chain Science on varies coins (**Apply-L3**)

**CO4:** Classify Permission Block chain and use cases – Hyper ledger, Corda (**Analyze-L4**)

**CO5:** Use of Block-chain in E-Governance, Land Registration, Medical Information Systems and others (**Apply-L3**)

### **UNIT I:**

**Introduction:** Introduction, basic ideas behind block chain, how it is changing the Landscape of digitalization, introduction to cryptographic concepts required, Block chain or distributed trust, Currency, Cryptocurrency, how a Cryptocurrency works, Financial services, Bitcoin prediction markets.

### **UNIT II:**

Hashing, public key cryptosystems, private vs public block chain and use cases, Hash Puzzles, Extensibility of Block chain concepts, Digital Identity verification, Block chain Neutrality, Digital art, Block chain Environment

### **UNIT III:**

**Introduction to Bitcoin:** Bitcoin Block chain and scripts, Use cases of Bitcoin Block chain scripting language in micropayment, escrow etc Downside of Bit coin mining, Block chain Science: Grid coin, folding coin, Block chain Genomics, Bit coin MOOCs.

### **UNIT IV:**

Ethereum continued, IOTA, The real need for mining, consensus, Byzantine Generals Problem, and Consensus as a distributed coordination problem, Coming to private or permissioned block chains, Introduction to Hyper ledger, Currency, Token, Campus coin, Coin drop as a strategy for Public adoption, Currency Multiplicity, Demurrage currency

### **UNIT V:**

Technical challenges, Business model challenges, Scandals and Public perception, Government Regulations, Uses of Block chain in E-Governance, Land Registration, Medical Information systems.

### **Text Books:**

1. Block chain Blue print for Economy by Melanie Swan

### **Reference Books:**

1. Block chain Basics: A Non-Technical Introduction in 25 Steps 1st Edition, by Daniel Drescher

L	T	P	Cr.
3	0	0	3

**Course Description:**

In this course, students will learn fundamental concepts and contributions of management. This course also teaches human resources practices which play a vital role in the organisation it gives knowledge about use of improve quality of work and project management.

**Course Objectives:**

1. To make students understand management, its principles, contribution to management, organization, and its basic issues and types
2. To make students understand the concept of plant location and its factors and plant layout and types, method of production and work study importance
3. To understand the purpose and function of statistical quality control. And understand the material management techniques
4. To make students understand the concept of HRM and its functions
5. To make students understand PERT & CPM methods in effective project management and need of project crashing and its consequence on cost of project

**Course Outcomes:**

**CO1:** Understand management principles to practical situations based on the organization structures. (L2)

**CO2:** Design Effective plant Layouts by using work study methods. (L2)

**CO3:** Apply quality control techniques for improvement of quality and materials management. (L3)

**CO4:** Develop best practices of HRM in corporate Business to raise employee productivity. (L2)

**CO5:** Identify critical path and project completion time by using CPM and PERT techniques. (L3)

**UNIT I**

**Introduction:** Management - Definition, Nature, Importance of management Functions of Management - Taylor's scientific management theory, Fayal's principles of management, Contribution of Elton mayo, Maslow, Herzberg, Douglas MC Gregor, Basic Concepts Of Organisation- Authority, Responsibility Delegation of Authority, Span of control, Departmentation and Decentralization - Organisation structures (Line organization, Line and staff organization, Functional organization, Committee organization, Matrix organization)

**UNIT II**

**Operations Management:** Plant location, Factors influencing location, Principles and types of plant layouts - Methods of production (job, batch and mass production), Work study - Basic procedure involved in method study and Work measurement

**UNIT III**

**Statistical quality control** – Concept of Quality & Quality Control-functions, Meaning of SQC - Variables and attributes - X chart, R Chart, C Chart, P Chart, (simple Problems) Acceptance sampling, Sampling plans, Deming's contribution to quality.

**Materials management** – Meaning and objectives, inventory control-Need for inventory control, Purchase procedure, Store records, EOQ, ABC analysis, Stock levels

#### UNIT IV

**Human Resource management (HRM):** Concepts of HRM, Basic functions of HR manager: Man power planning, Recruitment, Selection, Training and development, Placement, Wage and salary administration, Promotion, Transfers Separation, performance appraisal, Job evaluation and merit rating.

#### UNIT V

**Project management:** Early techniques in project management - Network analysis: Programme evaluation and review technique (PERT), Critical path method (CPM), Identifying critical path, Probability of completing project within given time, Project cost analysis, project crashing (simple problems)

#### Text Books:

Dr. A.R.Aryasri, Management Science, TMH, 10<sup>th</sup> edition, 2012

#### References:

1. Koontz & wehrich – Essentials of management, TMH, 10<sup>th</sup> edition, 2015
2. Stoner, Freeman, Gilbert, Management, 6<sup>th</sup> edition Pearson education, New Delhi, 2004
3. O.P. Khana, Industrial engineering and Management L.S.Srinath, PERT & CPM



B.Tech. (VII Sem.)

**20CSS4– Continuous Integration and Delivery using DevOps**

L	T	P	Cr.
1	0	2	2

**Pre-requisite** : Basic knowledge of certain basic programming languages such as Java, and Python

**Course Educational Objective:** This course is designed to provide the core education necessary to build your DevOps vocabulary and to understand its principles and practices. With the help of key DevOps concepts and terminology, real-life case studies, examples and interactive group discussions and exercises, you will acquire a fundamental understanding of DevOps.

**Course Outcomes (CO):** *At the end of this course, the student will be able to:*

**CO1:** Understand the why, what, and how of DevOps adoption(**Understand – L2**)

**CO2:** Attain literacy on DevOps(**Apply L3**)

**CO3:** Align capabilities required in the team and create an automated CICD pipeline using a stack of tools(**Apply L3**)

**CO 4:** Improve individual / teamwork skills, communication & report writing skills with ethical values.

**Module 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

**Module 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

**Module 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

**Module 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.
--

**Module 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.
--

**Module 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.
---

**Module 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.
--

**Module 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.
---

**Module 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.
---

**Module 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.
--

**Module 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.
--

**Module 12:**

Module name: Course end assessment
------------------------------------

In the configured Jenkins pipeline created in Exercise 8 and 9, implement quality gates for code coverage.
--

**Reference 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>]

**Hardware and Software Configuration:**

1. Git [GitHub or Gitlab]
2. Sonarqube
3. Jenkins
4. JUnit
5. Eclipse
6. Tomcat server
7. Maven
8. Cobertura or JaCoCo
9. Java SDK
10. All necessary drivers and jar files for connecting the software
11. Windows machine with 16GB RAM

**Web Links:** (Courses mapped to Infosys Springboard platform)

1. [https://infyspringboard.onwingspan.com/en/app/toc/lex\\_auth\\_013382690411003904735\\_share/overview](https://infyspringboard.onwingspan.com/en/app/toc/lex_auth_013382690411003904735_share/overview) [Software Engineering and Agile software development]
2. [https://infyspringboard.onwingspan.com/en/viewer/html/lex\\_auth\\_01350157819497676810467](https://infyspringboard.onwingspan.com/en/viewer/html/lex_auth_01350157819497676810467) [Development & Testing with Agile: Extreme Programming]
3. [https://infyspringboard.onwingspan.com/en/viewer/html/lex\\_auth\\_01353898917192499226\\_shared](https://infyspringboard.onwingspan.com/en/viewer/html/lex_auth_01353898917192499226_shared) [DevOps CICD]