DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

PACE R-21 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

I Year - I Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21HST01	Communicative English	3	0	0	3
2	P21BST01	Linear Algebra & Differential Equations	3	0	0	3
3	P21BST02	Applied Physics	3	0	0	3
4	P21EST04	Computer Engineering Workshop	3	0	0	3
5	P21EST03	C-Programming for Problem Solving	3	0	0	3
6	P21HSL01	English Language Communication Skills Lab	0	0	3	1.5
7	P21BSL01	Applied Physics Lab	0	0	3	1.5
8	P21ESL02	C-Programming for Problem Solving Lab	0	0	3	1.5
9	P21MCT01	Induction program	2	0	0	0
Total Credits						19.5

I Year - II Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21BST04	Applied Chemistry	3	0	0	3
2	P21BST06	Numerical Methods & Vector calculus	3	0	0	3
3	P21EST12	Digital Electronics	3	0	0	3
4	P21EST13	Data Structures	3	0	0	3
5	P21EST14	Python Programming	3	0	0	3
6	P21BSL03	Applied Chemistry Lab	0	0	3	1.5
7	P21ESL06	Data Structures Lab	0	0	3	1.5
8	P21ESL07	Python Programming Lab	0	0	3	1.5
Total Credits						19.5

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
PACE R-21 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

II Year - I Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21BST07	Probability and Statistics	3	0	0	3
2	P21CIT01	Object Oriented Programming Through C++	3	0	0	3
3	P21CST02	Front end Web Technologies	3	0	0	3
4	P21CST03	Computer Organization	3	0	0	3
5	P21CST04	Software Engineering	3	0	0	3
6	P21CIL01	Object Oriented Programming Through C++ Lab	0	0	3	1.5
7	P21CSL02	Front End Web Technologies Lab	0	0	3	1.5
8	P21CSL03	Free Open Source Software Lab	0	0	3	1.5
9	P21CIS01	Skill Oriented Course-I	1	0	2	2
10	P21MCT03	Environmental Science	2	0	0	0
		Total Credits				21.5

II Year - II Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21CST05	Mathematical Foundations of Computer Science	3	0	0	3
2	P21CST08	Design and Analysis of Algorithms	3	0	0	3
3	P21CST01	Java Programming	3	0	0	3
4	P21CST07	Database Management Systems	3	0	0	3
5	P21SST01	Developing Soft Skills and Personality	3	0	0	3
6	P21CSL06	R Programming Lab	0	0	3	1.5
7	P21CSL01	Java Programming Lab	0	0	3	1.5
8	P21CSL05	Database Management Systems Lab	0	0	3	1.5
9	P21CIS02	Skill Oriented Course-II	1	0	2	2
					Total Credits	21.5
Internship 2 Months (Mandatory) during summer vacation						

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
R-21 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

III Year - I Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21CST09	Computer Networks	3	0	0	3
2	P21CIT02	Web Technologies	3	0	0	3
3	P21CST10	Operating Systems	3	0	0	3
4	P21CIXXX	Professional Elective-I	3	0	0	3
5	P21XXXXX	Open Elective-I	3	0	0	3
6	P21CSL07	Computer Networks & Operating systems Lab	0	0	3	1.5
7	P21CIL02	Web Technologies Lab	0	0	3	1.5
8	P21CIS03	Skill Oriented Course-III	1	0	2	2
9	P21XXXX	Professional Ethics & Human Values	2	0	0	0
10	P21XXXX	Summer Internship 2 Months (Mandatory) after II Year (to be evaluated during III Year I Semester)	0	0	0	1.5
Total Credits						21.5

Professional Elective - I		
S.No	**Course Code**	**Course Title**
1	MOOC-I	NPTEL

III Year - II Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21CST12	Cryptography & Network Security	3	0	0	3
2	P21AMT01	Machine Learning	3	0	0	3
3	P21CST14	Automata Theory & Compiler Design	3	0	0	3
4	P21CIXXX	Professional Elective-II	3	0	0	3
5	P21XXXXX	Open Elective-II	3	0	0	3
6	P21CIL03	Network Programming & Security Lab	0	0	3	1.5
7	P21AML01	Machine Learning Lab	0	0	3	1.5
8	P21CSL11	Unified Modeling Languages Lab	0	0	3	1.5
9	P21CIS04	Skill Oriented Course-IV	1	0	2	2
10	P21XXXXX	Design Thinking for Innovation	2	0	0	0
					Total Credits	21.5
Industrial/Research Internship (Mandatory) 2 Months during summer vacation						

Professional Elective - II		
S.No	**Course Code**	**Course Title**
1	MOOC- II	NPTEL

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
R-21 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

IV Year - I Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21CIXXX	Professional Elective-III	3	0	0	3
2	P21CIXXX	Professional Elective-IV	3	0	0	3
3	P21CIXXX	Professional Elective-V	3	0	0	3
4	P21XXXXX	Open Elective-III	3	0	0	3
5	P21XXXXX	Open Elective-IV	3	0	0	3
6	P21XXXXX	Universal Human Values-II	3	0	0	3
7	P21CIXXX	Skill Advanced Oriented/Soft Skill Course	1	0	2	2
8	P21CIXXX	Industrial/Research Internship 2 Months (Mandatory) after III Year (to be evaluated during IV Year I Semester)	0	0	0	3
					Total Credits	23

Professional Elective - III		
S.No	**Course Code**	**Course Title**
1	P21CIXXX	Block chain Technology
2	P21CIXXX	Deep Learning
3	P21CIXXX	Cloud Computing
3	P21CIXXX	Service Oriented Architecture

Professional Elective - IV		
S.No	**Course Code**	**Course Title**
1	P21CIXXX	Wireless Networks And Mobile Computing
2	P21CIXXX	Medical Image Data Processing
3	P21CIXXX	Cyber Security
3	P21CIXXX	Full Stack Development

Professional Elective - V		
S.No	**Course Code**	**Course Title**
1	P21CIXXX	Advanced Java Programming
2	P21CIXXX	Data Visualization Techniques
3	P21CIXXX	Computer Vision
3	P21CIXXX	Natural Language Processing

DEPARTMENT OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
R-21 UNDER GRADUATE (B.Tech) COURSE STRUCTURE

IV Year - II Semester						
S.No	**Course Code**	**Course Title**	**L**	**T**	**P**	**C**
1	P21CIXXX	Project Work, Seminar and Internship in Industry	0	0	0	12
Internship (6 Months)						
					Total Credits	12

Course Code	Course Name	Course Structure			
		L	T	P	C
P21HST01	Communicative English	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: The students should have basic knowledge of English grammar and LSRW skills.

Course Objectives: The student will be able

1. To enable the engineering students to develop their basic communication skills in English for academic and social purposes.
2. To equip the students with appropriate oral and written communication skills.
3. To inculcate the skills of listening, reading and critical thinking.
4. To integrate English Language learning with employability skills and training.
5. To enhance the students' proficiency in reading skills enabling them to meet the academic demands of their course.

Course Outcomes: After going through this course the student will be able to

1. Use English Language effectively in spoken and written forms
2. Interpret the contextual meaning of words
3. Comprehend the given texts and respond appropriately
4. Recall and reproduce the theme in a given context
5. Communicate confidently in formal and informal contexts

UNIT–I **(9 Lectures)**

a. Reading Skills: Leela's Friend – R.K. Narayan
b. Vocabulary: Synonyms, Antonyms and Word formation, Root Words
c. Grammar: Parts of Speech, Sentence structure and Types of sentences
d. Writing: Letter Writing, Note Making and Note Taking

UNIT–II **(10 Lectures)**

a. Reading Skills: Dr. A.P.J. Abdul Kalam's Biography
b. Vocabulary: Prefixes, Suffixes and Affixes
c. Grammar: Prepositions and Articles
d. Writing: Paragraph Writing and Precis Writing

UNIT–III **(9 Lectures)**

a. Reading Skills: Three Days to See – Helen Keller
b. Vocabulary: Collocations, One word substitutes & Idioms
c. Grammar: Tenses, Active voice & Passive voice
d. Writing: Technical Report Writing

UNIT–IV **(9 Lectures)**

a. Reading Skills: Satya Nadella's Email to His Employees on His First Day as CEO of Microsoft
b. Vocabulary: Phrasal verbs and Commonly confused words
c. Grammar: Subject–Verb Agreement (Concord) and Question tags
d. Writing: Curriculum vitae, Cover Letter and Resume Writing. (Functional, Chronological and standard Resumes)

UNIT–V **(9 Lectures)**

a. Reading Skills: Mokshagundam Visveswaraya
b. Vocabulary: Homonyms, Homophones and Homographs
c. Grammar: Modal Auxiliaries, Degrees of Comparison and Direct speech & Indirect Speech
d. Writing: E- mail Writing and Essay Writing

Text Books:

1. New Horizons – Pearson Publishers
2. Fluency in English", A Course Book for Engg. Students, Published by Orient Black Swan, Hyderabad, 2016 print.
3. "Technical Communication- Principles and Practice", Third Edition. New Delhi: Oxford University press.
4. Epitome of Wisdom – Maruthi Publications

Reference Books:

1. Meenakshi raman, Sangeetha, Sharma Fundamentals of technical communication, Pg: 119-153 Oxford University press, 2015
2. Rutherford, Andhrea. J, Communication skills for technology. Pearson, New Delhi.2001
3. Raymond Murphy, Murphy's English Grammar, Cambridge University Press 2004
4. Meenakshi raman, Sangeetha, Sharma,Technical communication: English Skills for Engineers, Oxford University press, 2009
5. Michael Swan, Practical English Usage, Oxford University press, 1996

Web Resources:

1. `www.englishhints.com`
2. `www.enchantedlearning.com`
3. `www.learnenglish.de/grammar/prefixtext.html`

4. http://www.magickeys.com/books/riddles/words.html
5. http://www.pinnacle.edu.in/campusfiles/1826_campusFile_1.pdf
6. http://www.yourdictionary.com
7. http://www.learnenglish.com
8. http://www.cambridge.org
9. http://www.eslcafe.com
10. http://www.eslgames.com
11. http://www.penguin.co.uk
12. http://www.edufind.com/english/practice
13. www.englishhints.com, www.enchantedlearning.com,
14. www.learnenglish.de/grammar/prefixtext.html
15. http://www.magickeys.com/books/riddles/words.html

Course Code	Course Name	Course Structure			
P21BST01	Linear Algebra & Differential Equations	L	T	P	C
		3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Basics of Matrix Algebra, Differentiation, Integration
Course Objectives: The student will be able to

1. The concept of rank of a matrix which is used to know the consistency of system of linear equations and find the solution by using various analytical and numerical methods.
2. Eigen values and eigenvectors of a given matrix. Cayley-Hamilton theorem to find the inverse and power of a matrix and determine the nature of the quadratic form,
3. Recognize and model differential equations, apply analytical techniques to compute solutions for engineering problems.
4. The general solution to the higher order linear differential equations and applies to calculate the current in electrical circuits.
5. Explore the use of Laplace transform method to solve with initial value problems of ordinary differential equations.

Course Outcomes: After going through this course the student will be able to

1. Demonstrate the understanding of rank of a matrix. Analyze the solution of the system of linear equations.
2. Find the Eigen values and Eigenvectors of a matrix, apply Cayley-Hamilton theorem to determine inverse and power of a matrix and identify the nature of the quadratic form.
3. Solve the differential equations of first order and first degree related to various engineering fields.
4. Find the complete solution to the higher order linear differential equations and apply these methods to find the current in complex electrical circuits.
5. Apply the technique of Laplace transform and solve differential equations for analytical solutions with the initial conditions.

UNIT–I: Solving System of Linear Equations (8 Lectures)

Rank of a matrix by Echelon form-Normal form- Normal form through PAQ method – Solving system of homogeneous and non-homogeneous linear equations – Gauss elimination – Gauss Jordon methods.

UNIT–II: Eigen values – Eigenvectors, Cayley-Hamilton Theorem and Quadratic forms (10 Lectures)

Eigen values - Eigenvectors– Properties – Cayley-Hamilton theorem (without proof) - Finding inverse and power of a matrix by Cayley-Hamilton theorem–Reduction to Diagonal form. Quadratic forms: Rank, index, signature and nature of the quadratic forms–Reduction of quadratic form to canonical forms by orthogonal transformation.

UNIT–III: Differential Equations of First Order and First Degree (10 Lectures)

Linear differential equation - Bernoulli's differential equation–Exact equations and equations reducible to exact form.

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

UNIT–IV: Linear Differential Equations of Higher order (8 Lectures)

Non-homogeneous equations of higher order with constant coefficients-with RHS term of the type e^{ax}, $sinax$, $cosax$, polynomials in x, $e^{ax}V(x)$, $x^nV(x)$ and general method - Method of Variation of parameters.

Applications: LCR circuit

UNIT–V: Laplace Transforms (9 Lectures)

Laplace transforms of standard functions– First shifting Theorem-Change of scale property multiplication by t^n–division by t, transforms of derivatives and integrals – Second shifting theorem– Laplace transform of Periodic functions.

Inverse Laplace transforms – Convolution theorem (without proof)

Applications: Evaluation of integrals using Laplace transforms - Solving ordinary differential equations (Initial value problems) using Laplace transforms.

Text Books:

1. B. S. Grewal, Higher Engineering Mathematics, 44^{th} Edition, Khanna Publishers
2. B.V. Ramana, Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.

Reference Books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10^{th} Edition, Wiley-India.
2. H. K. Das, Advanced Engineering Mathematics, 22^{nd} Edition, S. Chand & Company Ltd.
3. David Poole, Linear Algebra- A modern introduction, 4^{th} edition, Cengage.
4. Peter O' Neil, Advanced Engineering Mathematics, Cengage
5. Srimantha Pal, S C Bhunia, Engineering Mathematics, Oxford University Press.

Web Resources:

1. `http://tutorial.math.lamar.edu/Classes/DE/DE.aspx`
2. `http://mathworld.wolfram.com/topics`
3. `http://www.nptel.ac.in/course.php`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21BST02	Applied Physics	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: The basics of analytical and conceptual understanding of physics

Course Objectives: The student will be able

1. To study the wave nature of light through Interference and diffraction.
2. To learn the basic principles of Lasers and fiber optics.
3. To express the physics of electrostatics and electromagnetic wave concepts through Maxwell's equations.
4. To study the basic concepts of Quantum mechanics.
5. Aware of limits of classical free electron theory and apply band theory of solids.
6. Acquire the knowledge of semiconductor physics.

Course Outcomes: After going through this course the student will be able to

1. Understanding the basic concepts of optics and how to apply them for engineering applications.
2. Acquire the knowledge of fundamentals of Lasers and fiber optics enables the students to develop Laser devices to apply them in various systems like communications, Industries and medicine.
3. Set students to be exposed to Electrostatics, Maxwell's equations, electromagnetic waves and fundamental concepts of quantum mechanics.
4. Enable to learn the fundamental concepts of free electron theory and band theory of solids.
5. Develop knowledge of band theory of solids for fundamentals of Semiconductor physics enables the students to apply the knowledge to various systems like communications, solar cell, photo cells and so on.

UNIT–I: Wave Optics **(9 Lectures)**

Interference: Introduction, Principle of Superposition of waves, colors in thin films, interference in thin films, Newton's rings: Determination of wavelength and refractive index.

Diffraction: Introduction, differences between interference and diffraction, difference between Fraunhofer and Fresnel's diffraction, Fraunhofer diffraction at single slit, Fraunhofer diffraction due to double slit, Diffraction grating (N-slits qualitative), resolving power of grating.

UNIT–II: Lasers and Fiber Optics **(9 Lectures)**

Lasers: Introduction, Characteristics of laser, absorption, spontaneous emission, stimulated emission, Einstein's coefficients, population inversion, pumping, pumping mechanisms, Types of Lasers: Ruby laser, He-Ne laser, diode laser, Applications of Lasers.

Fiber optics: Introduction, Total internal reflection-wave propagation in optical fiber, Acceptance angle, numerical aperture, applications of optical fiber.

UNIT–III: Electrostatics, Maxwell's Equations and Electromagnetic Waves (9 Lectures)

Electrostatics: Coulombs law, electric field, electric field intensity, electric flux density, electrostatic potential, divergence of electric field, Laplace's and Poisson's equations for electrostatic potential, Gauss theorem in electrostatics.

Maxwell's equations and electromagnetic waves: Gauss theorem in magneto statics, Faraday's law of electromagnetic induction, Ampere's law, displacement current, Maxwell's equations in vacuum, electromagnetic wave equation in dielectric medium, velocity of propagation of electromagnetic wave, poynting vector and poynting theorem.

UNIT–IV: Quantum Mechanics, Free Electron Theory and Band Theory (10 Lectures)

Quantum Mechanics: Introduction to quantum physics, de-Broglie's hypothesis and properties of matter waves, Schrodinger's time independent wave equation, Schrodinger's time dependent wave equation, Particle in one dimensional box.

Free electron theory: classical free electron theory of metals- assumptions and failures, quantum free electron theory of metals-assumptions and failures, Fermi Dirac distribution function- Fermi level, Femi energy, density of states.

Band theory of solids: Introduction, Bloch's theorem, Kronig penny model (qualitative), E-K diagram, Brillouin's zones, classification of solids into metals, semiconductors and insulators, effective mass of electron and concept of hole.

UNIT–V: Semiconductor Physics (8 Lectures)

Semiconductor physics: Introduction, intrinsic and extrinsic semiconductors, carrier concentration in intrinsic semiconductors, electrical conductivity of intrinsic semiconductor, Fermi energy, carrier concentration in N-type and P-type semiconductors, dependence of Fermi energy on carrier-concentration and temperature, drift and diffusion, Hall effect and its applications, mechanism in LED, solar cell and photo conductor.

Text Books:

1. A Textbook of Engineering Physics by Dr. M. N. Avadhanulu, Dr. P.G. Kshirsagar - S. Chand.
2. Optics by Ajoy Ghatak, Tata McGraw-Hill Publishing company limited
3. Introduction to Electrodynamics by David Griffiths, Cambridge University Press
4. Introduction to Quantum physics by Eisberg and Resnick.

Reference Books:

1. Applied physics by Palanisamy (Scitech publications)
2. Optics by Eugene Hecht, Pearson Education.
3. Principle of Lasers by O. Svelto
4. Electricity, magnetism and light by W. Saslow
5. Semiconductor Optoelectronics by J. Singh, Physics and Technology, Mc Graw-Hill inc
6. Engineering Physics by B.K. Pandey, S. Chaturvedi - Cengage Learing.

Web Resources:

1. `https://nptel.ac.in/courses/115/106/115106066/`
2. `https://ocw.mit.edu/high-school/physics/exam-prep/electromagnetism/maxwells-equations/#1`
3. `https://ocw.mit.edu/courses/materials-science-and-engineering/3-091sc-introduction-to-solid-state-chemistry-fall-2010/electronic-materials/14-semiconductors/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21EST03	C - Programming for Problem Solving	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Nill

Course Objectives: The student will be able

1. To impart adequate knowledge on the need of programming languages and problem solving techniques.
2. To impart problem solving skills.
3. To enable student to write programs in C and to solve the problems.

Course Outcomes: After going through this course the student will be able to

1. Design algorithms and flowchart / Pseudo code for a given problem.
2. Design programs involving decision structures and loops.
3. Implement different operations on arrays and solve problems using functions.
4. Understand pointers and strings.
5. Implement structures, unions and file operations in C programming for a given application problem.

UNIT–I **(8 Lectures)**

Introduction to Programming: Computer hardware, Bits and Bytes, programming languages, application and system software, the software development process.

Idea of algorithm: steps to solve logical and numerical problems. Representation of algorithm: flowchart/pseudo code with examples, from algorithms to programs.

UNIT–II **(9 Lectures)**

Introduction to C: Overview of C, Constants, Variables and Data Types, Operators and Expressions, Managing Input and Output. Decision Making - Branching and Looping. Enumerated Data type, Renaming Data type with type def, Type Casting.

UNIT–III **(10 Lectures)**

Arrays: Definition, Declaration, Initialization, Assignment, Processing array, Passing array to a function, Two and multi dimensional array. **Functions:** Defining a function, Accessing a function, Passing argument to functions, Function prototypes, Nested function call, Storage classes.

UNIT–IV **(9 Lectures)**

Pointers: Definition, initialization, operations on pointers, functions and pointers, arrays and pointers, pointers to pointers, dynamic memory allocation.

Strings: C Strings, String Input / Output functions, arrays of strings, string manipulation functions.

UNIT–V **(9 Lectures)**

Structures: Definition, declaration, initialization, accessing members, array of structures, arrays within structure, functions and structures, pointers to structures, nested structures, unions.

File Handling: Types, operations on files, modes, file I/O functions, Random Access Functions.

Text Books:

1. Byron S Gottfried, —Programming with C, Schaums Outlines, Second Edition, Tata McGraw-Hill, 2006.
2. Problem Solving and Program Design in C, 4th edition, by jeri R. Hanly and Elli B.Koffman.
3. Balagurusamy. 2011. C Programming. Tata Mc Graw Hills, New Delhi, India.

Reference Books:

1. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Prentice Hall of India.
2. Yashavant P. Kanetkar. —Let Us C, BPB Publications, 2011.

Web Resources:

1. `https://www.studytonight.com/c/`
2. `https://www.cprogramming.com/tutorial/c-tutorial.html`
3. `https://www.javatpoint.com/c-programming-language-tutorial`
4. `https://www.tutorialspoint.com/cprogramming/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21EST04	Computer Engineering Workshop	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NILL

Course Objectives:

1. To make the students aware of the basic hardware components of a computer and installation of operating system.
2. To introduce Raptor Tool for flowchart creation.
3. To introduce programming through Visual Programming tool using scratch.
4. To get knowledge in awareness of cyber hygiene that is protecting the personal computer from getting infected with the viruses, worms and other cyber-attacks.
5. To introduce the usage of Productivity tools in crafting professional word documents, excel spreadsheets and power point presentations using open office tools.

UNIT–I:

Simple Computer System: Central processing unit, the further need of secondary storage, Types of memory, Hardware, Software and people. Peripheral Devices: Input, Output and storage, Data Preparation, Factors affecting input, Input devices, Output devices, Secondary devices, Communication between the CPU and Input/ Output devices.

TASK 1: PC Hardware: PC Hardware introduces the students to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers. In addition hardware and software level troubleshooting process, tips and tricks would be covered. Every student should 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. Every student should disassemble and assemble the PC back to working condition.

TASK 2: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition.

Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition.

UNIT–II:

Problem Solving and Programming: Algorithm development, Flowcharts, Looping, some programming features, Pseudo code, the one-zero game, some structured programming concepts, documents. Programming Languages: Machine Language and assembly language, high –level and low level languages, Assemblers, Compilers, and Interpreters.

TASK 3: Drawing flowcharts (Raptor Tool)

a) Create flowcharts for take-off landing of an Aeroplane.

b) Create a flowchart to validate an email id entered by user.

c) Create flowchart to print first 50 prime numbers.

TASK 4: Productivity tool:LaTeX and Microsoft (MS) office: Importance of MS office, Details of the three tasks and features that should be covered in each, MS word, Power Point, Excel.

UNIT–III:

Operating systems: Introduction, Evolution of operating systems, , Command Interpreter, Popular operating systems- Microsoft DOS, Microsoft Windows, UNIX and Linux. Introduction to Unix Shell Commands, directory management commands, file operations, users commands, Time and Date commands.

TASK 5: Operating System Installation: Every student should individually install operating system like Linux or MS windows on the personal computer. The system should be configured as dual boot with both windows and Linux.

TASK 6: Basic Commands: Unix Shell Commands, directory management commands, file operations, users commands, Time and Date commands.

UNIT–IV:

Computer Networks: Introduction to computer Networks, Network topologies-Bus topology, star topology, Ring topology, Mesh topology, Hybrid topology, Types of Networks: Local area Network, Wide Area Networks, Metropolitan Networks, Campus/ Corporate Area Network, Personal Area Network, Network Devices- Hub, Repeater, Switch, Bridge, Router, Gateway, Network interface Card, Basic Networking Commands.

TASK 7: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IPsetting. Finally students should demonstrate how to access the websites and email.

TASK 8: Networking Commands: ping, ssh, ifconfig, scp, netstat, ipstat, nslookup, traceroute, telnet, host, ftp, arp, wget,route

UNIT–V:

Introduction to HTML : Basics in Web Design, Brief History of Internet ,World Wide Web Why create a web site ,Web Standards, HTML Documents ,Basic structure of an HTML document Creating an HTML document ,Mark up Tags ,Heading-Paragraphs ,Line Breaks ,HTML Tags. Elements of HTML: Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls.

TASK 9: Basic HTML tags

a) Head Section and Elements of Head Section, Paragraphs, Formatting Styles.

b) Colour tags, Creating Hyperlinks, Images, Tables, lists

c) HTML Forms, Form Attributes, Form Elements.

TASK 10: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured. Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. Usage of search engines like Google, Yahoo, ask.com and others should be demonstrated by student.

TASK 11: Cyber Hygiene: Students should learn about viruses on the internet and install antivirus software. Student should learn to customize the browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

Text Books:

1. Fundamentals of Computers –ReemaThareja-Oxford higher education
2. Computer Fundamentals, Anita Goel, Pearson Education, 2017
3. PC Hardware Trouble Shooting Made Easy, TMH
4. Programming the World Wide Web, 7^{th} Edition, Robet W Sebesta, Pearson, 2013.

Reference Books:

1. An Introduction to Web Design, Programming, 1^{st} Edition, Paul S Wang, Sanda S Katila, Cengage Learning, 2003.
2. An Introduction to Computer studies –Noel Kalicharan-Cambridge

Course Code	Course Name	Course Structure			
		L	T	P	C
P21HSL01	English Language Communication Skills Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: Basic knowledge of English grammar, Basic understanding of English vocabulary, Ability to speak simple sentences, Have interest to learn the language.

Course Objectives: The student will be able

1. To facilitate computer assisted multimedia instructions enabling individualized and independent language learning.
2. To sensitize the students to the nuances of English speech sounds, word accent, intonation and rhythm.
3. To bring about a consistence accent and intelligibility in students' pronunciation of English by providing an opportunity for practice in speaking.
4. To improve the fluency of students in spoken English and neutralize their mother tongue influence.
5. To train students to use language appropriately for public speaking, group discussion and interviews.

Course Outcomes: After going through this course the student will be able to

1. Better understanding of nuances of English language through audio visual experience and group activities.
2. Neutralization of accent for intelligibility.
3. Speaking skills with clarity and confidence which in turn enhances their employability skills.
4. Better understanding of the production of sounds of language.
5. Suitable body language for employability.

EXERCISE–I **(3 Sessions)**

A. Introduction to Phonetics
Consonant sounds
Vowel sounds – Pure Vowels & Diphthongs

B. Greeting, Introducing & taking leave and Ice – Breaking Activity

EXERCISE–II **(2 Sessions)**

A. Structure of Syllables - Plural markers & Past tense Markers

B. JAM Session & Situational Dialogues

EXERCISE–III **(2 Sessions)**

A. Word Stress & Rules of 'r' pronunciation

B. Role play, Giving Directions & Story Narration

EXERCISE–IV **(2 Sessions)**

A. Consonant Cluster, Neutralization of Mother Tongue Influence and Listening Comprehension – Listening for General Details

B. Describing objects, events, places etc. & Presentation Skills – Extempore, Public Speaking.

EXERCISE–V **(3 Sessions)**

A. Intonation & Listening Comprehension – Listening for Specific Details

B. Interview Skills & Group Discussion

Text Books:

1. Strengthen your Communication Skills - Maruthi Publication, Hyderabad 2013
2. A textbook of English Phonetics for Indian Students by T. Balasubramanian (Macmillan)

Reference Books:

1. INFOTECH English (Maruthi Publications).
2. Personality Development and Soft Skills (Oxford University Press, New Delhi)
3. Suresh Kumar, E. & Sreehari, P. 2009. A Handbook for English Language Laboratories. New Delhi: Foundation
4. Speaking English Effectively 2^{nd} Edition by Krishna Mohan and N. P. Singh, 2011. Macmillan Publishers India Ltd. Delhi.
5. Sasi Kumar, V & Dhamija, P.V. How to Prepare for Group Discussion and Interviews. Tata McGraw Hill
6. Hancock, M. 2009. English Pronunciation in Use. Intermediate. Cambridge: CUP
7. Spoken English: A Manual of Speech and Phonetics by R. K. Bansal & J. B. Harrison. 2013. Orient Blackswan. Hyderabad
8. Hewings, M. 2009. English Pronunciation in Use. Advanced. Cambridge: CUP
9. Marks, J. 2009. English Pronunciation in Use. Elementary. Cambridge: CUP
10. Nambiar, K.C. 2011. Speaking Accurately. A Course in International Communication. New Delhi : Foundation

11. Soundararaj, Francis. 2012. Basics of Communication in English. New Delhi: Macmillan

12. Spoken English (CIEFL) in 3 volumes with 6 cassettes, OUP.

13. English Pronouncing Dictionary Daniel Jones Current Edition with CD.

Web Resources:

1. `http://www.cambridge.org`

2. `http://www.edufind.com/english/practice`

3. `http://www.learnenglish.com`

4. `http://www.penguin.co.uk`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21BSL01	Applied Physics Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: The basics of analytical and conceptual understanding of physics.

Course Objectives:

1. Deploy scientific method of experiments in the laboratory.
2. Develop the procedures and observational skills for appropriate use of simple and complex apparatus.
3. Enable analytical techniques, statistical analysis and graphical analysis.
4. Reinforce ideas and concepts covered in lecture host of experiments.
5. Train to find the radius of curvature of a Plano-convex lens forming Newton's rings.

Course Outcomes:

1. Apply the phenomenon of interference and diffraction of light waves.
2. HImplement the concept of resonance in LCR circuit and sonometer.
3. HM toAnalyze the S determine its dependent properties.
4. Evaluate the behavior of electronic components and its characteristics.

LIST OF EXPERIMENTS: (any eight of the following to be done)

1. Determination of Radius of Curvature of Plano-Convex lens by forming Newton's Rings.
2. Determination of Wavelengths of various spectral lines using diffraction grating with the normal incidence method.
3. Study of magnetic field along the axis of a current carrying coil and to verify Stewart-Gee's method.
4. Determination of energy gap of PN junction Diode.
5. Determination of hall coefficient and carrier concentration using Hall effect
6. Study of V-I characteristics of Zener diode.
7. Determination of frequency of a vibrating bar or electrical tuning fork using Melde's apparatus.
8. Determination of acceleration due to gravity using compound pendulum
9. Verification of laws of transverse waves by Sonometer.
10. Determination of Velocity of sound by volume resonator.
11. Determination of rigidity modulus by Torsional Pendulum.

Text Books:

1. Physics lab manual, department of physics, PACE Institute of Technology and Sciences.
2. Madhusudhanrao, “Engineering Physics lab manual” 1^{st} edition, Scietech Publication, 2015.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21ESL02	C - Programming for Problem Solving Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: Nil

Course Objectives: The student will be able

1. To understand the various steps in program development.
2. To understand the basic concepts in C Programming Language.
3. To understand different modules that includes conditional and looping expressions.
4. To understand how to write modular and readable C Programs.
5. To write programs in C to solve problems using arrays, structures and files.

Course Outcomes: After going through this course the student will be able to

1. Apply and practice logical ability to solve the problems.
2. Understand C programming development environment, compiling, debugging, and linking and executing a program using the development environment.
3. Analyzing the complexity of problems, modularize the problems into small modules and then convert them into programs.
4. Understand and apply the in-built functions and customized functions for solving the problems.
5. Understand and apply the pointers, memory allocation techniques and use of files for dealing with a variety of problems.
6. Document and present the algorithms, flowcharts, and programs in form of user-manuals

Experiment Wise Programs

EXERCISE–I

a. Write a simple C program to Print “Hello World”

b. Write a simple C Program to Calculate Area and Circumference of Circle

c. Write a simple C program to implement basic arithmetic operations - sum, difference, product, quotient and remainder of given numbers.

EXERCISE–II: Write C programs to demonstrate the following operators

a. Assignment Operator.

b. Relational and Logical Operator.

c. Increment and decrement operator.

d. Bitwise operators.

e. Ternary operator.

EXERCISE–III

a. Write a C programs - to find the largest and smallest of 2 numbers(if – else), to find the largest and smallest of 3 numbers(Nested if – else), roots of quadratic equation(else – if ladder).

b. The total distance travelled by vehicle in‘t’ seconds is given by distance=ut+1/2 at 2 where ‘u’ and ‘a’ are the initial velocity and acceleration.

Write a c program to find the distance travelled at regular intervals of time given the Values of ’u’ and ‘a’. The program should provide the flexibility to the user to select his own time intervals and repeat the calculations for different values of ‘u’ and ‘a’.

c. Write a c program, which takes two integer operands and one operator from the user, performs the operation and the prints the result. (consider the operators $+, -, *, /, \%$ and use switch statement).

EXERCISE–IV

a. Write a C program to find the sum of individual digits of a positive integer

b. A Fibonacci Sequence is defined as follows: the first and second terms in the sequence are 0 and 1.Subsequent terms are found by adding the preceding two terms in the sequence .Write a c program to generate the first n terms of the sequence.

c. Write a c program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.

EXERCISE–V

a. Write a C program to read in two numbers, x and n, and then compute the sum of this geometric progression: $1 + x + x^2 + x^3 + ... + x^n$

b. Write a C program to generate Pascal’s triangle.

c. Write a C program to construct a pyramid of numbers

EXERCISE–VI

a. Write a c program to find both the largest and smallest number in a list of integers.

b. Write a c program that uses functions to perform the following:

   i. Addition of Two Matrices.

   ii. Multiplication of Two Matrices.

EXERCISE–VII

a. Write a programs that use both recursive and non-recursive functions

b. To find the factorial of a given integer.

c. To find the GCD of two given integers.

EXERCISE–VIII

a. Write a c program that uses functions to perform the following operations:

   i. To insert a sub-string in given main string from a given position.

   ii. To delete n Characters from a given position in a given string.

b. Write a C program to determine if the given string is a palindrome or not.

EXERCISE–IX

a. Write a C program that displays the position or index in the string S Where the string T begins, or - 1 if S doesn't contain T.

b. Write a C program to count the lines, words and characters in a given text .

EXERCISE–X

a. Write a program to print the details of a student like(Name, Rollno, marks) using nested structures.

b. Write a C Program to Calculate Difference Between Two Time Period.

EXERCISE–XI

a. Write a C program that uses functions to perform the following operations:

   i. Reading a complex number

   ii. Writing a complex number

   iii. Addition of two complex numbers

   iv. Multiplication of two complex numbers
   (Note: represent complex number using a structure.)

EXERCISE–XII

a. Write a C program which copies one file to another and display the contents of a file

b. Write a C program to reverse the first n characters in a file.

c. Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file)

Course Code	Course Name	Course Structure			
		L	T	P	C
P21BST04	Applied Chemistry	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Basic Chemistry at Intermediate or equivalent level.
Course Objectives: The student will be able

1. To analyze the different types of composite plastic materials and interpret the mechanism of conduction in conducting polymers.
2. To utilize the theory of construction of electrodes, batteries and fuel cells in redesigning new engineering products and categorize the reasons for corrosion and study methods to control corrosion.
3. To understand various synthetic methods of nonmaterials for modern advances of engineering technology. Summarize the preparation of semiconductors; analyze the applications of liquid crystals and superconductors
4. To analyze the principles of different analytical instruments and their applications.
5. To Design models for energy by different natural sources.

Course Outcomes: After going through this course the student will be able to

1. Analyze the different types of composite plastic materials and interpret the mechanism of conduction in conducting polymers.
2. Utilize the theory of construction of electrodes, batteries and fuel cells in redesigning new engineering products and categorize the reasons for corrosion and study methods to control corrosion.
3. Understand various synthetic methods of nonmaterials for modern advances of engineering technology. Summarize the preparation of semiconductors; analyze the applications of liquid crystals and superconductors
4. Analyze the principles of different analytical instruments and their applications.
5. Design models for energy by different natural sources.

UNIT–I: Polymer Technology **(9 Lectures)**

Polymerization: Introduction, classification, methods of polymerization (Emulsion and Suspension), mechanical properties.
Plastics: Compounding, fabrication (compression, injection, blown film and extrusion), preparation, properties and applications (Poly ethylene, PVC, Polycarbonates and Bakelite).
Elastomers: Introduction, preparation, properties and applications (Buna S, Thiokol and Polyurethanes).
Composite materials: Fiber reinforced plastics, conducting polymers, biodegradable polymers,

UNIT–II: Electrochemical Cells and Corrosion (10 Lectures)

Galvanic Cells, Single electrode potential, electrochemical series and uses of series, standard hydrogen electrode, calomel electrode, construction of glass electrode, batteries (Dry cell, Li ion battery, Lead Acid battery and Ni-Cd cells).

Corrosion: Definition, theories of corrosion (Chemical and Electrochemical), galvanic corrosion, differential aeration corrosion, stress corrosion, galvanic series, corrosion control (Proper designing and cathodic protection), protective coatings (Surface preparation, Cathodic coatings, Anodic coatings, Electroplating and Electroless plating).

UNIT–III: Chemistry of Advanced Materials (10 Lectures)

Nano materials: Introduction, sol-gel method, characterization by (Brunauer Emmet Teller (BET), and transmission electron microscopy (TEM) with example (TiO2), applications of fullerenes, carbon nanotubes (types, preparation and applications).

Liquid crystals: Introduction-types-applications.

Super conductors: Type –I, Type II-characteristics and applications

Non-elemental semiconducting materials: Stoichiometric, controlled valency & chalcogen photo/ semiconductors preparation of semiconductors (zone refining, Czochralski crystal pulling, epitaxy, diffusion, ion implantation)– semiconductor devices (p-n junction diode as rectifier, junction transistor).

UNIT–IV: Spectroscopic Techniques & Synthesis Of Essential Drug Molecules (9 Lectures)

Spectroscopic Techniques: Electromagnetic spectrum-types of molecular spectra and their absorption criteria ,UV-visible spectroscopy (electronic spectroscopy), Beer-Lambert's law and its limitations ,– applications of UV visible spectroscopy ,IR spectroscopy principle, Molecular vibrations – stretching and bending vibrations – applications of IR, NMR (Nuclear magnetic resonance)–working principle and instrumentation of NMR, chemical shift(δ) – applications of NMR

Synthesis of essential drug molecules: Preparation, properties and uses of Paracetamol , Aspirin, Ibuprofen

UNIT–V: Non-Conventional Energy Sources (7 Lectures)

Design, working, schematic diagram, advantages and disadvantages of photovoltaic cell, hydropower, geothermal power, tidal and wave power, ocean thermal energy conversion.

Text Books:

1. P.C. Jain and M. Jain "Engineering Chemistry", 15/e, Dhanpat Rai & Sons, Delhi, (Latest edition).

2. Shikha Agarwal, "Engineering Chemistry", Cambridge University Press, New Delhi, (2019).

3. S.S. Dara, "A Textbook of Engineering Chemistry", S.Chand & Co, (2010).

4. Shashi Chawla, "Engineering Chemistry", Dhanpat Rai Publicating Co. (Latest edition).

Reference Books:

1. K. SeshaMaheshwaramma and MridulaChugh, “Engineering Chemistry”, Pearson India Edition.
2. O.G. Palana, “Engineering Chemistry”, Tata McGraw Hill Education Private Limited, (2009).
3. CNR Rao and JM Honig (Eds) “Preparation and characterization of materials” Academic press, New York (latest edition).
4. B. S. Murthy, P. Shankar, “Textbook of Nanoscience and Nanotechnology”, University press (latest edition).

Web Resources:

1. `http://jntuk-coeerd.in/`
2. `http://en.wikipedia.org/wiki/title`
3. `http://nptel.ac.in/coures/105106/.com`
4. `https://en.wikipedia.org/wiki/Electrochemistry`
5. `https://www.youtube.com/watch?v=WLyaZbT97EI&list=PLzW3l18TEXrpqo3jRarGr9ao-61tB2184`
6. `https://encyclopedia.che.engin.umich.edu/`
7. `http://encyclopedia.che.engin.umich.edu/Pages/ProcessParameters/Spectrometers/Spectrometers.html`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21BST06	Numerical Methods & Vector Calculus	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Differentiation, Partial differentiation, Integration, Differential Equations

Course Objectives: The student will be able to

1. The different numerical techniques to solve algebraic and transcendental equations and evaluate the polynomials from the numerical data.
2. The approximate solutions using numerical methods in the absence of analytical solutions of various systems of ordinary differential equations and integrations.
3. Enhance the knowledge level to visualize integrals in higher dimensional coordinate systems, possible representation and evaluation of geometrical and physical quantities in terms of multiple integrals.
4. Interpret concepts of vector functions, vector fields, differential calculus of vector functions in Cartesian coordinates and apply them for various engineering problems.
5. Evaluate line, surface and volume integrals and construct relation between line, surface and volume integrals using vector integral theorems.

Course Outcomes: After going through this course the student will be able to

1. Evaluate approximate roots of the polynomial and transcendental equations by different algorithms and apply Newton's forward, backward interpolation and Lagrange's formulae for equal and unequal intervals.
2. Apply different algorithms for approximating the integrals of numerical data and solutions of ordinary differential equations to its analytical computations.
3. S Evaluate the multiple integrals by using change of variables and change of order of integration. Also apply double and triple integration techniques in evaluating areas and volumes bounded by regions and solids.
4. nterpret the physical meaning of different operators such as gradient, curl and divergence.
5. Determine line, surface and volume integrals. Apply Green's, Stoke's and Gauss divergence theorems to calculate line, surface and volume integrals.

UNIT–I: Iterative Methods, Finite differences and Interpolation (10 Lectures)

Introduction-Solution of algebraic and transcendental equations-Bisection method -Method of false position-Newton-Raphson method (Single variable only)

Interpolation: Introduction-Errors in polynomial interpolation-Finite differences – Forward differences-Backward differences-Relations between operators-Newton's forward and backward formulae for interpolation -Interpolation with unequal intervals -Lagrange's interpolation formula.

UNIT–II: Numerical integration, Solution of ordinary differential equations with initial (9 Lectures)

Trapezoidal rule – Simpson's 1/3rd and 3/8th rule– Solution of ordinary differential equations by Taylor's series – Picard's method of successive approximations – Euler's method – Modified Euler's method-Runge-Kutta method (second and fourth order).

UNIT–III:Multiple Integrals: (9 Lectures)

Double integrals (Cartesian and Polar) – Change of order of integration – Change of variables (Cartesian to Polar coordinates) –Triple integrals- Change of variables (Cartesian to Cylindrical and Spherical coordinates).

Applications: Areas by double integrals and Volumes by triple integrals.

UNIT–IV: Vector Differentiation: (8 Lectures)

Scalar and Vector point functions-Vector Differential operator- Gradient – Directional derivatives – Divergence – Curl – Laplacian second order operator- Vector identities- Applications: Scalar Potential function.

UNIT–V: Vector Integration: (9 Lectures)

Line integral – Work done – Circulation- Surface integral- Volume integral

Vector Integral Theorems (without proof): Application of Green's theorem in a plane- Stoke's theorem- Gauss Divergence theorem.

Text Books:

1. B.S. Grewal, Higher Engineering Mathematics, 44^{th} Edition, Khanna Publishers.
2. B.V. Ramana, Higher Engineering Mathematics, 2007 Edition, Tata Mc. Graw Hill Education.

Reference Books:

1. Erwin Kreyszig, Advanced Engineering Mathematics, 10^{th} Edition, Wiley-India.
2. H. K. Das, Advanced Engineering Mathematics, 22^{nd} Edition,S. Chand & Company Ltd.
3. David Poole, Linear Algebra- A modern introduction, 4^{th} edition, Cengage.
4. Peter O' Neil, Advanced Engineering Mathematics, Cengage
5. Srimantha Pal, S C Bhunia, Engineering Mathematics, Oxford University Press.

Web Resources:

1. `http://tutorial.math.lamar.edu/Classes/DE/DE.aspx`
2. `http://mathworld.wolfram.com/topics`
3. `http://www.nptel.ac.in/course.php`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21EST13	Data Structures	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: C Programming
Course Objectives: The student will be able to

1. Comprehensive knowledge of data structures and ability to implement the same in software applications.
2. Exposure to algorithmic complexities, recursive algorithms, searching techniques.
3. Exposure to sorting technique, Applying stack techniques for logical operations.
4. Applying queue techniques for logical operations, Exposure to list representation models in various types of applications.
5. Implementation of tree in various forms, Advanced understanding of other variants of trees and their operations.
6. Orientation on graphs, representation of graphs, graph traversals, spanning trees Graphs.

Course Outcomes: After going through this course the student will be able to

1. Student will be able to choose appropriate data structure as applied to specified problem definition.
2. Implement appropriate sorting/searching technique for given problem.
3. Implement operations like searching, insertion, and deletion, traversing mechanism etc. on various data structures.
4. Students will be able to implement Linear and Non-Linear data structures.
5. Analyze variety of Graph data structures that are used in various applications.

UNIT–I: (9 Lectures) Data Structure, Recursion & Searching:
Preliminaries of algorithm, Algorithm analysis and complexity. Data Structure: Definition, types of data structures. **Recursion:** Definition, Design Methodology and Implementation of recursive algorithms, Types of recursion (Linear, binary and Tail), recursive algorithms for factorial function, GCD Computation, Fibonacci sequence.
Searching: List Searches using Linear Search, Binary Search.
UNIT–II: (10 Lectures)
Sorting Techniques: Basic Concepts, Sorting by: Insertion (Insertion Sort), Selection (heap sort), Exchange(Bubble sort, Quick Sort) Merging(Merge sort) Algorithms.
Stacks: Basic Stack operations, Representation of a stack using arrays, Stack Applications: Reversing list, Infix to postfix transformation.

UNIT–III: **(10 Lectures)**

Queues: Introduction, Representation of a Queue using arrays, Queue Operations, Applications of queues- Round Robin Algorithm, Circular Queues, Priority Queues.
Linked List: Introduction, single linked list, representation of a linked list in memory, Operations on a single linked list, Reversing a single linked list, Applications: single linked list to represent polynomial expressions, Double linked list. , Circular linked list

UNIT–IV: **(9 Lectures)**

Trees: Basic tree concepts, Binary Trees: Properties, Representation of Binary Trees using arrays, operations on a Binary tree, Binary Tree Traversals (recursive).
Advanced Tree Concepts: Binary search tree, Basic concepts, BST operations: Searching, insertion, deletion, Balanced search trees-AVL Trees.

UNIT–V: **(7 Lectures)**

Graphs: Basic concepts, Graph Representations- Adjacency matrix, Adjacency lists, Graph algorithms: Graph Traversals (BFS & DFS), applications: Dijkstra's shortest path, Minimum Spanning Tree using Prim's & Kruskal's Algorithm.

Text Books:

1. Data Structures, 2/e, Richard F, Gilberg , Forouzan, Cengage,2007.
2. Data Structures and Algorithms, G.A.V.Pai, TMH, 2008
3. Data Structures and Algorithms Made Easy, Narasimha Karumanchi , Second Edition, 2011.

Reference Books:

1. Data Structure with C, Seymour Lipschutz, TMH,2010.
2. Classic Data Structures, 2/e, Debasis ,Samanta,PHI,2009
3. Fundamentals of Data Structure in C, 2/e, Horowitz, Sahni, Anderson Freed, University Press.

Web Resources:

1. www.geeksforgeeks.org
2. www.hackr.io.
3. www.letsfindcourse.com

Course Code	Course Name	Course Structure			
		L	T	P	C
P21EST12	Digital Electronics	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NILL

Course Objectives:

1. Relate the conversion among different number systems.
2. Outline of basic logic gates – AND, OR & NOT, XOR, XNOR and understand Boolean algebra and basic properties of Boolean algebra.
3. Able to optimize simple logic using Karnaugh maps, understand ”don’t care” concepts.
4. Design simple combinational using basic gates.
5. Understand different memories and able to design different programmable devices.

Course Outcomes:

1. Demonstrate the various number systems and conversion of number systems.
2. Develop Boolean algebra & the underlying features ofvarious logic gates.
3. Conceptualize Design mapping method upto 4-variables.
4. Apply the concepts of Boolean algebra for the analysis & design of various combination logic circuits.
5. Able to compare different memories and their programmable devices.

UNIT–I: **(9 Lectures)**

Number Systems and Binary Codes : Number System, Types of Number Systems, Number base Conversions from one radix to another radix, Representation of Signed Binary Numbers,1’s complement arithmetic, 2’s complement arithmetic. Gray code, Excess-3 code, BCD code. Conversions.

UNIT–II: **(9 Lectures)**

Boolean algebra : Logic gates, Laws of Boolean algebra, Principle of Duality, Principle of Complements, Reducing Boolean Expressions using Boolean algebra, Canonical and Standard Forms, M- Notations: Minterms and Maxterms.

UNIT–III: **(9 Lectures)**

Gate level Minimization : Map Method, Two-Variable K-Map, Three-Variable K-Map, Four Variable K-Maps: Sum of Products Simplification, Products of Sum Simplification, Don’t – Care Conditions. Implementation using NAND and NOR.

UNIT–IV: **(9 Lectures)**

Combinational Logic Design: Introduction, Design Procedure, Adders, Subtractors, Binary Adder–Subtractor, Decoders, Encoders, Multiplexers and Demultiplexers.

UNIT–V: **(9 Lectures)**

Programmable Logic Devices: Classification of memories, PROM,PAL,PLA – basic Structures, Realization of Boolean function with PLDs , Comparison of PROM, PAL, PLA.

Text Books:

1. Digital Design, 5/e, M.Morris Mano, Michael D Ciletti, PEA, 2011.
2. Fundamentals of Logic Design, 5/e, Roth, Cengage, 2010.
3. A.K.Singh,Digital Logic Circuits, New Age International Publishers.

Reference Books:

1. Switching Theory and Logic Design, A.Anand Kumar, 2016.
2. Digital Electronics and Logic Design, Dr. Sanjay Sharma, 2010.
3. Modern Digital Electronics, R.P. Jain, TMH, 2010.

Web Resources:

1. `www.researchgate.net`
2. `www.digital-logic-design.en.softonic.com`
3. `https://nptel.ac.in/courses/117/106/117106086/`
4. `https://www.coursera.org/learn/digital-systems`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21EST14	Python Programming	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NILL

Course Objectives:

1. To read and write simple Python programs.
2. 2. To develop Python programs with conditionals and loops.
3. To define Python functions and apply OOP concept.
4. To use Python data structures — lists, tuples, dictionaries.
5. To develop GUI applications in Python.

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

1. Understand the basics of python programming.
2. Understand control flow and implement various data structures provided by python.
3. Implement packages, methods and functions.
4. Develop real-world applications using oops and exception handling.
5. Build GUI Applications in Python.

UNIT–I: **(8 Lectures)**

Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

UNIT–II: **(9 Lectures)**

Types, Operators and Expressions: Types - Integers, Strings, Booleans, Expressions and order of evaluations, Control Flow- if, if-elif-else, for, while, break, continue, pass.

Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences, Comprehensions..

UNIT–III: **(10 Lectures)**

Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions (Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

Modules: Creating modules, import statement, from. Import statement, name spacing,

Python packages: Introduction to PIP, Installing Packages via PIP, Using Python Packages

UNIT–IV: **(9 Lectures)**

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

Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions.

UNIT–V: **(9 Lectures)**

Brief Tour of the Standard Library & Files - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics, file operations.

Text Books:

1. Allen B. Downey, “Think Python: How to Think Like a Computer Scientist”, 2^{nd} edition, Updated for Python 3, Shroff/O’Reilly Publishers, 2016 (http://greenteapress.com/wp/thinkpython/).
2. Guido van Rossum and Fred L. Drake Jr, “An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.

Reference Books:

1. John V Guttag, “Introduction to Computation and Programming Using Python”, Revised and expanded Edition, MIT Press , 2013
2. Kenneth A. Lambert, “Fundamentals of Python: First Programs”, CENGAGE Learning, 2012.
3. Robert Sedgewick, Kevin Wayne, Robert Dondero, “Introduction to Programming in Python: An Inter-disciplinary Approach, Pearson India Education Services Pvt. Ltd., 2016.
4. Charles Dierbach, “Introduction to Computer Science using Python: A Computational Problem Solving Focus, Wiley India Edition, 2013.

Web Resources:

1. `https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/`
2. `https://www.codecademy.com/learn/learn-python`
3. `https://www.codementor.io/collections/learn-python-bwbc63ulz`
4. `http://www.diveintopython3.net/`
5. `https://www.python.org/3/`
6. `https://www.learnpython.org`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21BSL03	Applied Chemistry Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: Basic Chemistry at Intermediate or equivalent level.
Course Objectives: The purpose of this course to provide students with practical knowledge of quantitative analysis of materials by classical and instrumental methods for developing experimental skills in building technical competence.
Course Outcomes: After going through this course the student will be able to

1. Carrying out different types of titrations for estimation of concerned in materials using comparatively more quantities of materials involved for good results.
2. Handling different types of instruments for analysis of materials using small quantities of materials involved for quick and accurate results.

LIST OF EXPERIMENTS: Introduction to Chemistry laboratory – Molarity, normality, primary, secondary standard solutions, volumetric titrations, quantitative analysis.

1. Determination of HCl using standard Na_2CO_3 solution.
2. Determination of alkalinity of water sample containing Na_2CO_3 and NaOH.
3. Determination of Mn^{+2} using standard oxalic acid solution.
4. Determination of ferrous iron using standard $K_2Cr_2O_7$ solution.
5. Determination of Cu^{+2} using standard hypo solution.
6. Determination of temporary and permanent hardness of water using standard EDTA solution.
7. Determination of Fe^{+3} by a colorimetric method.
8. Determination of the concentration of acetic acid using sodium hydroxide ($p^{H}metry$ method).
9. Determination of isoelectric point of amino acids using $p^{H}metry$ method (or) conductometric method.
10. Determination of the concentration of strong acid vs strong base (by conductometric method).
11. Determination of strong acid vs strong base (by potentiometric method).
12. Estimation of Vitamin C.
13. Preparation of Nylon-6, 6 and Bakelite (demonstration only).

Reference Books:

1. Dr. Jyotsna Cherukuris (2012) Laboratory Manual of engineering chemistry-II,

2. VGS Techno Series 3. Chemistry Practical Manual, Lorven Publications

Web Resources:

1. `https://vlab.amrita.edu/index.php?sub=2&brch=193.`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21ESL06	Data Structures Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: C- Programming
Course Objectives: The objective of this lab is

1. To choose the appropriate data structure and algorithm design method for a specified application.
2. To solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps binary search trees, and graphs and writing programs for these solutions.

Course Outcomes: At the end of the course student can able to

1. Analyze worst-case running times of algorithms using asymptotic analysis and implement various data structures like linked lists.
2. Understand and implement stacks and queues using arrays and linked lists.
3. Analyze and implement various searching and sorting algorithms.
4. Design and implement appropriate hash function and collision-resolution algorithms
5. Experiment with different graph traversal techniques.

List of Experiments:
Exercise 1:
Write recursive program for the following

a. Write recursive C program for calculation of Factorial of an integer
b. Write recursive C program for calculation of GCD (n, m)
c. Write recursive program which computes the n^{th} Fibonacci number

Exercise 2:

a. Write recursive C program for functions to perform Linear search for a Key value in a given list.
b. Write recursive C program for functions to perform Binary search for a Key value in a given list.

Exercise 3:

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

Exercise 4:

a. Write C program that implement Insertion sort, to sort a given list of integers in ascending order
b. Write C program that implement merge sort, to sort a given list of integers in ascending order

Exercise 5:

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

Exercise 6:

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

Exercise 7:

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

Exercise 8:

a. Write a C program to Create a Binary Tree of integers
b. Write a recursive C program for Traversing a binary tree in preorder, inorder and postorder.

Exercise 9:

Write a C program for BST operations (insertion, deletion)

Exercise 10:

a. Write a C program for finding minimum spanning tree in a graph by using Prim's algorithm.
b. Write a C program for finding minimum spanning tree in a graph by using Kruskal's algorithm.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21ESL07	Python Programming Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Outcomes: At the end of the course student can able to

1. ADescribe the Numbers, Math functions, Strings, List, Tuples and Dictionaries in Python.
2. Express different Decision Making statements and Functions.
3. Interpret Object oriented programming in Python.
4. Understand File handling operations.
5. Design GUI Applications.

Exercise1 - Basics

a. Running instructions in Interactive interpreter and a Python Script

b. Write a program to purpose fully raise Indentation Error and Correct it

Exercise 2 - Operations

a. Write a program to compute distance between two points taking input from the user (Pythagorean Theorem).

b. Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

Exercise 3 – Control Flow

a. Write a Program for checking whether the given number is a even number or not.

b. Using a for loop, write a program that prints out the decimal equivalents of 1/2,1/3,1/4,...,1/10.

c. Write a program using a for loop that loops over a sequence. What is sequence?

d. Write a program using a while loop that asks the user for a number, and prints a count down from that number to zero.

Exercise 4 – Control Flow-Continued

a. Find the sum of all the primes below two million. Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be:1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

b. By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Exercise 5 - DS

a. Write a program to count the numbers of characters in the string and store them in a dictionary data structure.
b. Write a program to use split and join methods in the string and trace a birth day with a dictionary data structure.

Exercise 6- DS-Continued

a. Write a program combine lists that combines these lists into a dictionary.
b. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

Exercise 7 - Files

a. Write a program to print each line of a file in reverse order.
b. Write a program to compute the number of characters, words and lines in a file.

Exercise 8 - Functions

a. Write a function dups to find all duplicates in the list.
b. b) Write a function unique to find all the unique elements of a list.

Exercise 9 - Functions –Problem Solving

a. Write a function cumulative product to compute cumulative product of a list of numbers.
b. Write a function reverse to reverse a list. Without using the reverse function.
c. Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

Exercise 10 – Multi - D Lists

a. Write a program to perform addition of two square matrices.
b. Write a program to perform multiplication of two square matrices.

Exercise 11 - OOP

Class variables and instance variable and illustration of the self variable
i)Robot.
ii)ATM Machine.

Exercise - 12 GUI, Graphics

a. Write a GUI for an Expression Calculator using tk.
b. Write a program to implement the following figures using turtle

Course Code	Course Name	Course Structure			
		L	T	P	C
P21BST07	Probability and Statistics	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Principle of counting, Permutations and Combinations.

Course Objectives:

1. To familiarize the students with the foundations of probability and statistical methods.
2. To impart probability concepts and statistical methods in various applications of engineering.

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

1. Explain the basic terms of Statistical Inference.
2. Interpret the association of characteristics and through correlation and regression tools.
3. Make the use of the concepts of probability and their applications. Also apply discrete and continuous probability distributions to solve various engineering problems.
4. Understand the various distribution and sampling and identify the estimation errors in sampling distributions.
5. Apply the proper test statistics to test the hypothetical data by Tests of Hypothesis.

UNIT–I: Descriptive Statistics (9 Lectures)

Introduction - Measures of Central tendency - Measures of Variability (Spread or variance) - Moments – Skewness - Kurtosis.

UNIT–II: Curve Fitting and Correlation and Regression (9 Lectures)

Method of least squares - Straight line - Parabola-Exponential curve - Power curve – Correlation - Correlation coefficient - Rank correlation - Regression and Regression lines.

UNIT–III: Probability Theory and Random Variable: (14 Lectures)

Probability Theory: (14 Lectures)Probability - Axioms of Probability - Elementary theorems - Conditional probability - Baye’s theorem (Without Proofs).

Random Variables: Discrete random variable - Distribution function of a discrete random variable - Probability mass function: Properties - Mean and Variance - Continuous random variable - Distribution function - Density function: Properties-Mean and variance.

Probability Distributions: Binomial distribution - Poisson distribution and their fitting to data - Normal distribution - Mean and Variance (Without proof).

UNIT–IV: Sampling theory and Theory of estimation (9 Lectures)

Sampling Theory: Introduction - Population and Samples - Sampling distribution of means (σ known)-Central limit theorem (without proof).

Theory of estimation: Point estimation- Interval estimation - Estimation of one mean and two means - Estimation of one proportion and two proportions.

UNIT–V: Tests of Hypothesis: **(9 Lectures)**

Introduction – Hypothesis - Null and Alternative Hypothesis - Type I and Type II errors - Level of significance - One tail and two-tail tests -Tests concerning one mean and two means (Large and Small samples) -Tests on proportions.

Text Books:

1. S. C. Gupta and V. K. Kapoor, Fundamentals of Mathematical Statistics, 11/e (Reprint) 2019, Sultan Chand & Sons Publications.
2. Miller and Freund's,Probability and Statistics for Engineers, 7/e, Pearson, 2008.

Reference Books:

1. T. K. V. Iyenger, Probability and Statistics, S. Chand & Company Ltd, 2015
2. Jay I. Devore, Probability and Statistics for Engineering and the Sciences, 8^{th} Edition, Cengage.
3. Sheldon M. Ross, Introduction to probability and statistics Engineers and the Scientists, 4^{th} Edition, Academic Foundation, 2011.

Web References:

1. `https://leanpub.com/LittleInferenceBook`
2. `https://www.coursera.org/learn/statistical-inference`
3. `https://www.datacamp.com/community/open-courses/statistical-inference-and-data-analysis`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CIT01	Object Oriented Programming Through C++	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NIL

Course Objectives:

1. Understand the syntax and principles of Object Oriented Programming.
2. Design and development of secure and extendable C++ applications.
3. Describe the concept of function overloading, operator overloading, virtual functions and polymorphism.
4. Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming.
5. Demonstrate the use of various OOP's concepts with the help of programs.

Course Outcomes:

1. Illustrate the process of Object Oriented Paradigm.
2. Demonstrate classes, member functions, constructors and their importance in developing real world applications.
3. Apply C++ features such as Inheritance, operator overloading to make programs reusable.
4. Understand Dynamic Memory Management techniques using pointers.
5. Apply the concept of Generic Programming and Exception handling to build an efficient and error free code.

UNIT–I: **(9 Lectures)**

Introduction to C++: Difference between C and C++, Evolution of C++, The Object Oriented Technology, Disadvantage of Conventional Programming, Key Concepts of Object Oriented Programming, Advantages of OOP's, Object Oriented Language.

UNIT–II: **(9 Lectures)**

Classes and Objects: Classes in C++, Declaring Objects, Access Specifiers and their Scope, Defining Member Function, Overloading Member Function, Nested class.

Constructors and Destructors: Introduction, Constructors and Destructor, Characteristics of Constructor and Destructor, Application with Constructor, Constructor with Arguments, Parameterized Constructor, Destructors, Anonymous Objects.

UNIT–III: **(9 Lectures)**

Operator Overloading, Type Conversion and Inheritance: The Keyword Operator, Overloading Unary Operator, Operator Return Type, Overloading Assignment Operator (=), Rules for Overloading Operators, Inheritance, Reusability, Types of Inheritance, Virtual Base Classes, Object as a Class Member, Abstract Classes, Advantages of Inheritance, Disadvantages of Inheritance.

UNIT–IV: **(9 Lectures)**

Pointers: Pointer, Features of Pointers, Pointer Declaration, Pointer to Class, Pointer Object, The this Pointer, Pointer to Derived Classes and Base Class. **Binding Polymorphisms and Virtual Functions:** Introduction, Binding in C++, Virtual Functions, Rules for Virtual Function, Virtual Destructor.

UNIT–V: **(9 Lectures)**

Generic Programming with Templates & Exception Handling: Definition of class Templates, Normal Function Templates, Over Loading of Template Function, Bubble Sort Using Function Templates, Difference between Templates and Macros, Exception Handling, Principles of Exception Handling, and The Keywords-try throw and catch, Multiple Catch Statements, Specifying Exceptions. Overview of Standard Template Library.

Text Books:

1. A First Book of C++, 4th Edition, Gary Bronson, Cengage Learning.
2. The Complete Reference, C++, 5th Edition, Herbert Schildt, McGraw-Hill Education.

Reference Books:

1. Object Oriented Programming C++, Joyce Farrell, Cengage Learning.
2. C++ Programming: from problem analysis to program design, 6th Edition, DS Malik, Cengage Learning.
3. Programming in C++, Ashok N Kamthane, and Pearson.
4. Object Oriented Programming using C++, 8th Edition, E. Balaguruswamy, PHI.

Web References:

1. `https://nptel.ac.in/courses/106/105/106105151/`
2. `https://github.com/topics/object-oriented-programming`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST02	Front End Web Technologies	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Object Oriented Programming

Course Objectives:

1. To develop the static web pages using HTML and CSS.
2. To enables the students to identify the fundamental concepts for developing web application using PHP language for server side scripting.
3. To analyze how data can be transported using XML.
4. To develop a web applications with server side programming using java servlets.
5. To develop a web applications with server side programming using JSP.

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

1. Summarize the basic tags and properties in HTML, XHTML and CSS.
2. Create web pages using .client side scripting, validating of forms and XML.
3. Identify the role of server side scripting using PHP programming
4. Design dynamic web application using server side programming with java servlets.
5. Contraston how to connect and retrieve data through web page from database usingJDBC.

UNIT–I: **(9 Lectures)**

HTML Common tags- List, Tables, images, forms, Frames, Links and Navigation, **CSS:** Introduction, CSS Properties, Controlling Fonts, Text Formatting, Pseudo classes, Selectors.

UNIT–II: **(9 Lectures)**

Client side Scripting: Introduction to Javascript: Javascript language – declaring variables, scope of variables, functions, event handlers (onclick, onsubmit etc.), Document Object Model, Form validation.

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model.

UNIT–III: **(9 Lectures)**

Introduction to PHP: Creating PHP script, Running PHP script, Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies.

UNIT–IV: **(9 Lectures)**

A: Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a servlet, deploying a servlet,

B: The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions.

UNIT–V: **(9 Lectures)**

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, JSP application design with MVC, Declaring variables and methods, sharing data between JSP pages, Requests and users passing control and data between pages, Sharing sessions and application data.

JDBC connectivity in JSP: Data base programming using JDBC, Studying javax.sql.* package, Accessing a database from a JSP page, Application specific database actions.

Text Books:

1. Web Technologies, Uttam K Roy, Oxford UniversityPress
2. The Complete Reference PHP – Steven Holzner, TataMcGraw-Hill

Reference Books:

1. JWeb Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
2. Java Server Pages –Hans Bergsten, SPDO'Reilly
3. Java Script, D. Flanagan,O'Reilly,SPD.
4. Beginning Web Programming-Jon DuckettWROX.
5. Programming World Wide Web, R. W. Sebesta, Fourth Edition,Pearson.
6. Internet and World Wide Web – How to program, Dietel and Nieto,Pearson.

Web References:

1. `https://www.w3schools.com/html/`
2. `https://www.javatpoint.com/servlet-tutorial`
3. `http://nptel.ac.in/courses/106105084/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST03	Computer Organization	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NILL

Course Objectives:

1. Understand the architecture of a modern computer with its various processing units and Performance measurement of the computer system
2. To understand various data transfer techniques in digital computer.
3. To understand the memory management system of computer.

Course Outcomes:

1. Ability to understand basic structure of comTo perform computer arithmetic operations.
2. To understand control unit operations.
3. To design memory organization that uses banks for different word size operations.
4. Ability to understand the concept of cache mapping techniques.
5. Ability to understand the concept of I/O organization.

UNIT–I: **(9 Lectures)**

Basic Structure of Computers: Organization and Architecture, Structure and Function, Computer Components, Computer Function, Bus Interconnection, Processor Organization, Register Organization.

Basic Computer Organization and Design: Instruction codes, Computer instructions, Memory reference instructions, Instruction Cycle. Central Processing Unit: Stack organization, instruction formats, addressing modes, data transfer and manipulation, program control, RISC.

UNIT–II: **(10 Lectures)**

Register Transfer and Micro Operations: Register transfer language, Register transfer, Bus and Memory transfers, Arithmetic Micro operations, Logic Micro operations, Shift Micro operations, Arithmetic Logic Shift Unit.

Micro Programmed Control: Control Memory, Address Sequencing, Micro Program examples, Design of control unit, Hardwired control.

UNIT–III: **(9 Lectures)**

Computer Arithmetic: Data representation- Fixed point representation, Floating point representation, Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating-point Representations, Floating-point Arithmetic Operations, Decimal Arithmetic Units, Decimal Arithmetic Operations.

UNIT–IV: **(9 Lectures)**

Memory Organization: Memory system overview, Memory Hierarchy, Semi-conductor Main Memory, Cache Memory principle, Elements of cache design, Virtual Memory, Magnetic Disk, Optical Memory, Magnetic Tape, RAID.

Input- Output: External Devices, I/O modules, Interrupts, Programmed I/O, Interrupt-driven I/O, Direct Memory Access, I/O Channels and Processors, PCI. Asynchronous Data Transfer, Priority Interrupt, Serial Communication.

UNIT–V: (9 Lectures)

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

Multi Processors: Multiprocessors and Multi computers, Characteristics of Multiprocessors, Multiple Processor Organizations, Symmetric Multi-Processors, Cache Coherence, Clusters, Non Uniform Memory Access (NUMA).ons.

Text Books:

1. Computer Systems Architecture, M. Moris Mano, 3rd Edition, Pearson Education, 2007.
2. Computer Organization and Architecture, William Stallings, 8th Edition, Pearson Education, 2010.
3. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill,2012.

Reference Books:

1. Computer Systems Organization and Architecture, John D. Carpinelli, 3rd Edition, Pearson Education, 2001.
2. Computer Organization, Carl Hamacher, Zvonks Vranesic, SafeaZak , 5th Edition, TMH, 2011.

Web References:

1. `www.hackr.io`
2. `www.nptel.ac.in`
3. `www.coursera.org`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST04	Software Engineering	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NILL

Course Objectives:

1. To understand the software life cycle models.
2. To understand the software requirements and SRS document.
3. To understand the importance of modeling and modeling languages.
4. To design and develop correct and robust software products.
5. To understand the quality control and how to ensure good quality software.
6. To understand the planning and estimation of software projects.
7. To understand the implementation issues, validation and verification procedures.
8. To understand the maintenance of software

Course Outcomes:

1. Compare and contrast basic software engineering methods and practices.
2. Analyze the software process models.
3. Analyze the project management essentials.
4. Outline the importance of software testing and quality control approaches.
5. Analyze the Software Reliability and Software Maintenance.

UNIT–I: (9 Lectures)

Software and Software Engineering: The Nature of Software, The Unique Nature of WebApps, Software Engineering, Software Process, Software Engineering Practice, Software Myths.

Process Models: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Terminology, Product and Process.

UNIT–II: (10 Lectures)

Requirements Analysis And Specification: Requirements Gathering and Analysis, Software Requirement Specification (SRS), Formal System Specification.

Software Design: Overview of the Design Process, How to Characterise of a Design?, Cohesion and Coupling, Layered Arrangement of Modules, Approaches to Software Design

UNIT–III: (9 Lectures)

Function-Oriented Software Design: Overview of SA/SD Methodology, Structured Analysis, Developing the DFD Model of a System, Structured Design, Detailed Design, Design Review, over view of Object Oriented design.

User Interface Design: Characteristics of Good User Interface, Basic Concepts, Types of User Interfaces, Fundamentals of Component-based GUI Development, A User Interface Design Methodology.

UNIT–IV: **(9 Lectures)**

Coding And Testing: Coding, Code Review, Software Documentation, Testing, Unit Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tool, Integration Testing, Testing Object-Oriented Programs, System Testing, Some General Issues Associated with Testing

UNIT–V: **(9 Lectures)**

Software Reliability And Quality Management: Software Reliability, Statistical Testing, Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity Model.

Software Maintenance: Software maintenance, Maintenance Process Models, Maintenance Cost, Software Configuration Management.

Text Books:

1. Software engineering A practitioner's Approach, Roger S. Pressman, Seventh Edition McGrawHill International Edition.
2. Fundamentals of Software Engineering, Rajib Mall, Third Edition, PHI.
3. Software Engineering, Ian Sommerville, Ninth edition, Pearson education

Reference Books:

1. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
2. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.

Web References:

1. `https://www.tutorialspoint.com/software_engineering`
2. `https://nptel.ac.in/courses/106101061/1`
3. `http://ceit.aut.ac.ir/~91131079/SE2/SE2%20Website/Lecture%20Slides.html`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CIL01	Object Oriented Programming Through C++ Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: C- Programming

Course Objectives:

1. To get a clear understanding and to solve problems using Object-Oriented programming paradigms.

Course Outcomes:

1. Develop C++ programs with basic programming knowledge.
2. Develop C++ programs using Classes & Object Concepts.
3. Develop C++ programs using overloading Concepts.
4. Develop C++ programs using Constructors and Destructors.
5. Develop C++ programs using advanced Concepts.

Exercise1

1. Write a program to find the sum of given non-negative integer numbers using a recursive function.
2. Write a program to find the factorial of the given number using the recursive function.
3. Write a function in C++ to generate a Fibonacci series of n numbers, where n is defined by a programmer.

Exercise 2

1. Write a C++ Program to demonstrate parameter passing methods.
2. Write a program in C++ to solve a general quadratic equation.ax2+bx+c=0

Exercise 3

Write a C++ program to perform the following

1. Matrix Addition
2. Matrix Subtraction
3. Matrix Multiplication
4. Transpose of a Matrix.

Exercise 4:

1. Write a C++ Program to demonstrate function overloading.

2. Write a C++ Program to demonstrate overload the following operators.
    - (a) Arrow operator
    - (b) Function call operator
    - (c) Insertion Operator(¡¡)
    - (d) Extraction Operator

Exercise 5

Write a program to perform the following arithmetic operations of a complex number using a class.

1. Addition of two complex numbers
2. substation of two complex numbers
3. Multiplication of tow complex numbers
4. Division of tow complex numbers

Exercise 6

1. Write a C++ Program to demonstrate template functions.
2. Write a C++ Program to demonstrate template class

Exercise 7

1. Write a C++ program to implement the following
    - (a) Single inheritance
    - (b) Multiple inheritances
    - (c) Multi-Level inheritance
    - (d) Hybrid-inheritance.
2. Write a C++ program to demonstrate constructor and destructor calling sequence.

Exercise 8

Write a C++ program to demonstrate the usage of C++ Exception Handling mechanism.

Exercise 9

Write a C++ program to demonstrate runtime polymorphism

Exercise 10

Write a C++ program to demonstrate following

1. This pointer
2. Static data member

3. Static member function.

Exercise 11:

Write a C++ program to demonstrate following

1. Friend function
2. Friend member function
3. Friend class

Exercise - 12

Implement stack and queue data structures using templates.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL02	Front End Web Technologies Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: Object-Oriented Programming

Course Objectives:

1. To develop the static web pages using HTML and CSS.
2. To enables the students to identify the fundamental concepts for developing web application using PHP language for server side scripting.
3. To analyze how data can be transported using XML.
4. To develop a web applications with server side programming using java servlets.
5. To develop a web applications with server side programming using JSP.

Course Outcomes:

1. Create static web pages using HTML and CSS.
2. Develop JavaScript code for data validation.
3. Integrate frontend and backend technologies in client-server systems.
4. Design dynamic web applications using PHP and JSP.
5. Demonstrate database connectivity for developing web applications.

The students have to choose one of the following project and do the all 12 experiments related to that project.

1. Training and placementcell.
2. School EducationSystem.
3. University ManagementSystem.
4. Hospital ManagementSystem.

The following are the experiments related to Training and Placement cell project. For the remaining projects, the concern lab instructor has to decide the experiments according to the websites given as examples.

List of Experiments:

Experiment 1:

Design the following static web pages required for a Training and placement cell web site.

1) Home Page 2) Login Page 3) Registration page

Experiment 2:

4) Company Details Page 5) Alumni Details Page 6) Placement Staff Details Page

Experimen 3:

7) Student personal Info Page 8) Student Academic Info page 9) SemesterWise Percentage & their Aggregate page

Experiment 4:

Validate login page and registration page using regular expressions.

Experiment 5:

Apply different font styles, font families, font colors and other formatting styles to the above static web pages.

Experiment 6:

Install wamp server and tomcat server, access above developed static web pages using these servers.

Experiment 7:

Write a servlet/PHP to connect to the database, Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration.

Experiment 8:

Write a JSP/PHP to connect to the database, Insert the details of the student academic information with student academic info page.

Experiment 9:

User Authentication:

Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a servlet for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in thecookies.

If he is a valid user (i.e., user-name and password match) you should welcome him by name (user-name) else you should display "You are not an authenticated user ". Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters()method.

Experiment 10: Write a JSP which does the followingjob:

Authenticate the user when he submits the login form using the user name and password from the database.

Experiment 11:

write a JSP to insert the student's semester wise percentages and calculate aggregate and insert into database.

Experiment 12:

write a JSP to search the students according to their aggregate and produce sorted list or according to their Enroll number.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL03	Free Open Source Software Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: NIL

Course Objectives: To teach students various unix utilities and shell scripting

1. **Session-1**

   (a) Log into the system

   (b) Use vi editor to create a file called myfile.txt which contains some text.

   (c) Correct typing errors during creation.

   (d) Save the file

   (e) logout of the system

   Session-2

   (a) Log into the system

   (b) open the file created in session 1

   (c) Add some text

   (d) Change some text

   (e) Delete some text

   (f) Save the Changes

   (g) Logout of the system

2. (a) Log into the system

   (b) Use the cat command to create a file containing the following data. Call it mytable use tabs to separate the fields

   1425 Ravi 15.65

   4320 Ramu 26.27

   6830 Sita 36.15

   1450 Raju 21.86

   (c) Use the cat command to display the file, mytable.

   (d) Use the vi command to correct any errors in the file, mytable.

   (e) Use the sort command to sort the file mytable according to the first field. Call the sorted file my table (same name)

   (f) Print the file mytable

   (g) Use the cut and paste commands to swap fields 2 and 3 of mytable. Call it my table (same name)

   (h) Print the new file, mytable

- (i) Logout of the system
- (j) Login to the system
- (k) Use the appropriate command to determine your login shell
- (l) Use the /etc/passwd file to verify the result of step b.
- (m) Use the who command and redirect the result to a file called myfile1. Use the more command to see the contents of myfile1.
- (n) Use the date and who commands in sequence (in one line) such that the output of date will display on the screen and the output of who will be redirected to a file called myfile2. Use the more command to check the contents of myfile2.

3. Write a sed command that deletes the first character in each line in a file.
    - (a) Write a sed command that deletes the character before the last character in each line in a file.
    - (b) Write a sed command that swaps the first and second words in each line in a file.
4. Pipe your /etc/passwd file to awk, and print out the home directory of each user.
    - (a) Develop an interactive grep script that asks for a word and a file name and then tells how many lines contain that word.
    - (b) Repeat
    - (c) Part using awk
5. (a) Write a shell script that takes a command –line argument and reports on whether it is directory, a file, or something else.
    - (b) Write a shell script that accepts one or more file name as arguments and converts all of them to uppercase, provided they exist in the current directory.
6. Write a shell script that accepts a file name starting and ending line numbers as arguments and displays all the lines between the given line numbers.
7. (a) Write a shell script that computes the gross salary of a employee according to the following rules
    - i. If basic salary is $<$ 1500 then HRA =10% of the basic and DA =90% of the basic.
    - ii. If basic salary is $>=$1500 then HRA =Rs500 and DA=98% of the basic The basic salary is entered interactively through the key board.
8. Write a shell script to search given number using binary search.
9. (a) Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.

(b) Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file.

(c) Write a shell script to perform the following string operations:

i. To extract a sub-string from a given string.

ii. To find the length of a given string.

10. Write a shell script which will display Armstrong numbers from given arguments

11. Write a shell script to display factorial value from given argument list

12. Write a C program that simulates ls Command (Use system calls / directory API)

Do the following Shell programs also

1. Write a shell script to check whether a particular user has logged in or not. If he has logged in, also check whether he has eligibility to receive a message or not
2. Write a shell script which will display the username and terminal name who login recently in to the unix system
3. Write a shell script to find no. of files in a directory
4. Write a shell script to check whether a given number is perfect or not
5. Write a menu driven shell script to copy, edit, rename and delete a file
6. Write a shell script for concatenation of two strings
7. Write a shell script which will display Fibonacci series up to a given number of argument
8. Write a shell script to accept student number, name, marks in 5 subjects. Find total, average and grade. Display the result of student and store in a file called stu.data

   Rules: avg>=80 then grade A

   Avg < 80&&Avg>=70 then grade B

   Avg<70&&Avg>=60 then grade C

   Avg<60&&Avg>=50 then grade D

   Avg<50&&Avg>=40 then grade E

   Else grade F

9. Write a shell script to accept empno,empname,basic. Find DA,HRA,TA,PF using following rules. Display empno, empname, basic, DA,HRA,PF,TA,GROSS SAL and NETSAL. Also store all details in a file called emp.data

   Rules: HRA is 18% of basic if basic > 5000 otherwise 550 DA is 35% of basic

PF is 13% of basic

IT is 14% of basic

TA is 10% of basic

10. Write a shell script to demonstrate break and continue statements
11. Write a shell script to display string palindrome from given arguments
12. Write a shell script to display reverse numbers from given argument list
13. 13. Write a shell script which will find maximum file size in the given argument list
14. Write a shell script which will greet you “Good Morning”, ”Good Afternoon”, “Good Evening” and “Good Night” according to current time
15. Write a shell script to sort the elements in a array using bubble sort technique
16. Write a shell script to find largest element in a array
17. Write an awk program to print sum, avg of students marks list
18. Write an awk program to display students pass/fail report
19. Write an awk program to count the no. of vowels in a given file
20. Write an awk program which will find maximum word and its length in the given input File
21. Write a shell script to generate the mathematical tables.
22. Write a shell script to sort elements of given array by using selection sort.
23. Write a shell script to find number of vowels, consonants, numbers, white spaces and special characters in a given string.
24. Write a shell script to search given number using binary search.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21MCT03	Environmental Science	2	0	0	0

Internal Marks: 100

Course Prerequisite: Basic knowledge about sciences up to intermediate or equivalent level.

Course Objectives: The student will be able to

1. Overall understanding of the natural resources
2. Basic understanding of the ecosystem and its diversity
3. Acquaintance on various environmental challenges induced due to unplanned anthropogenic activities
4. An understanding of the environmental impact of developmental activities
5. Awareness on the social issues, environmental legislation and global treaties

Course Outcomes: After going through this course the student will be able to acquire

1. The concepts of the ecosystem and its function in the environment. The need for protecting the producers and consumers in various ecosystems and their role in the food web
2. The natural resources and their importance for the sustenance of the life and recognize the need to conserve the natural resources
3. Various attributes of the pollution and their impacts and measures to reduce or control the pollution along with waste management practices
4. The biodiversity of India and the threats to biodiversity, and conservation practices to protect the biodiversity
5. Social issues both rural and urban environment and the possible means to combat the challenges and environmental assessment stages involved in EIA and the environmental audit.

UNIT–I: (9 Lectures)

MULTIDISCIPLINARY NATURE OF ENVIRONMENTAL STUDIES: Definition, Scope and Importance– Need for Public Awareness. Renewable energy Resources, Solar energy-solar cells, solar batteries, wind energy, wind mills, ocean energy, tidal energy and nonrenewable energy resources: LPG, water gas, producer gas. World food problems, degradation and Soil erosion - overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging and salinity.

UNIT–II: (8 Lectures)

ECOSYSTEMS: Concept of an ecosystem. – Structure, Components and function of an ecosystem – Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Ecological pyramids - Food chains, food webs and Introduction, types, characteristic features, structure and function of the following ecosystem: a. Forest ecosystem. b. Grassland ecosystem c. Desert ecosystem d. Aquatic – River and Lake Ecosystems.

UNIT–III: **(8 Lectures)**

BIODIVERSITY AND ITS CONSERVATION: Introduction, Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India –Value of biodiversity: consumptive use, Productive use, social, ethical and aesthetic values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT–IV: **(9 Lectures)**

ENVIRONMENTAL POLLUTION: Definition, Cause, Effects and Control measures of : a. Air Pollution, b. Water pollution, c. Soil pollution, d. Marine pollution, e. Noise pollution, f. Nuclear hazards.

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes

Disaster management: floods, earthquake, cyclone and landslides.

UNIT–V: **(8 Lectures)**

SOCIAL ISSUES AND THE ENVIRONMENT: From unsustainable to sustainable development – Urban problems related to energy – Water conservation, rain water harvesting and watershed management –Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies Population growth – Impacts on society, variation among nations. Environmental Impact Assessment (EIA) and Environmental Protection Acts.

Text Books:

1. Text book of Environmental Studies for Undergraduate Courses by ErachBharucha for University Grants Commission, Universities Press, 2005.
2. Environmental Studies by Benny Joseph, Tata McGraw Hill Co, New Delhi, 2008.

Reference Books:

1. Environmental Science & Engineering by Dr. A. Ravikrishnan, Hitech Publishing Company Pvt. Ltd. 2013.
2. Perspectives in Environmental Studies, Second edition, AnubhaKoushik and C.P. Koushik, New Age International (P) Limited, Publishers, 2004.

Online References:

1. Environmental Science - Oxford Research Encyclopedia

2. Environmental Science - Museum of Science and Industry
3. Collegesat.du.ac.in/UG/Envinromental%20Studies_ebook.pdf

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST05	Mathematical Foundations of Computer Science	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: An understanding of Mathematics in general is sufficient.
Course Objectives: This course is designed to:

1. To introduce the concepts of mathematical logic.
2. To state the definitions of binary relation, equivalence relation, equivalence class.
3. To introduce the basic counting principles of permutations, combinations.
4. To introduce generating functions and recurrence relations.
5. To use Graph Theory for solving Engineering problems.

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

1. Understand and apply the logic statements and express logical sentences in terms of logical connectives.
2. Understand sets, relations and functions.
3. Demonstrate in practical applications the use of basic counting principles of permutations, combinations, inclusion/exclusion principle.
4. Analyze the various types of recurrences relations and apply the methods to find out their solutions.
5. Solve Science and Engineering problems using Graph Theory.

UNIT–I: (12 Lectures)
Mathematical Logic: Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicate Calculus: Predicates, Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

UNIT–II: (13 Lectures)
Set Theory: Sets: Operations on Sets, Relations: Properties, Operations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering, Hasse Diagrams, Functions: Bijective, Composition, Inverse, Permutation, and Recursive Functions, Lattice and its Properties.

UNIT–III: (13 Lectures)
Combinatorics: Basis of Counting, Permutations, Permutations with Repetitions, Circular and Restricted Permutations, Combinations, Restricted Combinations, Binomial and Multinomial Coefficients and Theorems. The principles of Inclusion – Exclusion. Pigeon hole principles and its application. Algebraic Structures: Algebraic Systems, Properties, Semi Groups and Monoids, Group, Subgroup and Abelian Group, Homomorphism, Isomorphism.

UNIT–IV: **(11 Lectures)**

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

UNIT–V: **(11 Lectures)**

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

Text Books:

1. Discrete Mathematics and its Applications with Combinatorics and Graph Theory- Kenneth H Rosen, 7th Edition, TMH.2007.

2. Elements of DISCRETE MATHEMATICS- A computer Oriented Approach- C L Liu, D P Mohapatra. Third Edition, Tata McGraw Hill.2008.

3. Discrete Mathematics for Computer Scientists and Mathematicians, J.L. Mott, A. Kandel, T.P. Baker, PHI, 2008.

Reference Books:

1. Discrete Mathematical Structures with Applications to Computer Science-J.P. Tremblay and R. Manohar, TMH, 2008.

2. Discrete Mathematics- Richard Johnsonbaugh, 7^{th} Edn., Pearson Education,2009.

3. Discrete Mathematics with Graph Theory- Edgar G. Goodaire, Michael M. Parmenter.2/e, 2002.

4. Discrete and Combinatorial Mathematics - an applied introduction: Ralph.P. Grimald, 5 th edition, Pearson Education, 2004.

Web References:

1. `www.tutorialspoint.com`

2. `www.lecturenotes.in`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST07	Database Management Systems	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: NIL

Course Objectives:

1. Provides students with theoretical knowledge
2. Practical skills in the design, use of databases and database management systems in information technology applications.

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

1. Acquire knowledge in fundamentals of DBMS and identify the differences between traditional file system and DB systems.
2. Understand various DBMS models and how queries are being processed and executed in RDBMS.
3. Analyze DB design methodology and normalization process.
4. Discuss the various transaction and concurrency management techniques
5. Discuss various files indexing techniques.

UNIT–I: **(12 Lectures)**

iNTRODUCTION: Database system, Purpose of Database, view of data, Characteristics (Database Vs File System), Database Users (Actors on Scene, Workers behind the scene) and DBA, Advantages of Database Systems, Database applications. Brief introduction of different Data Models; Concepts of Schema, Instance, and data independence; Three-tier schema architecture for data independence; Database system structure

UNIT–II: **(13 Lectures)**

ENTITY RELATIONSHIP MODEL: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, subclasses, super class, inheritance, specialization, generalization using ER Diagrams.

RELATIONAL MODEL: Introduction to the relational model, concepts of domain, attribute, tuple, relation, the importance of null values, constraints, and their importance.

BASIC SQL: Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update), basic SQL querying (select and project) using where clause, arithmetic & logical operations, Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, views.

UNIT–III: **(11 Lectures)**

SCHEMA REFINEMENT (NORMALIZATION): Introduction to schema Refinement,functional dependencies, Normal forms : 1NF,2NF,3NF,BCNF, properties decompositions, normalization, schema refinement in database design, case studies.

UNIT–IV: (13 Lectures)

TRANSACTION MANAGEMENT AND CONCURRENCY CONTROL: Transaction, properties of transactions, transaction log, and transaction management with SQL using commit rollback and save point. Concurrency control for lost updates, uncommitted data, inconsistent retrievals and the Scheduler. Concurrency control with locking methods: lock granularity, lock types, two phases locking for ensuring serializability, deadlocks, Concurrency control with time stamp ordering: Wait/Die and Wound/Wait Schemes, Database Recovery management : Transaction recovery.

UNIT–V: (11 Lectures)

OVERVIEW OF STORAGES AND INDEXING: Data on External Storage- File Organization and Indexing – Clustered Indexing – Primary and Secondary Indexes, Index Data Structures, Hash-Based Indexing – Tree-Based Indexing, index sequential access method(ISAM).

Text Books:

1. Database Management Systems, 3/e Raghuram Krishnan, Johannes Gehrke, TMH,2002.
2. Database Management System, 6/e Ramez Elmasri, Shamkant B. Navathe, PEA,2010.
3. Database Principles Fundamentals of Design Implementation and Management, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning,2012.
4. Database Management Systems,Rajesh Narang,Second Edition,2018.

Reference Books:

1. Database System Concepts. 5/e Silberschatz, Korth, TMH,2002.
2. Introduction to Database Systems, 8/e C J Date, PEA,2000.
3. The Database book principles & practice using Oracle/MySql Narain Gehani, University Press,2008.

Web References:

1. www.academy.vertabelo.com
2. www.w3schools.com
3. www.codecademy.com

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST08	Design and Analysis of Algorithms	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Mathematics, Data Structures

Course Objectives:

1. Analyze the asymptotic performance of algorithms.
2. Write rigorous correctness proofs for algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations.

Course Outcomes:

1. Identify time, space complexities for different problems.
2. Implement Greedy Method to solve Problems.
3. Implement Dynamic Programming technique to solve Problems.
4. Able how to apply Backtracking and Branch & Bound Techniques in real-time problems.
5. Analyze the pattern-matching algorithms.

UNIT–I: **(12 Lectures)**

Introduction: What is an Algorithm, Pseudo code Conventions Recursive Algorithm, Performance Analysis, Space Complexity, Time Complexity, Asymptotic Notations .

Dived and Conquer: General Method, Binary Search, Finding the Maximum and Minimum, Merge Sort, Quick Sort.

UNIT–II: **(12 Lectures)**

The Greedy Method: The General Method, Knapsack Problem, Minimum-cost Spanning Trees, Prim's Algorithm, Kruskal's Algorithms, Huffman Coding, Optimal Merge Patterns, Single Source Shortest Paths.

UNIT–III: **(12 Lectures)**

Dynamic Programming: The General Method, All Pairs Shortest Paths, Single – Source Shortest paths General Weights, String Edition, 0/1 Knapsack, Travelling Salesperson Problem.

UNIT–IV: **(12 Lectures)**

Backtracking: The General Method, the 8-Queens Problem, Sum of Subsets, Graph Coloring, Hamiltonian Cycles.

Branch and Bound: The Method, The 15-Puzzle problem, Traveling Salesperson.

UNIT–V: **(12 Lectures)**

NP-Hard and NP-Complete Problems: Travelling salesman problem NP complete, NP-Hard Graph Problem (Clique Decision Problem). Pattern Matching Algorithms:

Knuth-Morris-Pratt KMP String Matching Algorithm, Rabin Karp String Matching Algorithm.

Text Books:

1. Fundamentals of computer algorithms E. Horowitz, S. Sahni , UniversityPress Second Edition and year 1997.
2. Introduction to Algorithms Thomas H. CormenLearning Third edition.

Reference Books:

1. The Design and Analysis of Computer Algorithms, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman, First edition and year 2006.
2. Algorithm Design, JonKleinberg, Pearson First edition 2008.

Web References:

1. `https://nptel.ac.in/courses/106/106/106106131/`
2. `https://www.tutorialspoint.com/design_and_analysis_of_algorithms/index.htm`
3. `https://slideplayer.com/slide/5877267/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST01	Java Programming	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: C Programming

Course Objectives:

1. To understand Object Oriented Programming concepts and basic characteristics of Java
2. To understand the principles of packages, inheritance and interfaces
3. To Implement exceptions and use I/O streams
4. To design and build simple Graphical User Interface application.

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

1. Implement OOPS concepts in Java programs
2. Develop Java programs with the concepts of inheritance and interfaces
3. Design a Java applications using exceptions and I/O streams
4. Develop Java applications using Multithreading
5. Design interactive Java application using swings

UNIT–I: **(9 Lectures)**

OOPS-Fundamentals: Object Oriented Programming concepts, Benefits of OOP, and Applications of OOP.

Java history, java features, Characteristics of Java, Java Source File –Structure, Compilation, Java keywords, Data Types, Variables and Arrays, Operators, Control Statements.

UNIT–II: **(9 Lectures)**

OOPS-class, object and Inheritance:

Class fundamentals, declaring objects, assigning object reference variables, introducing methods, overloading methods ,constructors, this keyword, static and final keywords, nested and inner classes.

Inheritance and its types, super keyword, method overriding, defining a package, creation of packages, importing packages, Access specifiers, Abstract classes, Interfaces ,Strings.

UNIT–III: **(9 Lectures)**

Exception Handling& Streams: Exception handling fundamentals, exception types, uncaught exceptions, using try and catch, multiple catch clauses, throw, throws, finally, creating your own exceptions Stream Classes: Byte Streams- InputStream, OutputStream, FileInputStream, FileOutputStream, Character Streams- Reader, Writer, FileReader, FileWriter.

UNIT–IV: **(9 Lectures)**

Multithread Programming & Applets: Defining thread, Creating a thread, creating multiple threads, thread life cycle, thread priorities- thread synchronization-Inter-thread communication, daemon threads.

Java Applets- Applet class, Applet structure, An Example Applet Program, Applet Life Cycle, Passing Parameters to Applets.

UNIT–V: **(10 Lectures)**

Event Handling & Graphics Programming:

Event Handling: The delegation event model- Events, Event Sources, Event Listeners. Event Classes, Event Listener Interfaces, Using the delegation handling the different events.

Graphics programming: Graphics programming, java.awt package, AWT event hierarchy, Frame , Components, Container class, Layouts. Swing Components-Text Fields, Text Areas - Buttons- Check Boxes – Radio Buttons - Lists- choices-Scrollbars - Windows -Menus - Dialog Boxes.

Text Books:

1. Java The complete reference, Herbert Schildt , 8th Edition, McGraw Hill Education, 2011.
2. Core Java Volume –I Fundamentals, Cay S. Horstmann, Gary cornell, 9th Edition, Prentice Hall, 2013.
3. Paul J. Dietel and Dr.Harvey M. Deitel, “Java How to Program”, 9th Edition, Prentice- Hall, Pearson Education, 2011.
4. Timothy Budd, “Understanding Object Oriented Programming with Java “, Updated edition, Pearson Education, 2013.
5. Programming with JAVA, E.Balaguruswamy, 5th Edition, McGraw Hill Education, 2014.

Reference Books:

1. Java 2 Black book, Steven Holzner, Dreamtech press, 2011.
2. The JAVA programming language, K. Arnold and J. Gosling, Third edition, Pearson Education, 2000.
3. An introduction to Object-oriented programming with Java, C. Thomas Wu, Fourth Edition, Tata McGraw-Hill Publishing company Ltd., 2006.

Web References:

1. `www.javaworld.com`
2. `www.ibm.com`
3. `www.java.sun.com`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21SST01	Developing Soft Skills and Personality	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: The students should have basic knowledge of soft skills.

Course Objectives:

1. To encourage the all round development of students by focusing on soft skills.
2. To make the engineering students aware of the importance, the role and the content of soft skills through instruction, knowledge acquisition, demonstration and practice.
3. To develop and nuture the soft skills of the students through individual and group activities.
4. To expose students to right attitudinal and behavioral aspects and to build the same through activities.

Course Outcomes:

1. To enhance soft skills and significance of soft skills in working environment.
2. To develop interpersonal l skills.
3. To become more effective individual through goal/target setting and self motivation.
4. To develop techniques to manage time.
5. To become self confident individual by mastering interpersonal skills, team management skills and leadership skills.

UNIT–I: **(9 Lectures)**

SOFT SKILLS

Introduction- What are soft skills?-Importance of soft skills-Selling your soft skills-Attributes regarded as soft skills-Soft skills-social-soft skills-Thinking soft skills-Negotiating- Exhibiting your soft skills-identifying your soft skills-improving your soft skills-will formal training enhance your soft skills- soft skills training- train yourself-Top 60 soft skills- practicing soft skills- measuring attitude

UNIT–II: **(9 Lectures)**

INTERPERSONAL SKILLS

Significance of interpersonal skills-factors that prevent building and maintaining positive relationships-advantages of positive relationships-disadvantages of negative relationships.

UNIT–III: **(9 Lectures)**

GOAL SETTING

Introduction-Goal -significance of goal setting-goal setting process-qualities of effective goals-barriers to reach goals

UNIT–IV: **(9 Lectures)**

TIME MANAGEMENT

Introduction-What are time wasters?-ABC technique-The Pickle Jar Theory of time management-First thing first-Tips to manage time effectively.

UNIT–V: **(9 Lectures)**

TEAM BUILDING AND TEAM WORK

Introduction-Meaning-Aspects of team building-skills needed for team work- A model of team building-Team Vs Group- Characteristics of effective team- Role of a team leader- Role of team members-nine persons a successful team should have-inter-group collaboration- Factors shaping inter-group collaboration.

Reference Books:

1. Soft Skills for Success: G.R.K Murthy
2. Soft Skills:Dr. K. Alex
3. Soft Skills: Dr. B.V. Pattabhi Ram
4. Life Skills: M.L.Suresh Kumar
5. Austin, Ten Keys to Time Management:

Web References:

1. http://www.coachville.com/tl/thomasleonard/karla/formsCD/161keystotimemanagement.
2. œTimeMangamenthttp://www.timethoughts.com/time-
3. management.htm.Studocu.com/in/docur
4. https://daringtolivefully.comself-awareness
5. œImportanceofTeamandTeamwork.http://www.managementstudyguide.com/importance-of-team.htm.
6. Pandita,R.PrinciplesofEffectiveTeamwork.2011.http://www.buzzle.com/articles/principles-of-efffective-teamwork.html.
7. TeamworkPrinciples.http://www.the-happy-manager.com/teamwork-principles.html.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL06	R Programming Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: NILL

Course Objectives: This course provides a solid undergraduate foundation in both probability theory and mathematical statistics and at the same time provides an indication of the relevance and importance of the theory in solving practical problems in the real world.

Course Outcomes:

1. Understand the fundamental syntax of R through readings, practice exercises, demonstrations, and writing R code.
2. Apply critical programming language concepts such as data types, iteration, control structures, functions, and boolean operators by writing R programs and through examples
3. Apply the Import a variety of data formats into R using RStudio
4. Able to Prepare or tidy data's for in preparation for analysis
5. Analyze a data set in R and present findings using the appropriate R packages

List of Experiments:

1. History of R.
2. Installing R and Packages in R.
3. Write a program to illustrate Data types in R.
4. Write a program to illustrate creating and manipulating a Vector in R.
5. Write a program to illustrate creates a matrix and manipulating matrix in R.
6. Write a program to illustrate performing Arithmetic of matrices.
7. Write a program to Creating and operations on Factors in R.
8. Write a program to illustrate various Data frame Operations.
9. Write a program to illustrate List & it's Operations in R.
10. Write a program to illustrate if-else-else-if in R.
11. Write a program to illustrate While and For loops in R.
12. Write a program to illustrate Compare and Matrices and Compare Vectors.
13. Write a program to illustrate to create Graphs and usage of plot( ) function in R.

14. Write a program to illustrate Customizing and Saving to graphs in R.

Text Book

1. The Book of R-A first course in programming and statistics by Tilman M.Davies.
2. R Programming-Robin Evans,version: November 5, 2014.
3. R for Beginners by Emmanuel Paradis.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL05	Database Management Systems Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: NILL

Course Objectives:

1. Understand, analyze and apply SQL commands like DDL,DML,DCL to perform different Database operations
2. Understand and practice PL/SQL block, control statements and cursors.
3. Develop PL/SQL programs using, functions, procedures, packages and Triggers.

Course Outcomes:

1. Know about SQL DDL,DML,DCL,TCL commands
2. Know how to write SQL Quires using set operators
3. Know about how to implement PL/SQL programs using conditional ,loops statements
4. Know about implementing of triggers, cursors and exceptions
5. Know about implementing procedures, functions and packages

PROGRAMS LIST

1. Creation, altering and droping of tables and inserting rows into a table (use constraints while creating tables).
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions, string functions and date functions
5. (a) Creation of simple PL/SQL program which includes declaration section, executable section and exception –Handling section

   (b) Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISEAPPLICATION ERROR.

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

PL/SQL

9. Usage of triggers to perform the operation on single and multiple Table.
10. Write a PL/SQL block illustrating packages.
11. Write a PL/SQL code using CURSOR.
12. PL/SQL procedures for data validation.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL01	Java Programming Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: C Programming and Object-Oriented Concepts.
Course Objectives: The objective of this lab is

1. To build software development skills using java programming for real-world applications.
2. To understand and apply the concepts of classes, packages, interfaces, arraylist, exception handling and file processing.
3. To develop applications using JDBC programming and event handling.

Course Outcomes:

1. Develop and implement Java programs for simple applications that make use of classes, packages and interfaces.
2. Develop and implement Java programs with arraylist, exception handling and multithreading .
3. Design applications using file processing, JDBC programming and event handling.

List of Experiments:
Exercise 1: (Basics)

1. Write a java program that display the roots of a quadratic equation ax2+bx=0. Calculate the discriminate D and basing on value of D, describe the nature of root.
2. Write a java Program to check the number is Prime or not.

Exercise 2: (Operations, Expressions, Control-flow, Strings)

1. Write a java program that display the roots of a quadratic equation ax2+bx=0. Calculate the discriminate D and basing on value of D, describe the nature of root.
2. Write a JAVA program to sort for an element in a given list of elements using bubble sort.
3. Write a JAVA program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome.

Exercise 3: (Class, Objects, Constructor)

1. Write a program to create a class Student with data ‘name, city and age’ along with method printData to display the data. Create the two objects s1 ,s2 to declare and access the values.

2. Write a program in JAVA to demonstrate the method and constructor overloading.
3. Write a program in JAVA to create a class Bird also declares the different parameterized constructor to display the name of Birds.

Exercise 4: (Inheritance, Method Overriding)

1. Write a JAVA program to implement Single Inheritance
2. Write a JAVA program to implement multi level Inheritance
3. Write a java program for abstract class to find areas of different shapes
4. Write a JAVA program that implements Runtime polymorphism(Method Overriding) problem

Exercise 5: (Array List & Exception)

1. Write a program to perform string operations using ArrayList. Write functions for the following
   a. Append
   b. Insert
   c. Search
   d. List all string starts with given letter.
2. Write a JAVA program that describes exception handling mechanism
3. Write a JAVA program Illustrating Multiple catch clauses

Exercise 6: (User defined Exception)

1. Write a JAVA program for creation of Illustrating throw.gram that implement Queue (its operations) using arrays.
2. Write a JAVA program for creation of Illustrating finally.
3. Write a JAVA program for creation of Java Built-in Exceptions.
4. Write a JAVA program for creation of User Defined Exception.

Exercise 7: (Threads)

1. Write a JAVA program that creates threads by extending Thread class .First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds ,(Repeat the same by implementing Runnable).
2. Write a java program for to solve producer consumer problem in which a producer produce a value and consumer consume the value before producer generate the next value.

Exercise 8: (File Handling)

1. Write a java program that displays the number of characters, lines and words in a text file.

Exercise 9: (JDBC & Packages)

1. Write a java program that connects to a database using JDBC of the following
   a. add
   b. Delete
   c. Modify
   d. Retrieve operations.
2. Write a java program to create a package called employee and implement this package out of the package.

Exercise 10: (Applet)

1. Write a JAVA program to paint like paint brush in applet.
2. Write a JAVA program to display analog clock using Applet.
3. Develop an Applet that receives an integer in one text field & compute its factorial value & returns it in another text filed when the button "Compute" is clicked.

Exercise 11: (Event Handling)

1. Write a JAVA program that display the x and y position of the cursor movement using Mouse.
2. Write a JAVA program that identifies key-up key-down event user entering text in a Applet.

Exercise 12: (Swings)

1. Write a JAVA program to build a Calculator in Swings.
2. Write a JAVA program to display the digital watch in swing tutorial.

Exercise 13: (Swings - Continued)

1. Write a JAVA program that to create a single ball bouncing inside a JPanel.
2. Write a JAVA program JTree as displaying a real tree upside down

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CIS02	Android Application Development	1	0	2	2

Internal Marks: 15 External Marks: 35

Course Prerequisite: NILL

Course Objectives: To enlighten the student with knowledge base in Android Applications Development.

Course Outcomes: After completion of the course, students would be able to:

1. Demonstrate various components of Android Framework.
2. Develop user Interfaces for the Android Application.
3. Develop Android Applications using Android API and Services.
4. Develop Android Applications which access data from Internet.

List of Experiment

1. Create Hello World Android App using Android Studio and explain each step in detail.
2. Create an Activity that receive name form the user and displays Hello Name to the user using Android Studio.
3. Create an Activity that demonstrates the Life Cycle of an Activity.
4. Create an Android Application which receives URL form the user and open appropriate page in the system browser with the help of Implicit Intents using Android Studio.
5. Create an Android App which receives name form the user and displays welcome name in Second Activity.
6. Create Login Screen Application which shows Home screen if Login success otherwise displays error message using Android Studio.
7. Write an Android application program that demonstrate the use of
   (a) RelativeLayout.
   (b) LinearLayout.
   (c) GridLayout.
   (d) TableLayout.
8. Write an Android application program that demonstrates the use ImageView.
9. Write an Android application program that demonstrates the use of ListView and ArrayAdapter.
10. Write an Android application program that demonstrates the use AsyncTask.

11. Write an Android application program that demonstrates Notifications.
12. Write an Android application program that demonstrates Shared Preferences.
13. Write an Android application program that connect to the internet, gets JSON data and displays the result in UI by parsing JSON data.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST09	Computer Networks	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Fundamentals of Computer.

Course Objectives:

1. To introduce the fundamental concepts of computer networking.
2. To familiarize with networking concepts to work on various Protocols of ISO-OSI and TCP/IP.

Course Outcomes:

1. Compare protocol models (OSI, TCP/IP) and select suitable protocol for network design.
2. Design a network by deciding relevant multiplexing and switching technique to improve performance of the network.
3. Apply flow control, error control techniques and protocols to verify the correctness of data in the communicated network.
4. Apply routing and congestion control algorithms to deliver data packets across the networks.
5. Use communication protocols like IP, TCP, UDP, DNS, HTTP, FTP across the Internet

UNIT–I: **(8 Lectures)**

Introduction: Introduction - components of data communication, data flow, network topologies, categories of networks-LAN, MAN, WAN, ISO-OSI model, TCP/IP.

UNIT–II: **(8 Lectures)**

Physical Layer: Multiplexing- frequency division multiplexing, synchronous time division multiplexing, statistical time division multiplexing, Introduction to switching - circuit switched networks, datagram networks, virtual circuit networks

UNIT–III: **(11 Lectures)**

Data Link Layer: Design issues, framing, error control, error detection and correction, CRC, checksum, hamming code. Elementary data link layer protocols-simplex protocol, simplex stop and wait, simplex protocol for noisy channel. Sliding window

protocol: one bit, Go back N, selective repeat, data link layer in HDLC, PPP. Medium Access Control Sub Layer ALOHA, CSMA, CSMA/CD, IEEE standards-standard Ethernet, wireless LAN, bridges..

UNIT–IV: **(9 Lectures)**

Network Layer: Routing algorithms- shortest path routing, distance vector, link state routing, and hierarchical routing. Congestion control algorithms-congestion control in virtual circuit subnets, datagram subnet, leaky bucket, token bucket. The network layer in the Internet: The IP protocol, IPAddresses-IPv4, IPv6.

UNIT–V: (12 Lectures)

Transport Layer: Transmission Control Protocol (TCP)- services, segment header, connection establishment, termination, transmission policy, congestion control. User Datagram Protocol (UDP)- header format.

Application Layer: The Domain Name System (DNS), electronic mailarchitecture, SMTP, FTP, HTTP.

Text Books:

1. Andrew S Tanenbaum, "Computer Networks", 4 th edition, Pearson. 2003.
2. Behrouz A Forouzan, "Data Communications and Networking", 5 th edition, TMH, 2007.

References:

1. S. Keshav, "An Engineering Approach to Computer Networks", 2 nd edition, Pearson Education, 1997.
2. W.A. Shay, Thomson, "Understanding Communications and Networks", 3 nd edition, Cengage Learning, 2001.

Web References:

1. http://www.cs.ccsu.edu/~stan/classes/CS490/Slides/Networks4-Ch4-4.pdf
2. http://ecourses.vtu.ac.in/nptel/courses/Webcourse-contents/IIT-MADRAS/ComputerNetworks/pdf/

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CIT02	Web Technologies	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisites: Object Oriented Programming

Course Objectives:

1. To develop the static web pages using HTML and CSS.
2. To enables the students to identify the fundamental concepts for developing web application using PHP language for server side scripting.
3. To analyze how data can be transported using XML.
4. To develop a web applications with server side programming using java servlets.
5. To develop a web applications with server side programming using JSP.

Course Outcomes:

1. Summarize the basic tags and properties in HTML, XHTML and CSS.
2. Create web pages using .client side scripting, validating of forms and XML.
3. Identify the role of server side scripting using PHP programming
4. Design dynamic web application using server side programming with java servlets.
5. Contraston how to connect and retrieve data through web page from database usingJDBC.

UNIT–I: **(9 Lectures)**

HTML Common tags- List, Tables, images, forms, Frames, Links and Navigation, **CSS:** Introduction, CSS Properties, Controlling Fonts, Text Formatting, Pseudo classes, Selectors.

UNIT–II: **(9 Lectures)**

Client side Scripting: Introduction to Javascript: Javascript language – declaring variables, scope of variables, functions, event handlers (onclick, onsubmit etc.), Document Object Model, Form validation.

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model.

UNIT–III: **(9 Lectures)**

Introduction to PHP: Creating PHP script, Running PHP script, Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies.

UNIT–IV: **(9 Lectures)**

A: Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a servlet, deploying a servlet,

B: The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions.

UNIT–V: **(9 Lectures)**

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, JSP application design with MVC, Declaring variables and methods, sharing data between JSP pages, Requests and users passing control and data between pages, Sharing sessions and application data.

JDBC connectivity in JSP: Data base programming using JDBC, Studying javax.sql.* package, Accessing a database from a JSP page, Application specific database actions.

Text Books:

1. Web Technologies, Uttam K Roy, Oxford UniversityPress
2. The Complete Reference PHP – Steven Holzner, TataMcGraw-Hill

Reference Books:

1. JWeb Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
2. Java Server Pages –Hans Bergsten, SPDO'Reilly
3. Java Script, D. Flanagan,O'Reilly,SPD.
4. Beginning Web Programming-Jon DuckettWROX.
5. Programming World Wide Web, R. W. Sebesta, Fourth Edition,Pearson.
6. Internet and World Wide Web – How to program, Dietel and Nieto,Pearson.

Web References:

1. `https://www.w3schools.com/html/`
2. `https://www.javatpoint.com/servlet-tutorial`
3. `http://nptel.ac.in/courses/106105084/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST10	Operating Systems	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Computer System fundamentals

Course Objectives:

1. Analyze the tradeoffs inherent in operating system design.
2. Summarize the various approaches to solving the problem of mutual exclusion in an operating system.
3. Understand the principles of Deadlocks.
4. Evaluate the trade-offs in terms of memory size (main memory, cache memory, auxiliary memory) and processor speed.
5. Demonstrate disk storage strategies, file strategies

Course Outcomes:

1. Describe the role, functions and structures of operating systems and demonstrate the concept of process.
2. Analyze the performance of CPU scheduling algorithms and develop software/hardware-based solutions for critical section problems
3. Design deadlock, prevention and avoidance algorithms.
4. compare different memory management schemes and apply page replacement algorithms in virtual memory.
5. Outline the concept of files, directory structures and and analyze the performance of different disk scheduling algorithms

UNIT–I: **(9 Lectures)**

Computer System and Operating System Overview: Overview of Computer System hardware, Operating System Objectives and functions, Evaluation of operating System, Operating System Services, System Calls.

Process Management: Process Description, Process Control, Process States, Co-operating Processes , Inter-process Communication.

UNIT–II: **(13 Lectures)**

CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Threads Overview, Threading issues.

Synchronization: Background, The Critical-Section Problem, Peterson solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors.

UNIT–III: **(11 Lectures)**

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

UNIT–IV: (13 Lectures)

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

Virtual Memory Management: Background, Demand Paging, Page Replacement, allocation of frames, Thrashing.

UNIT–V: (12 Lectures)

File system Interface: The concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection.

File System implementation: File system structure, allocation methods, free space management Mass storage structure, overview of Mass-storage structure, Disk scheduling.

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne “Operating System Concepts”, 9th Edition, John Wiley and Sons Inc.
2. William Stallings, 7th Edition, “ Operating Systems – Internals and Design Principles”, Prentice Hall, 2011.

References:

1. Andrew S. Tanenbaum “Modern Operating Systems”, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata Mc Graw Hill Education”, 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhere, Second Edition, Tata Mc Graw-Hill Education, 2007.

Web References:

1. `https://nptel.ac.in/courses/106/106/106106144/`
2. `https://www.tutorialspoint.com/operating_system`
3. `https://www.youtube.com/playlist?list=PLEJxKK7AcSEGPOCFtQTJhOElU44J_JAun`
4. `https://www.pdf-archive.com/2016/12/25/operating-system-concepts-9thedition/operating-system-concepts-9th-edition.pdf.`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL07	Computer Network and Operating System Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Outcomes:

1. Implement data link layer framing methods.
2. Experiment with error handling methods.
3. Make use of various routing algorithms for effective data transmission.
4. Illustrate various process scheduling algorithms.
5. Explain Page replacement algorithms and file allocation strategies.

Part-A:

1. Implement the data link layer farming methods such as character, character stuffing and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials -CRC 12, CRC 16 and CRC CCIP.
3. Implement Dijkstra's algorithm to compute the Shortest path through a graph.
4. Implementation of distance vector routing algorithm.
5. Write a program for congestion control using Leaky bucket algorithm.
6. Implementation of RSA algorithm.

Part-B:

1. Simulate the following CPU scheduling algorithms
   (a) Round Robin
   (b) SJF
   (c) FCFS
   (d) Priority
2. Implementation of fork (), wait (), exec() and exit () System calls
3. Simulate the following.
   (a) Multiprogramming with a fixed number of tasks (MFT)
   (b) Multiprogramming with a variable number of tasks (MVT)
4. Simulate Bankers Algorithm for Dead Lock Avoidance

5. Simulate the following page replacement algorithms.

   (a) FIFO

   (b) LRU

   (c) LFU

6. Simulate the following File allocation strategies

   (a) Sequenced

   (b) Indexed

   (c) Linked

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CIL02	Web Technologies Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Outcomes:

1. Create static web pages using HTML and CSS.
2. Develop JavaScript code for data validation.
3. Integrate frontend and backend technologies in client-server systems.
4. Design dynamic web applications using PHP and JSP.
5. Demonstrate database connectivity for developing web applications.

The students have to choose one of the following project and do the all 12 experiments related to that project.

1. Training and placementcell.
2. School EducationSystem.
3. University ManagementSystem.
4. Hospital ManagementSystem.

The following are the experiments related to Training and Placement cell project. For the remaining projects, the concern lab instructor has to decide the experiments according to the websites given as examples.

List of Experiments:

Experiment 1:

Design the following static web pages required for a Training and placement cell web site.

1) Home Page 2) Login Page 3) Registration page

Experiment 2:

4) Company Details Page 5) Alumni Details Page 6) Placement Staff Details Page

Experimen 3:

7) Student personal Info Page 8) Student Academic Info page 9) SemesterWise Percentage & their Aggregate page

Experiment 4:

Validate login page and registration page using regular expressions.

Experiment 5:

Apply different font styles, font families, font colors and other formatting styles to the above static web pages.

Experiment 6:

Install wamp server and tomcat server, access above developed static web pages using these servers.

Experiment 7:

Write a servlet/PHP to connect to the database, Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration.

Experiment 8:

Write a JSP/PHP to connect to the database, Insert the details of the student academic information with student academic info page.

Experiment 9:

User Authentication:

Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a servlet for doing the following.

Create a Cookie and add these four user id's and passwords to this Cookie.

Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in thecookies.

If he is a valid user (i.e., user-name and password match) you should welcome him by name (user-name) else you should display "You are not an authenticated user ". Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters()method.

Experiment 10: Write a JSP which does the followingjob:

Authenticate the user when he submits the login form using the user name and password from the database.

Experiment 11:

write a JSP to insert the student's semester wise percentages and calculate aggregate and insert into database.

Experiment 12:

write a JSP to search the students according to their aggregate and produce sorted list or according to their Enroll number.

Course Code	Course Name	Course Structure			
		L	T	P	C
	Professional Ethics & Human Values	2	0	0	0

Internal Marks: 30 External Marks: 70

Course Objectives:

1. To introduce the students to the Human values and help them to lead a peaceful life in the society by contributing to peace and safety in the society.
2. To help the students to know about the history of ethics and importance of social experimentation
3. To specify the students about the importance of their responsibility towards safety and risk as Engineers.
4. To specify the students about the importance of their responsibility as Engineers.
5. To help the student explore the ethical values globally.

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

1. To learn about the different Human values to be maintained by all the people.
2. To learn about the history of ethics and the importance of ethics for professionals and application of ethics in social experimentation.
3. To learn about the responsibilities of engineers for safety and risk.
4. To learn about the responsibilities and rights of engineers.
5. To learn about global work environment with respect to ethics.

UNIT–I: Human Values (9 Lectures)

Human Values: Morals, Values and Ethics – Integrity – Work Ethics – Service Learning – Civic Virtue – Respect for others – Living Peacefully – Caring – Sharing –Honesty –Courage – Value time – Co-operation – Commitment – Empathy – Self-confidence – Spirituality- Character.

UNIT–II: Engineering Ethics and Social Experimentation (12 Lectures)

Engineering Ethics: The History of Ethics-Purposes for Engineering Ethics-Engineering Ethics - Consensus and Controversy –Professional and Professionalism –Professional Roles to be played by an Engineer –Self Interest, Customs and Religion-Uses of Ethical Theories-Professional Ethics-Engineering and Ethics-Kohlberg's Theory – Gilligan's Argument –Heinz's Dilemma. Engineering as Social Experimentation: Comparison with Standard Experiments –Knowledge gained – Conscientiousness – Learning from the Past – Engineers as Managers, Consultants, and Leaders – Role of Codes – Codes and Experimental Nature of Engineering.

UNIT–III: Engineers' Responsibility for Safety and Risk (9 Lectures)

Engineers' Responsibility for Safety and Risk: Safety and Risk, Concept of Safety – Types of Risks – Voluntary v/s Involuntary Risk- Short term v/s Long term Consequences- Expected Probability- Reversible Effects- Threshold Levels for Risk- Delayed v/s Immediate Risk- Safety and the Engineer – Designing for Safety – Risk Benefit Analysis-Accidents.

UNIT–IV: Engineers' Responsibilities and Rights (12 Lectures)

Engineers' Responsibilities and Rights: Collegiality-Techniques for Achieving Collegiality –Two Senses of Loyalty - obligations of Loyalty-misguided Loyalty –professionalism and Loyalty - Professional Rights –Professional Responsibilities –confidential and proprietary information-Conflict of Interest-solving conflict problems – Self interest, Customs and Religion- Ethical egoism-Collective bargaining Confidentiality-Acceptance of Bribes/Gifts-when is a Gift and a Bribe examples of Gifts v/s Bribes-problem solving-interests in other companies Occupational Crimes-industrial espionage-price fixing-endangering lives Whistle Blowing-types of whistle blowing-when should it be attempted preventing whistle blowing.

UNIT–V:Global Issues (10 Lectures)

Global Issues: Globalization- Cross-culture Issues-Environmental Ethics-Computer Ethics computers as the instrument of Unethical behaviour-computers as the object of Unethical Acts-autonomous computers-computer codes of Ethics Weapons Development-Ethics and Research-Analysing Ethical Problems in Research-Intellectual Property Rights.

Text Books:

1. "Engineering Ethics and Human Values" by M.Govindarajan, S.Natarajan and V.S. Senthil Kumar-PHI Learning Pvt. Ltd-2009.

2. "Professional Ethics and Morals" by Prof.A.R.Aryasri, Dharanikota Suyodhana-Maruthi Publications.

3. "Professional Ethics and Human Values" by A.Alavudeen, R.Kalil Rahman and M. Jayakumaran- Laxmi Publications.

4. "Professional Ethics and Human Values" by Prof.

Reference Books

1. "Indian Culture, Values and Professional Ethics" by PSR Murthy, BS Publication.

2. "Ethics in Engineering" by Mike W. Martin and Roland Schinzinger – Tata McGraw-Hill – 2003.

3. "Engineering Ethics" by Harris, Pritchard and Rabins, CENGAGE Learning, India Edition, 2009.

Web References:

1. crescent.education/wp-content/.../12/Crescent-human-values-professional-ethics.pdf
2. https://www.crectirupati.com/.../HVPE-MBA-K%20YAMUNA-LECTURE%20NOTES...
3. https://nptel.ac.in/courses/109104068/30
4. https://lecturenotes.in/subject/576/professional-ethics-and-human-values-pehv
5. https://onlinecourses.nptel.ac.in/noc18_mg25

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST12	Cryptography and Network Security	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

1. The main objective of this course is to teach students to understand and how to address various software security problems in a secure and controlled environment.

2. During this course the students will gain knowledge in various kinds of software security problems, and techniques that could be used to protect the software from security threats.

Course Outcomes:

1. Evaluate the use of encryption algorithms for achieving data confidentiality.
2. Apply Secure hash functions for attaining data integrity.
3. Explain IP security issues and protection mechanisms to secure Emails.
4. Demonstrate the techniques of web security.
5. Examine the threats and remedial measures for system security.

UNIT–I: **(10 Lectures)**

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.

Symmetric Key Cryptography: Symmetric Encryption Principles, Symmetric Encryption Algorithms (DES,Triple DES and AES), Cipher Block Modes of Operations.

UNIT–II: **(8 Lectures)**

Public-Key Cryptography and Message Authentication: Approaches to Message Authentication, Secure Hash Functions, Message Authentication Codes, Public-Key Cryptography Principles, Public-Key Cryptography Algorithms, Digital Signatures.

UNIT–III: **(9 Lectures)**

Electronic Mail Security: Pretty Good Privacy (PGP) and S/MIME.

IP Security: IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange.

UNIT–IV: **(9 Lectures)**

Web Security: Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT–V: **(9 Lectures)**

Intruders and Malicious Software: Intruders, Intrusion Detection, Viruses and Related Threats, Trusted System.
Firewalls: Firewalls-Characteristics, Types of Firewalls, Placement of Firewalls, Firewall Configuration.

Text Books:

1. William Stallings, Principles and Practice "Cryptography and Network Security" ,6th Edition, Pearson Education,2011.
2. William Stallings, Network Security Essentials (Applications and Standards), Pearson Education, 2010.
3. Chwan Hwa Wu, J.David Irwin,Introduction to Computer Networks & Cyber Security CRC Press, 2013.

References:

1. Fundamentals of Network Security by Eric Maiwald (Dreamtech press), 2015.
2. Principles of Information Security, Withman , Thomson, 2017.
3. Introduction to Cryptography, Buchmann, Springer, 2010.

Web References:

1. `https://onlinecourses.nptel.ac.in/noc18_cs07/preview`
2. `https://www.coursera.org/learn/cryptography`
3. `https://www.coursera.org/specializations/computer-network-security`
4. `https://www.youtube.com/watch?v=Q-HugPvA7GQ&list=PL71FE85723FD414D7`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21AMT01	Machine Learning	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Objectives:

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

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

1. Explain the fundamental usage of the concept Machine Learning system
2. Demonstrate on various regression Technique
3. Analyze the Ensemble Learning Methods
4. Illustrate the Clustering Techniques and Dimensionality Reduction Models in Machine Learning.
5. Discuss the Neural Network Models and Fundamentals concepts of Deep Learning

UNIT–I: **(9 Lectures)**

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

UNIT–II: **(10 Lectures)**

Supervised Learning(Regression/Classification):Basic Methods: Distance based Methods, Nearest Neighbours, Decision Trees, Naive Bayes,

Linear Models: Linear Regression, Logistic Regression, Generalized Linear Models, Support Vector Machines, Binary Classification: Multiclass/Structured outputs, MNIST, Ranking.

UNIT–III: **(11 Lectures)**

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

UNIT–IV: **(8 Lectures)**

Unsupervised Learning Techniques: Clustering, K-Means, Limits of K-Means, Using Clustering for Image Segmentation, Using Clustering for Preprocessing, Using Clustering for Semi-Supervised Learning, DBSCAN, Gaussian Mixtures.

Dimensionality Reduction: The Curse of Dimensionality, Main Approaches for Dimensionality Reduction, PCA, Using Scikit-Learn, Randomized PCA, Kernel PCA.

UNIT–V: **(12 Lectures)**

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

Text Books:

1. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition, O'Reilly Publications, 2019
2. Data Science and Machine Learning Mathematical and Statistical Methods,Dirk P. Kroese, Zdravko I. Botev, Thomas Taimre, Radislav Vaisman,2020

Reference Books

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

Web References:

1. `https://www.geeksforgeeks.org/machine-learning/`
2. `https://nptel.ac.in/courses/106106139`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CST14	Automata Theory & Compiler Design	3	0	0	3

Internal Marks: 30 External Marks: 70

Course Prerequisite: Programming Languages, computer architecture, data structures and simple graph algorithms, logic or algebra.

Course Objectives:

1. To understand the various phases in the design of a compiler.
2. To understand the design of top-down and bottom-up parsers.
3. To understand syntax directed translation schemes.
4. To introduce lex and yacc tools.
5. To learn to develop algorithms to generate code for a target machine.

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

1. Ability to design, develop, and implement a compiler for any language.
2. Able to use lex and yacc tools for developing a scanner and a parser.
3. Able to design and implement LL and LR parsers.
4. Able to design algorithms to perform code optimization in order to improve the performance of a program in terms of space and time complexity.
5. Ability to design algorithms to generate machine code

UNIT–I: **(9 Lectures)**

Fundamentals: Introduction, Basic Concepts, Introduction to Formal Proofs, Inductive proofs, Introduction to Defining Language, Kleen Closures, Arithmetic Expressions, Graphs, Trees, Finite State Machine, Acceptance of Strings and Languages, Deterministic Finite Automata, Non-Deterministic Finite Automata.

UNIT–II: **(9 Lectures)**

Finite Automata: Introduction, Significance of Nondeterministic Finite Automation, NFA with ϵ – Transactions, Conversions and Equivalence , NFA to DFA Conversion, Minimization of FSM, Equivalence between Two FSMs.

UNIT–III: **(9 Lectures)**

Grammar Formalism: Introduction, Regular Grammar, Equivalence between Regular Grammar and FA, Conversation of Right – Linear Grammar to Left Linear Grammar, Context Free Grammar. Grammar Formalism: Introduction, Regular Grammar, Equivalence between Regular Grammar and FA, Conversation of Right – Linear Grammar to Left Linear Grammar, Context Free Grammar.

UNIT–IV: **(8 Lectures)**

Lexical Analysis : Introduction , Role of Lexical Analysis, Lexical Analysis Vs Parsing, Token, Patterns and Lexeme, Lexical Errors, Input Buffering, Regular Expressions

UNIT–V: (9 Lectures)

Syntax Analysis : Introduction, Role of Parser, Context Free Grammar(CFG), Classification of Parsing Techniques, Top Down Parsing, Recursive Descent Parser

Text Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman (2007), Introduction to Automata Theory Languages andComputation, 3rdedition, Pearson Education, India.

2. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman, Pearson.

REFERENCE BOOKS:

1. K. L. P Mishra, N. Chandrashekaran (2003), Theory of Computer Science-Automata Languages and Computation, 2nd edition, Prentice Hall of India, India.

2. Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning.

3. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.

4. Formal Languages & Automata Theory, A.A.Puntambekar, First Edition, Technical Publications

5. Compiler Design, A.A.Puntambekar, First Edition, Technical Publications

Web References:

1. `https://archive.nptel.ac.in/courses/111/103/111103016/`

2. `https://archive.nptel.ac.in/courses/106/104/106104123/`

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CIL03	Network Programming and Security Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Objectives:

1. Explain the objectives of information security.
2. Explain the importance and application of each of confidentiality, integrity, authentication and availability.
3. Understand various cryptographic algorithms.

Course Outcomes:

1. Understand basic cryptographic algorithms, message and web authentication and security issues.
2. Identify information system requirements for both of them such as client and server.
3. Understand the current legal issues towards information security.

Software Required: C/JAVA, Ubuntu.

List of Experiments:

1. Write a C program that contains a string (char pointer) with a value 'Hello world'.

   The program should XOR each character in this string with 0 and display the result.

2. Write a C program that contains a string (char pointer) with a value 'Hello world'.

   The program should AND or and XOR each character in this string with 127 and display the result.

3. Write a Java program to perform encryption and decryption using the following algorithms

   (a) Ceaser cipher

   (b) Substitution cipher

   (c) Hill Cipher

4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.

7. Write the RC4 logic in Java Using Java cryptography; encrypt the text “Hello world” using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement the RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA

Course Code	Course Name	Course Structure			
		L	T	P	C
P21AML01	Machine Learning Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Objectives:

1. To introduce students to the basic concepts and techniques of Machine Learning.
2. To develop skills of using recent machine learning software for solving practical problems.
3. To gain experience of doing independent study and research.

Course Outcomes:

1. Design java/python programs for various learning algorithms.
2. Apply appropriate data sets to the machine learning algorithms.
3. Identify and apply machine Learning algorithms to solve real world problems.

List of Experiments:

1. Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build a prediction model to perform logistic regression
5. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.
6. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file.
7. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.

8. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.

9. Perform clustering using k-means clustering algorithm.

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

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

Course Code	Course Name	Course Structure			
		L	T	P	C
P21CSL11	Unified Modeling Languages Lab	0	0	3	1.5

Internal Marks: 15 External Marks: 35

Course Prerequisite: Object Oriented Programming concepts
Course Objectives: The objective of this lab is

1. To know the practical issues of the different object-oriented analysis and design concepts.
2. Inculcate the art of object-oriented software analysis and design.
3. Apply forward and reverse engineering of a software system.
4. Carry out the analysis and design of a system in an object-oriented way.

Course Outcomes:

1. Know the syntax of different UML diagram
2. Create use case documents that capture requirements for a software system.
3. Create class diagrams that model both the domain model and design model of a software system.
4. Create interaction diagrams that model the dynamic aspects of a software system.
5. Write code that builds a software system.
6. Develop simple applications

Note: For performing the experiments consider any case study (ATM/ Banking / Library / Hospital management systems)

List of Experiments:

Experiment 1:

(Familiarization with Rational Rose or Umbrella environment

Experiment 2:

1. Identify and analyze events
2. Identify Use cases
3. Develop event table

Experiment 3:

1. Identify & analyze domain classes
2. Represent use cases and a domain class diagram using Rational Rose
3. Develop CRUD matrix to represent relationships between use cases and problem domain classes

Experiment 4:

1. Develop Use case diagrams

2. Develop elaborate Use case descriptions & scenarios

Experiment 5:

1. Develop system sequence diagrams and high-level sequence diagrams for each use case
2. Identify MVC classes / objects for each use case
3. Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects

Experiment 6:

1. Develop detailed design class model (use GRASP patterns for responsibility assignment)
2. Develop three-layer package diagrams for each case study

Experiment 7:

1. Develop Use case Packages
2. Develop component diagrams
3. Identify relationships between use cases and represent them
4. Refine domain class model by showing all the associations among classes

Experiment 8:
Develop sample diagrams for other UML diagrams - state chart diagrams, activity diagrams and deployment diagrams.

Course Code	Course Name	Course Structure			
		L	T	P	C
P21XXXXX	Design Thinking for Innovation	2	0	0	0

Internal Marks: 30 External Marks: 70

UNIT–I: Design thinking Evolution
Definitions and stories. Design thinking Importance, and Impact-History and Evolution of Design Thinking, - Three Space of Innovation in Design Thinking- knowledge funnel - Design Thinking Process, -Design thinking mindset for innovation

UNIT–II: Building confidence, Mindset and Building Team
Myths of Innovation- Myths of Creativity-Creative Confidence-Innovators DNA - 5 forces of growth (SEPIA),- 5 frictional forces (DCAFE),- 3 capacity levers (VAL)-Building Design Teams.

UNIT–III: Empathy-Define
Initial Problem Description - beginner's mindset-5whys,- persona development-Empathy mapping-interview with empathy and stories collection-Question the critical assumptions -Reframe Problem Definition – (PoV) point of view- how might we

UNIT–IV: Ideation
Ideation and Visualization- Brainstorming-SCAMPER-Mind mapping-sketch –structure idea-Storyboard-Customer Co-Creation-Provocation-Role-play

UNIT–V: Prototyping -Testing
Step-by-step prototyping & low fidelity prototyping -Testing Prototyping -feedback capturing grid, conduct A/B Testing-Experiment grid, user retrospective board-Create a Pitch of the prototype

Text Books:

1. An AVA Book, "Design Thinking", AVA Publishing, 2010
2. Dr.BalaRamaduri, "Karmic Design Thinking", 2020, ISBN:978-9354190100

Reference Books:

1. proach", 3rd edition, Springer, 2007
2. Tom Kelley, Jonathan Littman, "Ten Faces in Innovation", Currency Books, 2006
3. Liedtka, Jeanne and Ogilvie, Timothy, Ten Tools for Design Thinking
4. The Design Thinking Playbook: Mindful Digital Transformation of Teams, Products, Services, Businesses, and Ecosystems by Michael Lewrick
5. The Myths of Innovation by Scott Berkun, Publisher(s): O'Reilly Media, IncISBN: 9781449389628

6. The Myths of Creativity: The Truth About How Innovative Companies and People Generate Great Ideas, D BurkusJossey-Bass, San Francisco, CA (2014), 214 pp, ISBN: 978-1-118-61114-2

7. Creative Confidence: Unleashing the Creative Potential within Us All by (Author), David Kelley(Author)

8. The innovator's DNA: mastering the five skills of disruptive innovators Author: Dyer, JeffGregersen, Hal B, 1958-Christensen, Clayton MPublished: Boston, Mass: Harvard Business Press, [2011]

9. Collective Genius: The Art and Practice of Leading Innovation, Authors: Linda A Hill, Greg Brandeau, Emily Truelove, Kent Lineback

10. Change by Design, by Tim Brown

11. Unmukt-Science and Art of Design Thinking Authors Arun Jain School of Design Thinking 2019

12. The Design Thinking Play Book by Michael Lewrick, Patrick Link& Larry Leifer, Wiley Press,2018

13. The Design of Business: Why Design Thinking Is the Next Competitive Advantage. Martin, R. (2009). Boston, MA: Harvard Business Press.

Online Resources:

1. https://www.interaction-design.org/literature/topics/design-thinking

2. https://www.interaction-design.org/literature/article/how-to-develop-an-empathic-approach-in-design-thinking

3. https://medium.com/dc-design/what-is-human-centered-design-6711c09e2779

4. https://think.design/user-design-research/user-testing/

5. Mentor-DesignThinking.pdf (aim.gov.in)

6. Mentor-DesignThinking.pdf (aim.gov.in)