NITTE MEENAKSHI INSTITUTE OF TECHNOLOGY
(AN AUTONOMOUS INSTITUTION UNDER NITTE EDUCATION TRUST)

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

Curriculum Handbook for UG-Information Science and Engineering

VISION AND MISSION OF DEPARTMENT

1 Vision

To build a strong research and teaching environment in the field of Information Technology to meet the eve revolving global need sand to equip students with the latest knowledge, skills and practical orientation to face challenges in IT profession.

2 Mission

- To offer comprehensive educational programs in the field of Information Technology producing highly accomplished graduates.
- To inculcate among the students culture of research and innovation.
- To encourage students to participate in co-curricular and extra-Curricular activities leading to enhancement of their social and professional skills.

PROGRAMME EDUCATIONAL OBJECTIVE (PEOs)

1. Graduates will progress in their careers in IT industries of repute.

2. Graduates will succeed in higher studies and research.

3. Graduates of Information Science and Engineering will demonstrate highest integrity with ethical values, good communication skills, leadership qualities and self-learning abilities.

PROGRAM OUTCOMES (POs)

Programme Outcomes	
PO-1	Students will be able to apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
PO-2	Students will be able to identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences
PO-3	Students will be able to design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations
PO-4	Students will be able to use research-based knowledge and research methods including design of experiments, analysis and interpretation of data and synthesis of the information to provide valid conclusions.
PO-5	Students will be able to create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
PO-6	Students will be able to apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice .
PO-7	Students will be able to understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
PO-8	Students will be able to apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO-9	Students will be able to Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO-10	Students will be able to communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations.
PO-11	Students will be able to demonstrate knowledge and understanding of the engineering and management principles and apply these to ones own work, as a member and leader in a team, to manage projects and in multidisciplinary and give and receive clear instructions.
PO-12	Students will be able to recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES (PSOs)

Programme Specific Outcomes	
PSO-1	Student will be able to understand the architecture and working of computer system with relevant system software, and apply appropriate system calls.
PSO-2	Student will be able to apply mathematical methodologies in modeling real world problems for the development of software applications using algorithms, data structures and programming tools.

Process of Defining Vision and Mission of the Department

Developing strong vision and mission statements will help stakeholders of the college to attain their respective goals.

The process of defining vision and mission statements for department was started with formation of Program Assessment Committee with Head of the Department, being the chair person for the entire process.

Committee comprises of

- All faculty members.
- Distinguished students of 1st and 2nd years.

Functionality

- formulate few vision and mission statements.
- Refine selected Vision and Mission statement according to the suggestions of Departmental Advisory Committee.

Department advisory committe comprising of

- Distinguished academicians.
- Experts from industry.
- Alumini
- Dean-Academic
- Senior faculty members
- Distinguished students from final year

Department advisory committe functionalities

- Formulate few Vision and Mission Statements by program assessment committee.
- Presented in the advisory committee for suggestions.
- Refine selected Vision and Mission according to the suggestions of advisory committee.
- Finalize the Vision and Mission statements in consultation with departmental advisory committee.

Process used for establishing the PEOs

Program Coordinator, PAC and DAC are involved in establishing the PEOs.

The process of establishing PEOs are as follows:

- PAC will refer the guidelines of NBA while establishing PEOs.
- PAC will formulate few PEOs which are in line with departmental mission statement.
- PAC will discuss about the attainment levels in terms of percentage.
- PAC will refer NASCOM and Government reports for further refinement of PEOs
- Formulated PEOs are Presented to DAC
- DAC will go through the PEO Statements, approve/suggest modifications.
- PAC will refine the selected PEOs according to the suggestions given by the DAC.
- Finalized PEOs with their target attainment level are presented to IQAC
- PAC will publish finalized PEO statements.

SEMESTER:III(2017 Scheme)

sl.no	subject code	subject name	dept handling the subject	teaching hours/week			Examination			credits
				L	T	P	CIE	SEE	TOTAL	
1	17MAT31	ENGINEERING MATHEMATICS III	MATHS	3	2	0	50	50	100	4
2	17IS32	DIGITAL DESIGN	ISE	3	2	0	50	50	100	4
3	17IS33	DATA STRUCTURES USING C	ISE	4	0	0	50	50	100	4
4	17IS34	DISCRETE MATHEMATICS	ISE	3	2	0	50	50	100	4
5	17IS35	COMPUTER ORGANIZATION AND ARCHITECTURE	ISE	4	0	0	50	50	100	4
6	17ISL36	DIGITAL DESIGN LAB	ISE	0	0	2	50	50	100	1
7	17ISL37	DATA STRUCTURES LAB	ISE	0	0	2	50	50	100	1
1	17ISL38	UNIX AND SHELL PROGRAMMING LAB	ISE	0	2	2	50	50	100	2
							400	400	800	25

SEMESTER:IV(2017 Scheme)

sl.no	subject code	subject name	dept handling the subject	teaching hours/week			Examination			credits
				L	T	P	CIE	SEE	TOTAL	
1	17MAT41	ENGINEERING MATHEMATICS IV	MATHS	3	2	0	50	50	100	4
2	17IS42	ANALYSIS AND DESIGN OF ALGORITHMS	ISE	4	0	0	50	50	100	4
3	17IS43	OBJECT ORIENTED PROGRAMMING WITH C++	ISE	3	0	0	50	50	100	3
4	17IS44	MICROCONTROLLER	ISE	4	1	0	50	50	100	4.5
5	17IS45	OPERATING SYSTEMS	ISE	4	0	0	50	50	100	4
6	17ISL46	ANALYSIS AND DESIGN OF ALGORITHMS LAB	ISE	0	2	2	50	50	100	2
7	17ISL47	OBJECT ORIENTED PROGRAMMING WITH C++ LAB	ISE	0	0	2	50	50	100	1
1	17ISL48	MICROCONTROLLER LAB	ISE	0	2	2	50	50	100	2
							400	400	800	25

SEMESTER:V(2014 SCHEME)

sl.no	subject code	subject name	dept handling the subject	teaching hours/week			Examination			credits
				L	T	P	CIE	SEE	TOTAL	
1	14IS51	COMPUTER NETWORKS-I	ISE	4	0	0	50	50	100	4
2	14IS52	SYSTEM PROGRAMMING	ISE	4	0	0	50	50	100	4
3	14IS53	OPERATING SYSTEMS	ISE	4	0	0	50	50	100	4
4	14IS54	DATABASE MANAGEMENT SYSTEMS	ISE	4	0	0	50	50	100	4
5	14IS55	FORMAL LANGUAGES AND AUTOMATA THEORY	ISE	4	0	0	50	50	100	4
6	14ISE56X	ELECTIVE CORE -A	ISE	4	0	0	50	50	100	4
7	14ISL57	DATABASE MANAGEMENT SYSTEMS LAB	ISE	0	0	3	50	50	100	1.5
1	14ISL58	SYSTEM PROGRAMMING LAB	ISE	0	0	3	50	50	100	1.5
							400	400	800	27

PROGRAM ELECTIVE-A(2014 SCHEME)

Sl.No	**Subject code**	**Subject Name**
1	14ISE561	C# AND .NET PROGRAMMING
2	14ISE562	OBJECT ORIENTED ANALYSIS AND DESIGN
3	14ISE563	PYTHON FOR DATA SCIENCE
4	14ISE564	COMPUTER GRAPHICS WITH OPENGL AND CUDA
5	14ISE565	DIGITAL IMAGE PROCESSING

SEMESTER:VI(2014 SCHEME)

sl.no	subject code	subject name	dept handling the subject	teaching hours/week			Examination			credits
				L	T	P	CIE	SEE	TOTAL	
1	14IS61	DATA MINING	ISE	4	1	0	50	50	100	4
2	14IS62	COMPUTER NETWORKS-II	ISE	4	1	0	50	50	100	4
3	14IS63	SOFTWARE ENGINEERING	ISE	4	1	0	50	50	100	4
4	14IS64	FUNDAMENTALS OF JAVA PROGRAMMING	ISE	4	0	0	50	50	100	4
5	14ISE65X	ELECTIVE CORE-B	ISE	4	0	0	50	50	100	4
6	14IS066X	OPEN ELECTIVE-C	ISE	3	1	0	50	50	100	3
7	14ISL67	COMPUTER NETWORKS LAB	ISE	0	0	3	50	50	100	1.5
8	14ISL68	FUNDAMENTALS OF JAVA PROGRAMMING LAB	ISE	0	0	3	50	50	100	1.5
9	14ISL69	IDENTIFICATION OF PROJECT AND PRESENTATION	ISE	0	0	4	50	50	100	
							400	400	800	26

ELECTIVE CORE(E)-B

Sl.No	**Subject code**	**Subject Name**
1	14ISE651	DISTRIBUTED SYSTEMS
2	14ISE652	INTERNET OF THINGS
3	14ISE653	STORAGE AREA NETWORKS
4	14ISE654	COMPILER CONSTRUCTION

ELECTIVE OPEN(O)-C

Sl.No	**Subject code**	**Subject Name**
1	14IS0661	INTERNET OF THINGS
2	14IS0662	OBJECT ORIENTED PROGRAMMING WITH C++
3	14IS0663	UNIX FUNDAMENTALS
4	14IS0664	ESSENTIALS OF INFORMATION TECHNOLOGY

SEMESTER:VII(2014-SCHEME)

sl.no	subject code	subject name	dept handling the subject	teaching hours/week			Examination			credits
				L	T	P	CIE	SEE	TOTAL	
1	14IS71	PARALLEL COMPUTING	ISE	4	1	0	50	50	100	4
2	14IS72	SOFTWARE PROJECT MANAGEMENT	ISE	3	0	0	50	50	100	3
3	14ISH73	VENTURE PROCESS MANAGEMENT AND IPR	ISE	3	0	0	50	50	100	3
4	14ISE74X	ELECTIVE CORE-D	ISE	4	0	0	50	50	100	4
5	14ISO75X	OPEN ELECTIVE-E	ISE	3	0	0	50	50	100	3
6	14ISL76	WEB TECHNOLOGY LAB	ISE	0	0	3	50	50	100	1.5
7	14ISL77	DISTRIBUTED COMPUTING LAB	ISE	0	0	3	50	50	100	1.5
8	14ISLP78	PROJECT AND TECHNICAL SEMINAR	ISE	0	0	3	50	50	100	
9	14ISP79	INTERNSHIP / SELF STUDY/MINI PROJECT	ISE	0	0	4	50	50	100	2
							450	450	900	22

ELECTIVE CORE(E)-D

Sl.No	**Subject code**	**Subject Name**
1	14ISE741	BIG DATA
2	14ISE742	MACHINE LEARNING
3	14ISE743	ANDROID APPLICATION DEVELOPMENT AND VERSION CONTROL REPOSITORY
4	14ISE744	CLOUD COMPUTING

ELECTIVE OPEN (O)-E

Sl.No	Subject code	Subject Name
1	14ISO751	C# AND .NET PROGRAMMING
2	14ISO752	FUNDAMENTALS OF JAVA
3	14ISO753	DESIGN AND DEVELOPMENT OF WEB APPLICATIONS
4	14ISO754	MOBILE APP DEVELOPMENT
5	14ISO755	PYTHON PROGRAMMING

SEMESTER:VIII(2014 SCHEME)

sl.no	subject code	subject name	dept handling the subject	teaching hours/week			Examination			credits
				L	T	P	CIE	SEE	TOTAL	
1	14IS81	INFORMATION AND NETWORK SECURITY	ISE	4	0	0	50	50	100	4
2	14ISH82	FOSS AND CYBER LAWS	ISE	4	0	0	50	50	100	3
3	14ISE83	ELECTIVE CORE-F	ISE	3	0	0	50	50	100	3
4	14ISP84	PROJECT	ISE	0	0	28	50	50	100	14
							200	200	400	24

ELECTIVE CORE(E)-F

Sl.No	Subject code	Subject Name
1	14ISE831	ADHOC NETWORKS
2	14ISE832	MACHINE LEARNING
3	14ISE833	ADVANCED JAVA
4	14ISE834	ANALYSIS OF COMPUTER NETWORKS
5	14ISE835	CLIENT SERVER COMPUTING

Semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Engineering Mathematics-III	Course Code:17MAT31
L-T-P: 3-2-0	Credits: 04
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Course Outcomes:

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

1. Apply Fourier theory to analyse harmonic and periodic functions.
2. Apply concepts of Laplace transforms to control systems.
3. Apply the concept of numerical interpolation to fit the given data and extrapolate the same.
4. Numerically integrate and solve differential equations arising in engineering field.
5. Find Eigen values and Eigen vectors arising in engineering field using numerical field.

Teaching Methodology:

1. Blackboard teaching
2. Tutorials

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Evaluation of Tutorials -10 marks.
3. Quiz/ Aptitude Test-10 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3	2												3
CO2	3	2												3
CO3	3	2												3
CO4	3	2												3
CO5	3	2												3
17MAT31	3	2												3

Course Content

UNIT1
Laplace Transforms: Definition, Transforms of standard functions (derivation and problems), Transforms of , , , Laplace transforms of derivatives and integrals (no derivations), Laplace transforms of periodic functions, unit step function (no derivations), Dirac delta function . Inverse Laplace transforms, convolution theorem (no proof), solutions of 1st and 2nd order ODE using Laplace transforms.

08 HOURS

UNIT-II
Fourier series: Eulers formulae, Dirichlets conditions for Fourier series expansion, change of interval, Even and odd function, half range series, Practical harmonic analysis. Fourier Transforms: Definition, Complex Fourier transforms, Cosine and Sine transforms, Inverse Fourier transforms.

08 HOURS

UNIT-III
Interpolation: Newtons forward and backward formulae, Newtons divided difference formulae and Lagranges formula for unequal intervals and inverse interpolation by Lagranges formula, Stirlings and Bessels central difference formula, Numerical differentiation with Newtons forward and backward difference interpolation.

08 HOURS

UNIT-IV
Numerical Integration by Trapezoidal, Simpsons and rule, Weddles rule, Gaussian Quadrature. Numerical solution of ordinary differential equations: Taylors series method, Runge-Kutta 4th order method, Milnes predictor corrector method.

07 HOURS

UNIT-V
Linear algebra: LU decomposition, Solution of Tridiagonal system using Thomas algorithm, Eigen values of symmetric matrix by Jacobi method, Reduction to Tridiagonal system by Givens method, Largest Eigen value by Power method.

08 HOURS

Text Books:

1. Higher Engg. mathematicsby Dr. B S Grewal, 42nd Edition, 2012.
2. Advanced Engg. Mathematics by Erwin E Kreyszig, 10th edition, Wiley. 2013.
3. Introductory methods of numerical analysis, by S SSastry, PHI India, 2012.

Semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Digital Design	Course Code:: 17IS32
L-T-P: 3-2-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Basic Knowledge on Physics and Electronics

Course Outcomes:

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

1. Understand the basic concepts of logic families, Boolean algebra, combinational and sequential circuits.
2. Apply the concepts of simplification to realize the digital circuits.
3. Evaluate different techniques to realize the digital circuits.
4. Design the digital circuits for various applications.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Regular review of students by asking questions based on topics covered in the class
4. Flipped class room /TPS
5. Assignment

Assessment Methods:

1. Rubrics for evaluating Assignments for 10 Marks.
2. Aptitude Testsfor 10 marks.
3. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3												2	
CO2	3	2							1	1		1	2	
CO3	3	2											2	
CO4	3	2	3		1	1			1	1		1	2	
17IS32	3	2	2		1	1			1	1		1	3	

Course Content

UNIT-I
Boolean Algebra and Combinational Networks: Truth Tables, OR Operation with OR Gates, AND Operation with AND Gates ,NOT Operation, Describing Logic Circuits Algebraically, Evaluating Logic-Circuit Outputs ,Implementing Circuits fromBoolean Expressions, NOR Gates and NAND Gates, Boolean Theorems, DeMorgans Theorems, Universality of NAND, Alternate Logic-Gate Representations, Which Gate Representation to Use, Summary of Methods to Describe Logic Circuits

10 HOURS

UNIT-II
Simplification of Boolean Expressions:Designing Combinational Logic Circuits: SoP & PoS form, Simplifying Logic Circuits, Algebraic Simplification, Karnaugh Map Method, XOR and XNOR Minimization Circuits, Enable/Disable Circuits Analysis. Logic Design with MSI Components and Programmable Logic Devices: Binary adders & Subtractors, Magnitude Comparators, Decimal adder/BCD Adder, Decoder, Encoder, Multiplexer, Parity Generator and Checker

11 HOURS

UNIT-III
Synchronous Sequential Logic: NAND Gate Latch, NOR Gate Latch, Clock Signals and Clocked Flip-Flops, Clocked S-R Flip-Flop, Clocked J-K Flip-Flop, Clocked D Flip-Flop, D Latch(Transparent Latch), Asynchronous Inputs, Master/Slave Flip-Flops, Flip-Flop Applications, Flip-Flop Synchronization, Detecting an Input Sequence,Registers: Data Storage and Transfer, Serial Data Transfer: Shift Registers, Frequency Division and Counting

11 HOURS

UNIT-IV
Counters, D/A Conversion and A/D Conversion: Asynchronous (Ripple) Counters, Counters with MOD Numbers, Asynchronous Down Counter, Propagation delay in Ripple Counters, Synchronous (Parallel) Counters, Presentable Counters, The, Synchronous Counter Design.D/A Conversion and A/D Conversion: Variable Resistor Networks, Binary Ladders, D/A Converters, D/A Accuracy and Resolution, A/D Converter-Counter Method, Continuous A/D Conversion, A/D Conversion using Successive Approximation.

11 HOURS

UNIT-V
VHDL, VERILOG and FPGA: Introduction to VHDL, Capabilities, Hardware Abstraction, Introduction to Verilog HDL, Major Capabilities. FPGA: Introduction, Basic Concepts, Schematics and Logic Symbols, Digital Design and FPGAs, FPGA Based System Design, FPGA Fabrics- FPGA Architecture- SRAM Based FPGAs

09 HOURS

Text Books:

1. Digital Systems Principles and Applications, Ronald J. Tocci, Neal S. Widmer, Gregory L. Moss 9th Edition, 2013, Pearson Education.
2. Digital Principles and Applications by Donald P Leach, Albert Paul Malvino, GoutamSaha. 7th edition,2012,Tata McGraw-Hill

3. Wayne Wolf FPGA based system design, Pearson education Electronics Communication Systems, McGraw Hill, first Edition, 2009

Reference Books:

1. Digital logic and computer design M. Morris Mano
2. Fundamentals of Logic Design, Charles H. Roth, Jr., 5th Edition, Thomson, 2004.
3. https://onlinecourses.nptel.ac.in/noc18cs30/preview

SemesterIII

Department:Information Science and Engineering	Course Type:core
Course Title:Data Structures Using C	Course Code:17IS33
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have knowledge of C language

Course Outcomes:

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

1. Understand the fundamentals of data structures.
2. Apply the concept of arrays, pointers and recursion in problem solving.
3. Apply the notions of stacks and queues to design application specific module.
4. Apply the concept of linked lists and trees for the given applications.

Teaching Methodology:

1. Black board teaching
2. Power Point presentations (if needed)
3. Tutorials
4. Course Project
5. Certification based Learning

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for Evaluation of Course Project for 10 marks .
3. Online certification/aptitude test (GATE syllabus) for10 Marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													1
CO2	3	2	2	1					2	1		1		2
CO3	3	2	2	1					2	1		1		2
CO4	3	2	2	1					2	1		1		2
17IS33	3	2	2	1					2	1		1		2

Course Content

UNIT1
Introduction to data structures: revisit to pointers in C, Implementing one dimensional array. The stack: DefinitionPrimitive operations, Representing Stacks in C Implementing the POP operation, testing for exceptional conditions, implementing the PUSH operation. Infix, Postfix and PrefixBasic Definitions, evaluating a postfix expression, converting an expression from infix to postfix.

10 HOURS

UNIT-II
Recursion: Recursive Definition and Processesfactorial function, multiplication of natural numbers. Fibonacci sequence, binary search Recursion in C factorial, binary search, recursive chains Writing Recursive ProgramsTowers of Hanoi Queues: The Queue and its Sequential RepresentationC Implementation of Queues, Insert Operation, Priority Queue, Array Implementation of a Priority Queue

10 HOURS

UNIT-III
Linked Lists -Inserting and Removing Nodes from a List, Linked Implementation of Stacks, getnode and freenode Operations, Linked Implementation of Queues, Linked List as a Data Structure, List Operations, List Implementation of Priority Queues, Header Nodes, Array Implementation of Lists, Limitations of the Array Implementation. Lists in CAllocating and Freeing of Dynamic variables, Linked Lists using Dynamic Variables, Queues as List in C, List Operations in C, Non integer and Non homogeneous Lists, Comparing the Dynamic and Array Implementations of Lists, Implementing Header Nodes in linked list.

12 HOURS

UNIT-IV
Other List StructuresCircular Lists, Stack as a Circular List, Queue as a Circular List, Primitive Operations on Circular Lists, Header-nodes, Doubly Linked Lists, Binary Trees Operations on Binary Trees, Applications of Binary Trees.Binary Tree RepresentationsNode Representation of Binary Trees, Internal and External Nodes

10 HOURS

UNIT-V
Binary Trees:Implicit Array Representation of Binary Trees, Binary TreeTraversal in C, Threaded Binary Trees, Heterogeneous Binary Trees. Representing Lists as Binary Trees, Trees and Their Applications C Representations of Trees, Tree Traversals, General Expressions as Trees, evaluating an expression tree, constructing a Tree. Hashing: Open Addressing, Deleting Items from a Hash Table.

10 HOURS

Text Books:

1. Data Structure using C, Aaron M. Tanenbaum, YedidyahLangsam and Moshe J. Augenstein, Pearson Education/PHI, 2006
2. Introduction to algorithms, Thomas H cormen, Charles E Leiserson, Ronald L Rivest and Clifford stein, The MIT press, third edition, 2009

Reference Books:

1. The C Programming Language, Brian W Kernighan and Dennis M Ritchie, Prentice Hall Software Series, 2nd Edition.
2. Data structures and algorithms analysis in C, Allen Weiss, Second Edition, Pearson IN
3. Data Structures a Pseudocode approach with C,RichardF.GilbergandBehrouzA.Forouzan, Thomson, 2005.
4. Data Structures and Program Design in C, Robert Kruse and BruceLeung, Pearson.
5. Foundations of Data Structures, IIT Bombay, https://www.edx.org/course/foundations-of-data-structures.

Semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Discrete Mathematics	Course Code:17IS34
L-T-P: 3-2-0	Credits: 04
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Mathematics, Elementsof statistics and combinatorics, Basic knowledge of Graphs.

Course Outcomes:

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

1. Apply the propositional and predicate logic in symbolic representations and validity tests.
2. Solve recursive problems using combinatorial properties.
3. Solve problems related to Information Technologyusing mathematical reasoning, relations and functions.
4. Understand the concepts of graph theory, theorems and derivations.
5. Demonstrate the applications of discrete mathematics in different areas of Engineering.

Teaching Methodology:

1. Black board teaching
2. Tutorials
3. Assignment/ Case Study

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Tutorial evaluation 10 marks.
3. Rubrics for evaluating assignment/case study 10 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3	2												2
CO2	3	2												2
CO3	3	3	2											2
CO4	3	1												2
CO5	3	3	2			1			1	1		1		2
17IS34	3	3	2			1			1	1		1		2

Course Content

UNIT1
Logics and Proofs Propositional Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to proofs.

08 HOURS

UNIT-II
Combinatorics Recurrence Relations and Generating FunctionsSome number sequences, Linear homogeneous recurrence relations of order one and order two, Linear non-homogeneous recurrence relations of order one and two, Generating functions- Solution of linear non-homogeneous recurrence relations of order one and two using generating functions , A geometry example.

9 HOURS

UNIT-III
Relations and Digraphs The Pigeonhole PrincipleSimple form, Strong form. The InclusionExclusion Principle, Combination with repetition, Derangements, Properties of Relations, Equivalence Relations, Computer Representation of Relations and Digraphs, Operations on Relations, Partially ordered Sets, Extremal elements of Posets, Lattices.

08 HOURS

UNIT-IV
Functions Functions, Functions for Computer Science, Growth of Functions, Finite Boolean algebra, Functions on Boolean Algebra, Circuit Designs.

7 HOURS

UNIT-V
Graph Theory: Connected Graphs, Common classes of Graphs, Multigraphs and Digraphs, the Degree of a vertex, Regular Graphs, Degree sequences, Graphs and Matrices, Isomorphism, Eulerian Graphs, Hamiltonian Graphs, Planar Graphs.

7 HOURS

Text Books:

1. Discrete Mathematics and its Applications, Keneth H Rosen, 6th Edition, McGraw-Hill.
2. Discrete Mathematical Structures, Kolman, Busby and Ross, 4th Edition, Pearson Education Asia.
3. Introductory Combinatorics, Richard A. Brualdi, 4th Edition, Pearson.
4. Introduction to Graph Theory, GraryChartrand and Ping Zhang, Tata McGraw-Hill Edition 2006.

Reference Books:

1. Treatise on Discrete Mathematical Structures, JayanGanguly, Revised Edtion 2012, Pearson.
2. http://nptel.ac.in/courses/106106094/
3. http://nptel.ac.in/courses/111106050/

Semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Computer Organization and Architecture	Course Code:17IS35
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Fundamental knowledge of Computer Peripheral System and Functional Units

Course Outcomes:

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

1. Understand structure of computers and its working with the help of Machine Instructions.
2. Understand the working of memory and computing components for the execution of a programinstruction.
3. Design memory storage structures components using fundamental storage elements.
4. Apply bit-level signed and unsigned operations to carryout arithmetic computation.
5. Interpretprocess parallelism and inter process communication for computation performance.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Case Study

Assessment Methods:

1. Two Aptitude tests on GATE syllabus -10 Marks.
2. Rubrics for evaluating case study-10 Marks
3. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3												2	
CO2	3								1	1			2	
CO3	2	2	3				1		1	1			2	
CO4	2	2	2										2	
CO5	3	2					1		1	1			2	
17IS35	3	2	1				1		1	1			2	

Course Content

UNIT-I
Basic Structure of Computers: Computer Types, Functional Units, Basic Operations Concepts, Performance: Relative Performance, Measuring Performance, CPU performance, CPU Performance Equation, SPEC Rating, And Amdahls Law. Machine Instructions and Programs: Memory Location and Addresses, Memory Operations, Instructions and instruction sequencing: Register Transfer Notation, Assembly Language Notation, Instruction Types Instruction set architecture: CISC and RISC, Addressing modes

10 HOURS

UNIT-II
Basic processing unit: Some Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic Operation, Fetching a word from Memory, Storing a word in memory; Execution of a complete Instruction: Branch Instructions; Multiple Bus Organization, Hardwired Control: A Complete Processor; Micro programmed Control: Microinstructions

10 HOURS

UNIT-III
The Memory System: Semiconductor memories, Memory cells - SRAM and DRAM cells, Internal Organization of a memory chip, Organization of a memory unit, Interleaved memories, Cache memory unit - Concept of cache memory, Mapping methods, Organization of a cache memory unit, Fetch and write mechanisms, Memory management unit - Concept of virtual memory, Address translation, Hardware support for memory management.

10 HOURS

UNIT-IV
I/O Organization: Access of I/O devices, Interrupts, Direct Memory Access, Buses, I/O interfaces-Serial port, Parallel port, PCI bus, Serial port, Parallel port, PCI bus, SCSI bus, and USB bus. Arithmetic: Signed Operand Multiplication: Booth Algorithm; Fast Multiplication: Bit Pair recoding of Multipliers, Integer Division, Floating Point Representation.

11 HOURS

UNIT-V
Pipelining: Basic Concepts, Data hazards, Instruction Hazard Multi-core Architecture and parallel processing: Scaling Issues, Thread level parallelism, Communication in Multi-core, Shared Cache CMP, Performance ISSUE, Consistency and Coherence

11 HOURS

Text Books:

1. Computer Organization and embedded system, Carl Hamacher, ZvonkoVranesic, SafwatZaky, NaraigManjikiran5th Edition, TMH

Reference Books:

1. Computer Organization and Architecture, WilliamStallings, 7thEdition, PHI, 2006.

2. Computer Organization and Design, David A . Patterson and John L. Hennessy, The Hardware/Software Interface, ARM Edition, Elsevier, 2011 (Section 1.4)
3. http://nptel.ac.in/courses/106104025/2 (Unit-V- Multicore Architecture and parallel processing)
4. http://nptel.ac.in/courses/106102062/1.
5. https://www.coursera.org/learn/comparch.
6. https://www.edx.org/course/computation-structures-3-computer-mitx-6-004-3x-0

semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Digital Logic Design Lab	Course Code:17ISL36
L-T-P: 0-0-2	Credits: 01
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Basic Knowledge on Physics and Electronics

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

1. Apply the concepts of simplification to realize the Logic circuits.
2. Design the Logic circuits using combinational and sequential Circuits.
3. Implement the Logic circuits using Simulation tool.

Teaching Methodology:

1. Blackboard Teaching

Assessment Methods:

1. Rubrics for evaluating Viva-Voce 10 Marks.
2. MSE for 20 marks.
3. Record and observation20 marks.
4. SEE will be evaluated for 50 marks

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3	3		3									2	
CO2	3	2	3	2					2	2		1	2	
CO3	3	2	2	2	3								2	
17ISL36	3	2	2	2	3				1	1		1	3	

Course Content

PART-A

1. Realization of Full Adder and Full Sustractor Using NAND gate.
2. Implementation of Full Adder and Full Subsstractor using 4 : 1 MUX (IC 74153).
3. Converting Binary Number to Seven Segment Display Using Decoder IC 7747.
4. Design and Verify Parity Generator and Checker (Even Parity and Odd Parity).
5. Realization of SR, JK, D and T Flip Flops using NAND gates.
6. Implementation of J.K Master Slave Flip Flop using NAND Gates

PART-B

1. Design and Implement Asynchronous UP, DOWN and UP and DOWN Using IC-7476 (JK).
2. Design and Implementation of MOD-N Synchronous Counter Using IC-7476 (JK).
3. Implementation of Ring Counter and Johnson Counter Shift Register Using 7495.
4. Digital Analog Converter (DAC) using R to 2R Ladder Method.
5. Realizing the logic gates using HDL.
6. Realizing the combinational designs using HDL.
    - 2 to 4 decoder.
    - 8 to 3 (Encoder Without Priority and With Priority).
    - 8 to 1 Multiplexer .
    - 4 bit binary to gray Converter.
    - Multiplexer, De-Multiplexer, Comparator.

semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Data Structures Lab	Course Code:17ISL37
L-T-P: 0-0-2	Credits: 01
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have knowledge of C language

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

1. Develop programming solutions for the given problems using arrays/pointers/recursion.
2. Design programming solutions for the given problem using stacks and queues.
3. Develop programming solutions for the real time application module using linked list concepts.
4. Design hierarchical based programming solutions using different tree traversal techniques.

Teaching Methodology:

1. Black board teaching
2. Power Point presentations (if needed)
3. Course project.

Assessment Methods:

1. Rubrics for evaluating laboratory experiments for 20 marks
2. Rubrics for evaluating course project for 10 marks.
3. MSE for 20 marks.
4. SEE examination will be evaluated for 50 marks (Course Project-10 marks + Lab examination -40 marks).

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	2	3	3	2					2	2		1		2
CO2	2	3	3	2					2	2		1		2
CO3	2	3	3	2					2	2		1		2
CO4	2	3	3	2					2	2		1		2
17ISL37	2	3	3	2					2	2		1		2

Course Content

1. Design and implement C program to demonstrate advantages/usage of pointers. (E.g. add two Matrices using pointer concept, Functions for string manipulation, String manipulation functions).
2. Design and implement a database application in C. (E.g. Student/Faculty information using structures).
3. Design and implement a stack (Array implementation/ Linked list implementation) and demonstrate its working with necessary inputs. Display the appropriate messages in case of exceptions
4. Design and implement an algorithm for conversion of an expression from one form to another. Demonstrate its working with suitable inputs.
5. Design and implement an algorithm to evaluate an arithmetic expressions which may be any form (postfix, prefix, infix), and demonstrate its working with suitable examples.
6. Design and implement a given type of (ordinary queue, circular queue) queue in C (array implementation/ Linked list implementation). And demonstrate its working with suitable inputs. Display appropriate messages in case of exceptions.
7. Design and implement a dynamic list (Singly linked list/ doubly linked list) to store any information which needs a linear data structure.
8. Design and implement binary tree and demonstrate its working.

semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Unix and Shell Programming Lab	Course Code:17ISL38
L-T-P: 0-2-2	Credits: 02
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

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

1. Understand fundamentals of UNIX based computer systems.
2. Apply UNIX commands to solve given problem.
3. Develop shell/awk scripts to solve the given problem.
4. Develop programming solution for a given problem using file APIs .

Teaching Methodology:

1. Tutorial
2. Laboratory Experiments

Assessment Methods:

1. Rubrics for evaluating laboratory experiments for 30 marks
2. MSE for 20 marks.
3. SEE examination will be evaluated for 50 marks .

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3												2	
CO2	3	2											2	
CO3	3	2	2										2	
CO4	3	2	2										2	
17ISL38	3	2	1										2	

Course Content

PART-A

1. Execution of various file/directory handling commands.
2. Simple shell script for basic arithmetic and logical calculations.
3. Shell scripts to check various attributes of files and directories.
4. Shell scripts to check and list attributes of processes.
5. Write awk script that uses all of its features.
6. Write a shell script to display list of users currently logged in.

PART-B

1. Write a C/C++ program to implement the following commands using general file APIs (i)cat (ii)cp (iii)ln/rename( ).
2. Write a C/C++ program to create a file called file1 in blocking read-write mode and show how you can use fcntl api to modify its access control flags to non-blocking read-write mode.
3. Write a C/C++ program to duplicate the file descriptor of a file Foo to standard input file descriptor.
4. Write a C/C++ program to query and display the different attributes associated with a file.
5. Write a C/C++ program to implement ls l command using general file apis.
6. Write C/C++ program to read and display the last 10 characters of the input file.
7. Write a C/C++ program to demonstrate masking of read/write/execute permission of a specified input file for user group and others category.

Semester:V

Department:Information Science and Engineering	Course Type:core
Course Title: Computer Networks I	Course Code:14IS51
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Course Outcomes:

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

1. Understand the Layered architecture of computer networks.
2. Analyze the various technologies of digital transmission.
3. Apply error detection and flow control methods of data link layer to improve the Quality of Service.
4. Describe various protocols of Physicaland Data Link Layers.
5. Implement different network topologies using simulation tool.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Course Project

Assessment Methods:

1. Rubrics for evaluating Course project 20 marks.
2. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
3. SEE for 100 marks will be evaluated for 50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3												2	
CO2	3	2											2	
CO3	3	2											2	
CO4	3												2	
CO5	1	2	2	2	2				1	1		1	2	
14IS51	3	2	2	1	2				1	1		1	2	

Course Content

UNIT1
Data Communications and Fundamentals Introduction: Digital transmission fundamental digital representation of information, why digital communication? Network Models: Layered tasks; The OSI Model and the layers in the OSI model; TCP / IP Protocol Suite, addressing.

10 HOURS

UNIT-II
Physical Layer and Media Data and Signals: Analog and digital signals; Transmission impairment; Data rate limits; Performance; Digital Transmission: Digital-to-Digital conversion; Analog-to-Digital conversion; Transmission modes. Bandwidth Utilization: Multiplexing; Spread spectrum. Transmission Media: Twisted pair cable, Coaxial cable, Fiber-Optic cable, Radio waves, Microwaves, Infrared.

11 HOURS

UNIT-III
Data Link Layer Error Detection and Correction: Introduction to error detection / correction; Block coding; linear block codes; cyclic codes, Checksum. Data Link Control: Framing; Flow and Error control; Protocols; Noiseless channels; Noisy channels; HDLC; Point-to-point Protocol - framing, transition phases.

11 HOURS

UNIT-IV
Data Link Layer Continued Multiple Access, Ethernet: Random Access; Controlled Access; Channelization. Wired LAN's: Ethernet: IEEE standards; Standard Ethernet and changes in the standard; Fast Ethernet; Gigabit Ethernet.

11 HOURS

UNIT-V
Other Technologies WirelessLANs: IEEE 802.11; Bluetooth. Connection of LANs: Connecting devices; Backbone Networks; Virtual LANs ,SONET/SDH: Architecture, Layers, Frames, STS multiplexing.

09 HOURS

Text Books:

1. Behrouz A. Forouzan Data Communications and Networking, 4th Edition, Tata McGraw-Hill, 2006.
2. Alberto Leon-garcia and IndraWidjaja Communication Networks, Second Edition, Tata McGraw-Hill, 2004.

Reference Books:

1. William Stallings: Data and Computer Communication, 8th Edition, Pearson Education, 2007.
2. Andrew S. Tanenbaum: Computer Networks, 4th Edition, PHI.

Semester:V

Department:Information Science and Engineering	Course Type:core
Course Title:System Programming	Course Code:14IS52
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- UNIX and Shell Programming.
- Fundamentals of operating systems.

Course Outcomes:

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

1. Describe need for Standardizing the UNIX Environment.
2. Apply appropriate UNIX File APIs to solve the given problem.
3. Apply the appropriate Unix APIs for process and job control.
4. Describe signal concepts, use signal related APIs to solve the given problem.
5. Demonstrate inter-process communication using different IPC structures.
6. Understand the working of loaders and linkers.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Case studies
4. Programming Assignments

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluation of case studies.
3. Rubrics for evaluation of Programming Assignments.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3	1												
CO2	3	2	3						1	1		1	3	
CO3	3	2	3						1	1		1	3	
CO4	3	1	3						1	1		1	3	
CO5	3	1	3											
CO6	3	1	3						1	1		1	3	
14IS52	3	2	3						1	1		1	3	

Course Content

UNIT1
INTRODUCTION: UNIX and ANSI Standards: The ANSI C Standard, Difference between ANSI C and C++, The POSIX Standards, UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics. UNIX FILES: File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.

10 HOURS

UNIT-II
UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs. UNIX PROCESSES: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions.

10 HOURS

UNIT-III
PROCESS CONTROL: Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, waited, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times. PROCESS RELATIONSHIPS: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp, tcsetpgrp, and tcgetsid Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups.

10 HOURS

UNIT-IV
SIGNALS: Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers. INTERPROCESS COMMUNICATION: Introduction; Pipes, popen, pclose Functions; Coprocesses, FIFOs; Message Queues; Semaphores; Shared Memory.

10 HOURS

UNIT-V
DAEMON PROCESSES: Introduction, Daemon Characteristics, Coding Rules. NETWORK IPC: SOCKETS: Introduction, Socket Descriptors, Addressing, Connection establishment, Data transfer. LINKERS AND LOADERS: Linking and Loading: What do linkers and loaders do, Address Binding, Linking v/s loading, compiler drivers, linking-true life example.

12 HOURS

Text Books:

1. Terrence Chan UNIX System Programming Using C++, Prentice Hall India, 1999.
2. W.Richard Stevens Advanced Programming in the UNIX Environment, 2nd Edition, Addison-Wesley / PHI, 1992.

3. John R. Levine Linkers and Loaders.

Reference Books:

1. Marc J. Rochkind Advanced Unix Programming, 2nd Edition, Pearson Education, 2005.
2. Maurice.J.BachThe Design of the UNIX Operating System, Pearson Education / PHI, 1987.
3. UreshVahalia UNIX Internals, Pearson Education, 2001.

Semester:V

Department:Information Science and Engineering	Course Type:core
Course Title:Operating System	Course Code:14IS53
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Computer Organization and Fundamental of Data structures.

Course Outcomes:

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

1. Understand the operating system objectives and functions.
2. Analyse the performance of scheduling algorithms for the given problems.
3. Analyse the performance of memory management schemes.
4. Apply the deadlock handling mechanisms to solve the given problem.
5. Understand the principles of protection and security mechanisms.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations (if needed)
3. Regular review of students by asking questions based on topics covered in the class
4. Aptitude test .
5. Technical Presentation

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Aptitude Test(gate questions)- 10 marks
3. Rubrics for the evaluation of Technical Presentation- 10 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	2								1	1		1	2	
CO2	3	3		1					1	1		1	2	
CO3	3	3		1					1	1		1	2	
CO4	3	2		1					2	2		1	2	
CO5	3								2	2		1	2	
14IS53	3	3		1					1	1		1	3	

Course Content

UNIT1

Introduction: What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Protection and security; Distributed system; Special purpose systems; Operating system structures: operating system services, user operating system Interface, System calls, Types of system calls, Operating system structure, System boot

10 HOURS

UNIT-II

Process Management: Basic concept; Process scheduling; Operations on processes; Inter process Communication Threads: Overview; Multithreading models; Process scheduling: Basic concepts, Scheduling criteria, scheduling algorithms, multiple processor scheduling, Algorithm evaluation.

10 HOURS

UNIT-III

Process Synchronization: Synchronization, The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors. Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

12 HOURS

UNIT-IV

Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on write; Page replacement; Allocation of frames; Thrashing.

10 HOURS

UNIT-V

File System: File concept; Access methods; Directory structure; File system mounting; file sharing; Protection. Secondary Storage Structures: Disk scheduling; FCFS Scheduling, SSTF scheduling, SCAN,C-SCAN scheduling, Look Scheduling. System Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems.

10 HOURS

Text Books:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne Operating System Principles, 9th edition Wiley- India,2016.

Reference Books:

1. D.M Dhamdhere Operating systems - A concept based Approach, 2nd Edition, Tata McGraw-Hill, 2002.
2. Harvey M Deital Operating systems, 3rd Edition, Addison Wesley, 1990.
3. Operating Systems: Principles and Practice (2nd Edition), by Thomas Anderson and Michael Dahlin.
4. https://nptel.ac.in/downloads/106106144/

Semester:V

Department:Information Science and Engineering	Course Type:core
Course Title:Database Management System	Course Code:14IS54
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have basic programming knowledge.
- General concept of set theory.

Course Outcomes:

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

1. Understand the fundamentals of database, database languages and database architecture.
2. Design the Entity relationship diagrams for the given requirement specification.
3. Develop SQL queries for retrieving data for any given database.
4. Apply normalization techniques to eliminate anomalies for the efficient database design.
5. Understanding the concepts of transaction management.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Course Project

Assessment Methods:

1. Rubrics for evaluating Course project -20 marks.
2. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	2	2	3	1				1	1	1		1		2
CO3	3	2	3	2				1	1	1		1		2
CO4	2	2	3	2				1	1	1		1		2
CO5	2													2
14IS54	3	2	2	1				1	1	1		1		3

Course Content

UNIT1

Introduction: Database System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Object-Based semi-structured Databases, Data Storage and Querying, Data Mining and Analysis, Database Architecture, Database Users and Administrators. Database Design and the E-R Model: Overview of database design process, The Entity-Relationship Model, Constraints, Entity Relationship Diagrams, Entity Relationship Design Issues, Weak Entity Sets.

11 HOURS

UNIT-II

Database design for Banking Enterprise.Relational Model: Structure of Relational Databases, Fundamental Relational Algebra Operations, Additional Relational Algebra Operations, Extended Relational Algebra Operations, Null values, Modification of the Database. Assertion, Triggers and Views.

11 HOURS

UNIT-III

SQL: Data Definition, Basic Structure of SQL queries, Set operations, Aggregate Functions, Null values, Nested sub-queries. . PL/SQL: PL/SQL Overview, Basic Syntax

10 HOURS

UNIT-IV

Normalization: Informal Design Guidelines for Relation Schemas; Functional Dependencies: Definition of FD, Inference rules for FD; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms Properties of Relational Decompositions; Multi valued Dependencies and Fourth Normal Form; Join Dependencies and Fifth Normal Form.

10 HOURS

UNIT-V

Transactions: Introduction to transaction processing; Transaction and system concepts; Desirable Properties of transactions; Transactions and Schedules; Characterizing schedules based on recoverability ; Characterizing schedules based on Serializability; Transaction support in SQL;Concurrency Control Techniques : 2PL techniques for concurrency control; Recovery Concepts: Write-ahead logging, Checkpoints in the system log, Transaction Rollback; Recovery Techniques based on Deferred Update and Immediate update; Shadow paging; The ARIES recovery algorithm;Recovery in multidatabase system.

10 HOURS

Text Books:

1. Silberschatz, Korth and Sudharshan Data base System Concepts, 5th Edition, Mc-GrawHill, 2006.
2. Elmasri and Navathe: Fundamentals of Database Systems, 5th Edition, Addison-Wesley, 2007.
3. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3rd Edition, McGraw-Hill, 2003.

Reference Books:

1. C.J. Date, A. Kannan, S. Swamynatham A Introduction to Database Systems, 8th Edition, Pearson education.

Semester:V

Department:Information Science and Engineering	Course Type:core
Course Title:Formal Languages and Automata Theory	Course Code:14IS55
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have knowledge of set theory and concepts like union, intersection, closure.

Course Outcomes:

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

1. Understand the core concepts of automata theory.
2. Design DFA and NFA for Regular language.
3. Apply the concepts of Regular Expression, Context Free Grammar and Context Free Language for pattern recognition.
4. Design Push-Down Automata for the given Context Free Language.
5. Design Turing Machines for the given Context Free Language.

Teaching Methodology:

1. Blackboard teaching
2. Tutorials
3. Tool based Problematic Assignment (JFLAP)

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluating Tool based Assignments-10 Marks.
3. Rubrics for evaluating Tutorials -10 Marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	3	3	3		2				1	1		1		3
CO3	3	3	3		2				1	1		1		3
CO4	2	3	3		2				1	1		1		3
CO5	2	3	3		2				1	1		1		3
14IS55	3	3	3		1				1	1		1		3

Course Content

UNIT1

Introduction to Finite Automata: Introduction to Finite Automata; The central concepts of Automata theory; Deterministic finite automata; Nondeterministic finite automata ,An application of finite automata; Finite automata with Epsilon transitions; Regular expressions

11 HOURS

UNIT-II

Regular Expressions and Regular Languages: Finite Automata and Regular Expressions; Applications of Regular Expressions. Regular languages; Proving languages not to be regular languages; Closure properties of regular languages; Decision properties of regular languages; Equivalence and minimization of automata.

10 HOURS

UNIT-III

Context-Free Grammars and Languages, Push down Automata: Context -free grammars; Parse trees; Applications; Ambiguity in grammars and Languages. Definition of the Pushdown automata; The languages of a PDA

11 HOURS

UNIT-IV

Pushdown Automata, Properties of Context-Free Languages: Equivalence of PDA's and CFG's; Deterministic Pushdown Automata. Normal forms for CFGs; The pumping lemma for CFGs; Closure properties of CFLs

11 HOURS

UNIT-V

Turing Machine and Undecidability: The Turing machine, Programming techniques for Turing Machines; Extensions to the basic Turing Machines; A Language that is not recursively enumerable; An Undecidable problem that is RE; Posts Correspondence problem; Advanced Topics; Regulated rewriting L systems; Grammar Systems; New paradigms of Computing; DNA Computing; Membrane Computing

09 HOURS

Text Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey Ullman Introduction to Automata Theory, Languages and Computation, 3rd Edition, Pearson education, 2007.
2. Introduction to Formal Languages, Automata Theory and Computation by Kamala Krithivasan, Rama R, 2009.

Reference Books:

1. Raymond Greenlaw, H.James Hoover Fundamentals of the Theory of Computation, Principles and Practice, Morgan Kaufmann, 1998.
2. John C Martin: Introduction to Languages and Automata Theory 3rd Edition, Tata McGrawHill, 2007.
3. Daniel I.A. Cohen Introduction to Computer Theory, 2ndEdition, John Wiley and Sons, 2004.
4. Thomas A. Sudkamp. An Introduction to the Theory of Computer Science, Languages and Machines, 3rd Edition, Pearson Education, 2006.

Semester:V

Department:Information Science and Engineering	Course Type: Elective Core
Course Title:C# and .Net Programming	Course Code:14ISE561
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Student needs good programming skills in C, C++.

Course Outcomes:

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

1. Understand the basic concepts of.NET framework.
2. Design a system using Visual Studio .NET development tools for different applications .
3. Apply features of Master Nodefor solving the given problem.
4. Apply Exception Handling and garbage collection concepts.

Teaching Methodology:

1. Blackboard teaching
2. Power Point presentations
3. Course Project

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluating Course Project-20 marks.
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	2	2	2		1			1	1	1		1		2
CO3	2	2	2		1			1	1	1		1		2
CO4	2	2	2		1			1	1	1		1		2
14ISE561	3	2	2		1			1	1	1		1		3

Course Content

UNIT1

The Philosophy Of .Net:Understanding the Previous State of Affairs, The .NET Solution, TheBuilding Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries,What C# Brings to the Table, An Overview of .NET Binaries (aka Assemblies), the Role of the CommonIntermediate Language, The Role of .NET Type Metadata, The Role of the Assembly Manifest, CompilingCIL to Platform -Specific Instructions, Understanding theCommon Type System, Intrinsic CTS Data Types,Understanding the Common Languages Specification, Understanding the Common Language Runtime Atour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime. Building C# Applications: The Role of the Command Line Compiler (ese.exe), Building C# Application using csc.exe Working with csc.exe Response Files, Generating Bug Reports, Remaining C# Compiler Options, The Command Line Debugger (cordbg.exe) Using the, Visual Studio .NET IDE, Other Key Aspects of the VS.NET IDE, C# "Pre-processor:" Directives, An Interesting Aside: The System. Environment Class.

10 HOURS

UNIT-II

C# Language Fundamentals:The Anatomy of a Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System, Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types, Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C #,String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces.

12 HOURS

UNIT-III

Advanced C# type construction techniques:Building a custom indexer, Internal representation of type indexers, understanding operator overloading, overloading binary operators, overloading unary operators, overloading comparison operators, the internal representation of overloaded operators, interacting with overloaded operators from overloaded operator, the advanced keywords of C#, C# pre-processor directives.

10 HOURS

UNIT-IV

Object- Oriented Programming With C#:Forms Defining of the C# Class, Definition the "Default Public Interface" of a Type, Recapping the Pillars of OOP, The First Pillars: C#'s Encapsulation services, Pseudo- Encapsulation, Creating Read-Only Fields, The Second Pillar: C#'s Inheritance Supports, keeping Family Secrets: The "Protected" Keyword, Nested Type Definitions, The Third pillar: C #'s Polymorphic Support, Casting Between. EXCEPTIONS AND OBJECT LIFETIME: Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handing, the System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System - Level Exception (System. System Exception),Custom Application-Level Exception

10 HOURS

UNIT-V

Handling Multiple Exception: The Family Block, the Last Chance Exception Dynamically IdentifyingApplication - and System Level Exception. Debugging System Exception Using VS. NET, UnderstandingObject Lifetime, the CIT of "new, The Basics of Garbage Collection" Finalization Type, The FinalizationProcess, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.Implementing an interface in C#, understanding call-back interfaces, understanding the .Net delegate type,defining a delegate in C#

10 HOURS

Text Books:

1. Pro C# with .NET 3.0 -Andrew Troelsen, Special Edition, Dream tech Press, India, 2007.
2. Programming in C# - E.Balagurusamy, 51h Reprint, Tata McGraw Hill, 2004.

Reference Books:

1. Inside C# - Tom Archer, WP Publishers, 200 I.C#:The Complete Reference - Herbert Schildt, Tata McGraw Hill, 2004

Semester:V

Department:Information Science and Engineering	Course Type:Elective Core
Course Title:Python for Data Science	Course Code:14ISE563
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Student is required to have basic knowledge about Logic Building.

Course Outcomes:

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

1. Understand programming concepts of python.
2. Apply appropriate data types/ data structures for the given problem.
3. Apply numpy and pandas for Data Analysis.
4. Develop solution for the given problem.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations (if needed)
3. Programming Assignment
4. Summarization by the student at the end of the session.
5. Certification Based Learning.

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluating Programming Assignments for 10 Marks .
3. Certification for 10 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	2	2	3		3				2	1		1		2
CO3	2	2	2	1	3				2	1		1		2
CO4	2	3	3	1	3				2	1		1		2
14ISE563	3	2	3	1	3				2	1		1		3

Course Content

UNIT1

Introducing Python Object Types: The Python Conceptual Hierarchy, Why Use Built-in Types?, Pythons Core Data Types, Numbers, Strings, Sequence Operations, Immutability, Type-Specific Methods, Getting Help, Other Ways to Code Strings, Unicode Strings, Pattern Matching, Lists, Sequence Operations, Type-Specific Operations, Bounds Checking, Nesting, Comprehensions, Dictionaries, Mapping Operations, Nesting Revisited, Missing Keys: if Tests, Sorting Keys: for Loops, Iteration and Optimization, Tuples, why Tuples. Files.

11 HOURS

UNIT-II

Numeric Type Basics, Numeric Literals, Built-in Numeric Tools, Python Expression Operators, Numbers in Action, Variables and Basic Expressions, Numeric Display Formats, Comparisons: Normal and Chained, Division: Classic, Floor, and True, Integer Precision, Complex Numbers, Hex, Octal, Binary: Literals and Conversions, Bitwise Operations, Other Built-in Numeric Tools, Other Numeric Types, Decimal Type, Fraction Type, Sets, Booleans The Dynamic Typing Interlude: The Case of the Missing Declaration Statements Variables, Objects, and References, Types Live with Objects, Not Variables, Objects Are Garbage-Collected, Shared References, Shared References and In-Place Changes, Shared References and Equality, String Bsics and String Literals, Strings in action, String methods, String formatting expressions, String formatting method calls, why the format method?

10 HOURS

UNIT-III

Lists and Dictionaries: Lists, Lists in Action, Basic List Operations, List Iteration and Comprehensions, Indexing, Slicing, and Matrixes, Changing Lists in Place, Dictionaries, Dictionaries in Action, Basic Dictionary Operations, Changing Dictionaries in Place, More Dictionary Methods, Tuples, Tuples in Action, Why Lists and Tuples?, Records Revisited: Named Tuples, Files, Opening Files, Using Files, Files in Action, Text and Binary Files: The Short Story, Storing Python Objects in Files: Conversions, Storing Native Python Objects: pickle, Storing Python Objects in JSON Format, Storing Packed Binary Data: struct, File Context Managers, Other File Tools, Core Types Review and Summary, Object Flexibility, References Versus Copies, Comparisons, Equality, and Truth, The Meaning of True and False in Python, Pythons Type Hierarchies, Type Objects, Other Types in Python. Introducing python statements: Conceptual hierarchy, pythons statements, A tale of two Ifs, what python Adds, what python Removes, why indentation syntax, a few special cases, Interactive loops.

11 HOURS

UNIT-IV

NumPy Basics: Arrays and Vectorized Computation ,The NumPyndarray: A Multidimensional Array Object, Universal Functions: Fast Element-wise Array Functions Operations between Arrays and Scalars, Basic Indexing and Slicing, Boolean Indexing, Fancy Indexing, Transposing Arrays and Swapping Axes, Universal Functions: Fast Element-wise Array Functions, Data Processing Using Arrays, Expressing Conditional Logic as Array Operations, Mathematical and Statistical Methods, Methods for Boolean Arrays, Sorting, Unique and Other Set Logic, File Input and Output with Arrays, Storing Arrays on Disk in Binary Format, Saving and Loading Text Files, Linear Algebra, Random Number Generation.

10 HOURS

UNIT-V

Introduction to pandas Data Structures: Series, DataFrame, Index Objects, Essential Functionality, Reindexing, Dropping entries from an axis, Indexing, selection, and filtering, Arithmetic and data alignment, Function application and mapping, Sorting and ranking, Axis indexes with duplicate values, Summarizing and Computing Descriptive Statistics, Correlation and Covariance, Unique Values, Value Counts, and Membership, Handling Missing Data, Filtering Out Missing Data, Filling in Missing Data, Hierarchical Indexing, Reordering and Sorting Levels, Summary Statistics by Level, Using a DataFrames Columns

10 HOURS

Text Books:

1. Learning Python: Powerful Object-Oriented Programming: 5th Edition, Lutz M, OReilly 2013.
2. Python for Data Analysis, W Mckinney, OReilly, 2nd Edition, 2017.

Reference Books:

1. Data Science from Scratch: First Principles with Python, Joel Grus, OReilly.
2. Python Data Science Essentials,LucaMassaron and Alberto Boschetti, PACKT, OpenSource.
3. Introduction to Python:
   https://www.edx.org/course/introduction-to-python-absolute-beginner-0
4. Python for Data Science: https://www.edx.org/course/python-for-data-science

Semester:V

Department:Information Science and Engineering	Course Type:Elective Core
Course Title: Computer Graphics with OpenGL and CUDA	Course Code:14ISE564
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Computer Concepts and C Programming
- Object Oriented Programming

Course Outcomes:

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

1. Understand the concepts of 2D and 3D transformations, projection and viewing using OpenGL.
2. Apply the knowledge of geometrical transformations and projection matrix for handling multiple objects .
3. Apply the concepts of clipping algorithms and graphics pipeline in solving the given problems.
4. Design geometrical objects using GPU architecture with CUDA programming.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations (if needed)
3. Course Project

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluating course project-20Marks
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	2	2	3						1	1	1	1		2
CO3	2	3	3	1	2				1	1	1	1		2
CO4	2	3	2	1	2				1	1	1	1		2
14ISE564	3	3	2	1	2				1	1	1	1		3

Course Content

UNIT1

Overview: Computer Graphics and Open GL Graphs and Charts, Computer-Aided Design, Virtual-Reality Environments, Data Visualizations, Education and Training, Computer Art. Entertainment, Image Processing, Graphics User Interfaces. Video Display Devices, Raster Scan Systems, Graphic Workstations and Viewing Systems, Input Devices, Hard copy devices, Graphics Networks, Graphics on the Internet, Graphics software, Introduction of Open GL; Coordinate Reference Frames, Specifying a tow-Dimensional World-Coordinate Reference Frame in Open GL, OpenGL Point functions, OpenGL line function, Line drawing Algorithm Circle generating algorithms.

12 HOURS

UNIT-II

Open GL Primitives and Attributes Fill-Area Primitives, Polygon Fill Areas, OpenGL Polygon Fill, Area functions, OpenGL Vertex arrays, Pixel Array Primitives, OpenGL Pixel array functions, Character Primitives, OpenGL Character Functions, Open GL Display-Window Reshape Functions; Color and gray scale, OpenGL color functions, Point Attributes, Line Attributes, Curve Attributes, Open GL Point Attribute Functions, Open GL Line Attribute Functions, Fill-Area Attributes

10 HOURS

UNIT-III

Geometric Transformations Basic Two-dimensional Geometric Transformations, Inverse Transformations, Two Dimensional Composite Transformations, Other Two Dimensional transformations, Geometric transformations in Three dimensional Space, Three dimensional Translation, Three dimensional Rotation, Three dimensional Translation, Other Three dimensional Translation, Open GL Geometric-Transformation Functions

12 HOURS

UNIT-IV

Viewing and Interaction The Two-Dimensional Viewing Pipeline, The clipping window, Normalization and view port transformation., OpenGL Two-Dimensional Viewing functions, Clipping algorithms, Two-Dimensional point clipping., Two-Dimensional line clipping; The Three-dimensional Viewing pipeline, Graphical input data, Logical classifications of input Devices, Open GL menu functions

10 HOURS

UNIT-V

GPU Programming Overview of GPU Architecture, GPU Hardware: Streaming Multiprocessors, Kernel, Thread Blocks. Comparison of GPU and CPU architectures, GPU Programming Model (CUDA). Implementation of basic geometrical objects using GPU programming.

08 HOURS

Text Books:

1. Computer Graphics with OpenGL, 3/E Donald D Hearn and M. Pauline Baker, Publisher: Prentice Hall, 3rd edition, 2004.

Reference Books:

1. OpenGL Programming Guide, VI edition, Jackie Neider, Tom Davis, Mason Woo. Shreiner, Addison-Wesley Publishing Company
2. Interactive Computer Graphics A Top-Down Approach with OpenGL -Edward Angel, 5th Edition, Addison-Wesley, 2008.

semester:V

Department:Information Science and Engineering	Course Type:core
Course Title:Database management systems Lab	Course Code:14ISL57
L-T-P: 0-0-3	Credits: 1.5
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have basic knowledge of files, database.

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

1. Create database with different types of integrity constraints.
2. Develop SQL queries to manage relational database.
3. Access data from the database using PL/SQL blockse
4. Design forms using suitable front end tools.

Teaching Methodology:

1. Black board teaching
2. Course Project

Assessment Methods:

1. Rubrics for evaluation of course project -20 marks
2. Rubrics for evaluating laboratory experiments - 10 marks
3. MSE for 20 marks.
4. SEE examination will be evaluated for 50 marks

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	1	3	3	2			1	1	1	1	1	1		
CO2	1	2	3	2			1	1	1	1	1	1		3
CO3	1	3	2	2				1	1	1	1	1		3
CO4	1		2	2	2			1	1	1	1	1		3
14ISL57		3	2		3		1							3

Course Content

PART-A

1. To write the ER-Diagram, Schema Diagram, Basic table creation with proper interdependencies, Insertion of values,and quires for all of the following questions. A. Consider the following schema for a Library Database: BOOK (Book_id, Title, Publisher_Name, Pub_Year) BOOK_AUTHORS (Book_id, Author_Name) PUBLISHER (Name, Address, Phone) BOOK_COPIES (Book_id, Branch_id, No-of_Copies) BOOK_LENDING (Book_id, Branch_id, Card_No, Date_Out, Due_Date) LIBRARY_BRANCH (Branch_id, Branch_Name, Address) Write SQL queries to
   - Retrieve details of all books in the library id, title, name of publisher, authors, number of copies in each branch, etc.
   - Get the particulars of borrowers who have borrowed more than 3 books, but from Jan 2017 to Jun 2017.
   - Delete a book in BOOK table. Update the contents of other tables to reflect this data manipulation operation.
   - Partition the BOOK table based on year of publication. Demonstrate its working with a simple query.
   - Create a view of all books and its number of copies that are currently available in the Library.
2. Consider the following schema for Order Database: SALESMAN (Salesman_id, Name, City, Commission) CUSTOMER (Customer_id, Cust_Name, City, Grade, Salesman_id) ORDERS (Ord_No, Purchase_Amt, Ord_Date, Customer_id, Salesman_id) Write SQL queries to
   - Count the customers with grades above Bangalores average.
   - Find the name and numbers of all salesmen who had more than one customer.
   - List all salesmen and indicate those who have and dont have customers in their cities (Use UNION operation.)
   - Create a view that finds the salesman who has the customer with the highest order of a day.
   - Demonstrate the DELETE operation by removing salesman with id 1000. All his orders must also be deleted
3. Consider the schema for Movie Database: ACTOR (Act_id, Act_Name, Act_Gender) DIRECTOR (Dir_id, Dir_Name, Dir_Phone) MOVIES (Mov_id, Mov_Title, Mov_Year, Mov_Lang, Dir_id) MOVIE_CAST (Act_id, Mov_id, Role) RATING (Mov_id, Rev_Stars)

   Write SQL queries to
   - List the titles of all movies directed by Hitchcock.
   - Find the movie names where one or more actors acted in two or more movies.
   - List all actors who acted in a movie before 2000 and also in a movie after 2015 (use JOIN operation).
   - Find the title of movies and number of stars for each movie that has at least one rating and find the highest number of stars that movie received. Sort the result by movie title.
   - Update rating of all movies directed by Steven Spielberg to 5.
4. Consider the schema for College Database: STUDENT (USN, SName, Address, Phone, Gender) SEMSEC (SSID, Sem, Sec) CLASS (USN, SSID) SUBJECT (Subcode, Title, Sem, Credits) IAMARKS (USN, Subcode, SSID, Test1, Test2, Test3, FinalIA)

   Write SQL queries to
   - List all the student details studying in fourth semester C section.
   - Compute the total number of male and female students in each semester and in each section.
   - Create a view of Test1 marks of student USN 1BI15CS101 in all subjects.
   - Calculate the Final IA (average of best two test marks) and update the corresponding table for all students.

- Categorize students based on the following criterion: If FinalIA = 17 to 20 then CAT = Outstanding If FinalIA = 12 to 16 then CAT = Average If FinalIA¡ 12 then CAT = Weak Give these details only for 8th semester A, B, and C section students.

5. Consider the schema for Company Database: EMPLOYEE (SSN, Name, Address, Sex, Salary, SuperSSN, DNo) DEPARTMENT (DNo, DName, MgrSSN, MgrStartDate) DLOCATION (DNo,DLoc) PROJECT (PNo, PName, PLocation, DNo) WORKS_ON (SSN , PNo , Hours) Write SQL queries to

   - Make a list of all project numbers for projects that involve an employee whose last name is Scott, either as a worker or as a manager of the department that controls the project.
   - Show the resulting salaries if every employee working on the IoT project is given a 10 percent raise.
   - Find the sum of the salaries of all employees of the Accounts department, as well as the maximum salary, the minimum salary, and the average salary in this department
   - Retrieve the name of each employee who works on all the projects controlled by department number 5 (use NOT EXISTS operator). For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than Rs. 6,00,000.

semester:V

Department:Information Science and Engineering	Course Type:core
Course Title:System Programming Lab	Course Code:14ISL58
L-T-P: 0-0-3	Credits: 1.5
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have basic knowledge of UNIX commands, C programming.

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

1. Develop program to query POSIX defined configuration limits.
2. Design and implement real world application modules using file APIs.
3. Design and implement the real world application module which involves process control.
4. Design and implement solution for a given problem using signal APIs.
5. Design and develop applications to demonstrate inter-process communication using IPC techniques.

Teaching Methodology:

1. Blackboard Teaching

Assessment Methods:

1. Rubrics for evaluating Viva-Voce 10 Marks.
2. MSE for 20 marks.
3. Record and observation20 marks.
4. SEE will be evaluated for 50 marks

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	2	2	3	1									2	
CO2	2	2	3	2									2	
CO3	2	2	3	2									2	
CO4					3							1		
CO5	3	2	3	2	3							1		
14ISL58	3	2	3	2	3							1		

Course Content

1. Write a C/C++ POSIX compliant program to check the following limits: (i) No. of clock ticks (ii) Max. no. of child processes (iii) Max. path length (iv) Max. no. of characters in a file name (v) Max. no. of open files/ process.
2. Write a C/C++ POSIX compliant program that prints the POSIX defined configuration options supported on any given system using feature test macros.
3. Write your own version of cat to confirm to the POSIX specification of cat, you must implement all file I/O using the system calls open, close, read, write. You may use perror for printing error messages.
4. Write a C/C++ program to implement system() function.
5. Write a C/C++ Program to Implement and avoid race condition.
6. Write a C/C++ to implement real time clock interval timer using ALARM API
7. Write a C/C program to implement built-in functions.(i).rename(ii).alaram
8. Develop a server capable of receiving text messages from clients .The server should print these text messages on standard output, but should not print any other messages such as debug information. From the server perspective, a message corresponds to the data received from a Particular client during a communication session with that client.
9. Implement a message queue to pass an integer input by the user from a parent to a forked child by converting an integer to a string and writing that string to the message buffer with the parent and reading it as the child and converting it back to an integer and then printing it to the screen as the child.
10. Develop and implement a product which will sit on the shelf: this will be a shared memory segment that contains an integer, a count of the items on the shelf .This integer may never drop below 0 or raise above 5.Producer creates the shared memory segment and sets the value of count to 5.The producer may add one item to the shelf at a time and must report to STDOUT every time another item is added as well as the current shelf count. Consumer will remove one item from the shelf at a time, provided the item count has not dropped below zero. The consumer will decrement the counter and report the new value to STDOUT.
11. Write a C/C++ program to send data from parent to child over a pipe.

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title: Parallel computing	Course Code:14IS71
L-T-P: 4-1-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- C/C++ programming
- Linux Development Environment
- Computer Organization & Architecture

Course Outcomes:

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

1. Understand the architecture of parallel computers and contemporary parallel computing paradigms.
2. Analyze the performance of the parallel platforms.
3. Use PThreads /OpenMP/ MPI programming models to build parallel applications.
4. Design the parallel solution for the given problem by identifying the hotspot.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Tutorial.
4. Open Book Test
5. Programming Assignment

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Open Book Test -10 Marks
3. Rubrics for evaluating Programming Assignment -10 Marks.
4. SEE for 100 marks will be evaluated -50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3						2						3	
CO2	2	3					2						3	
CO3	2	2	3		3				1	1		1		3
CO4	2	2	3				2		1	1		1		3
14IS71	3	2	2		2		2		1	1		1	2	2

Course Content

UNIT1
Introduction: Motivating Parallelism, Scope of Parallel Computing, Implicit Parallelism, Limitations of Memory System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Routing Mechanisms for Interconnection Networks.

10 HOURS

UNIT-II
Analytical Modelling: Scalability of Parallel Systems, Asymptotic Analysis of Parallel Program, Other Scalability Metrics, Programming Using the Message-Passing Paradigm: Principles of Message-Passing Programming, building blocks, MPI, Topologies and Embedding, Overlapping Communication with Computation, collective Communication and Computation Operations, Groups and Communicator.

12 HOURS

UNIT-III
Basic Communication Operations: One-to-All Broadcast, All-to-One Reduction, All-to-All Broadcast and Reduction, All-Reduce, Scatter and Gather, Analytical Modelling: Sources of Overhead in Parallel Computing, Performance Metrics for Parallel Systems, Effect of Granularity on Performance

10 HOURS

UNIT-IV
ProgrammingSharedAddressSpacePlatformsThreadBasics,WhyThreads?ThePOSIXThreadAPI,Creation &Termination,and Synchronization Primitives in Pthreads, Controlling Thread and Synchronization Attributes, Thread Cancellation, Composite Synchronization Constructs, OpenMP: Specifying concurrent tasks, "for" directive, Assigning iterations to threads, "section" directive, merging directives, Nesting directives, Synchronization constructs in OpenMP, Data Handling in OpenMP, OpenMP library functions

12 HOURS

UNIT-V
Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models.

08 HOURS

Text Books:

1. AnanthaGrama, Anshul Gupta, George karypis, Vipin Kumar. Introduction to Parallel Computing by Addison - Wesley ,2nd Edition, 2003

Reference Books:

1. J.Ja'Ja': An Introduction to Parallel Algorithms, Addison-Wesley, 1992, ISBN 0-201-54856-9
2. T. Leighton: Introduction to parallel algorithms and architectures. MorganKaufmannPubl.,1992, ISBN 1-55860-117-1
3. https://nptel.ac.in/courses/106102114/

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title:Software Project Management	Course Code:14IS72
L-T-P: 3-0-0	Credits: 03
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Software concepts
- Programming languages such as C/C++ and software engineering

Course Outcomes:

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

1. Understand software project management issues and learn project planning.
2. Analyze appropriate software development process model and estimate the cost of a project.
3. Analyze risks and establish risk management plans for given project.
4. Apply suitable techniques for resources allocation and cost monitoring.
5. Apply the techniques of software configuration management and quality assurance policies for a project.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations (if needed)
3. Case Study

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for Case Study-20marks.
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3	1	1						1	1	2	1		
CO2	2	1	1						1	1	3	1		
CO3	2	1	1						1	1	3	1		
CO4	2	1	1						1	1	2	1		
CO5	2	1	1						1	1	2	1		
14IS72	3	1	1						1	1	3	1		

Course Content

UNIT1
Introduction to software project management: Introduction, Why is software project management important?, What is a project? , Software projects versus other types of project, Contract management and technical project management, Activities covered by software project management, Plans, methods and Methodologies, Some ways of categorizing software projects, Stakeholders, Setting objectives, The business case, Project success and failure, What is management?, Management control. Selection of an appropriate project approach: Introduction , Build or buy?, Choosing methodologies and Technologies, Choice of process models, Structure versus speed of delivery, The waterfall model, The spiral model, Software prototyping, Other ways of categorizing, Prototypes, Incremental delivery, Agile methods, Atern/Dynamic Systems, Development Method, Extreme programming (XP), Managing iterative processes, Selecting the most appropriate process model.

07 HOURS

UNIT-II
Project Evaluation A business case , project portfolio management, Evaluation of individual projects, Cost-benefit evaluation techniques .Software effort estimation: Introduction, Where are estimates done? , Problems with over- and underestimates, the basis for software estimating, Software effort estimation techniques, bottom -up estimating, the top-down approach and parametric models, Expert judgment, estimating by analogy. Activity planning: Introduction, The objectives of activity planning, When to plan, Project schedules, Projects and activities, Sequencing and scheduling activities, Network planning models, Formulating a network model, Adding the time dimension, The forward pass, The backward pass, Identifying the critical path, Activity float, Shortening the project duration, Identifying critical activities, Activity-on-arrow networks.

08 HOURS

UNIT-III
Risk management: Introduction, Risk, Categories of risk, A framework for dealing with risk, Risk identification, Risk assessment, Risk planning, Risk management, Evaluating risks to the schedule, Applying the PERT technique, Monte Carlo simulation, Critical chain concepts. Resource allocation: Introduction, The nature of resources, Identifying resource requirements, Scheduling resources, Creating critical paths, Counting the cost, Being specific, Publishing the resource schedule, Cost schedules, The scheduling sequence.

08 HOURS

UNIT-IV
Monitoring and control: Introduction, Creating the framework, Collecting the data, Visualizing progress, Cost monitoring, Earned value analysis, Prioritizing monitoring, Getting the project back to target, Change control. Quality management: Quality concepts, software quality assurance, software reviews, formal technical reviews, formal approaches to software quality assurance, statistical software quality assurance, software reliability, The ISO 9000 quality standards, The SQA plan.

08 HOURS

UNIT-V
Software configuration management: SCM scenario, elements of configuration management system, baselines, software configuration items, The SCM repository- the role of the repository, general features and content, SCM features, the SCM process- identification of objects in the software configuration, version control, change control, configuration audit, status reporting, configuration management for WebApps- dominant issues, WebApp configuration objects, content management, change management, version control, auditing and reporting. Reengineering: Business process reengineering, software reengineering, reverse engineering, restructuring, forward engineering, the economics of reengineering.

08 HOURS

Text Books:

1. Bob Hughes and Mike Cotterell, Software project management, 5th edition, McGraw-Hill Higher Education, 2010, ISBN-13:978-0-0-070653.
2. Roger S Pressman, Software Engineering: A Practitioner's Approach, 7thedition , McGraw-Hill Higher Education, ISBN:0072496681.

Reference Books:

1. Walker Royce, Software project management- A unified framework, 7th edition, 2012 by Pearson education, ISBN 978-81-7758-378-6.
2. Bob Hughes and Mike Cotterell, Software Project Management, 2nd edition, McGraw-Hill Higher Education

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title:Venture Process Management and IPR	Course Code:14ISH73
L-T-P: 3-0-0	Credits: 03
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Course Outcomes:

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

1. Understand the importance of entrepreneurship and its role in economic development of the Country.
2. Establish Vision and Mission of a venture.
3. Identify venture strategies, business model, risk and Uncertainty of business strategy .
4. Analyse Innovation Strategies, finance building and marketing the venture.
5. Understand the concept of IP rights.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Case Studies

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for Case Study evaluation-20marks
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	1							3						
CO2	1	2					1	3	1	1	1	1		
CO3	1	2					1	3	1	1	1	1		
CO4	1	2					1	3	1	1	1	1		
CO5	1							3						
14ISH73	3	2					1	3	1	1	1	1		

Course Content

UNIT1
Economic Growth and the Technology-The Entrepreneurs Challenge, The Entrepreneur ,Economics and the Firm , Creative Destruction , Innovation and Technology , Opportunity and the Concept , Opportunity Identification , Trends and Convergence Opportunity Evaluation , The Concept Summary.

07 HOURS

UNIT-II
Vision and the Business Model, The Mission Statement, The Value Proposition, The Business Model, Business Model Innovation in Challenging Markets, Core Competencies, Sustainable Competitive Advantage, Competitive Strategy, Venture Strategy, The Industry and Context for a Firm, Strengths and OpportunitiesSWOT

08 HOURS

UNIT-III
Innovation Strategies, First Movers Versus Followers, Imitation, Creativity and Invention, Types and Sources of Innovation, Technology and Innovation Strategy, New Technology Ventures, Risk and Return, Risk and Uncertainty, Scale and Scope, Network Effects and Increasing Returns.

08 HOURS

UNIT-IV
Risk Versus Return, Managing Risk, The Business Plan , Creating a New Business ,The New Venture Story , The Business Plan Types of Ventures , Independent Versus Corporate Ventures , Non profit and Social Ventures , Family-Owned Businesses and Franchising ,Financing and Building The Venture- Profit and Harvest ,The Revenue Model ,The Cost Model ,The Profit Model

08 HOURS

UNIT-V
Legal Formation and Intellectual Property ,Legal Form of the Firm ,Company Name ,Intellectual Property , Trade Secrets , Patents , Trademarks , Copyrights , Licensing ,Detailed functional Planning Forthe Venture- The Marketing and Sales Plan , Marketing , Marketing Objectives and Customer Target Segments.

08 HOURS

Text Books:

1. Technology Ventures: From Idea to Enterprise, 3rd Edition, Dorf, Richard, Byers, Thomas, and Nelson, Andrew,2013. ISBN 978-0073380186.

Reference Books:

1. New Venture Creation, 6th Edition or 5th Edition, Timmons, Jeffry A; ISBN: 0072498404, January 2004.

Semester:VII

Department:Information Science and Engineering	Course Type:Elective Core
Course Title:Big data	Course Code:14ISE741
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Course Outcomes:

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

1. Describe the basic concepts of Big Data and its importance.
2. Apply map-reduce based analytics using Hadoop.
3. Design and apply the queries using MongoDB and Cassandra.
4. Apply Pig and Hive query languages on Hadoop framework for Big Data Analytics.
5. Generate reports by connecting to NoSQL database.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Hands-On Big Data Framework
4. Case Study

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for the evaluation of course project for 20 marks.
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													3
CO2			2	2	2				2	2		2		3
CO3		1	2	2	2				2	2		2		3
CO4		1	2	2	2				2	2		2		3
CO5			2	2	2				2	2		2		3
14ISE741	3	1	2	2	2				2	2		2		3

Course Content

UNIT1
Types of Digital data: Classification of Digital data: Structured Data, Semi Structured Data, Unstructured Data; Introduction to Big Data : Characteristics of Data, Evolution of Big Data, Definition of Big Data, Challenges of Big Data, What is Big Data?, Why Big Data? Traditional Business Intelligence versus Big Data, A Typical Data Warehouse Environment, A Typical Hadoop Environment, Coexistence of Big Data and Data Warehouse Big Data Analytics: Big Data Analytics. Classification of Analytics, Greatest challenges on Big Data, Big Data Analytics importance, Data Science, Terminologies in Big Data.

10 HOURS

UNIT-II
The Big Data Technology Landscape: NoSQL:Types of NoSQL, Advantages of NoSQL, Use of NoSQL in industry, NewSQL, Comparison of SQL, NoSQL and NewSQL. Hadoop: Features, Advantages of Hadoop, Versions of Hadoop, Hadoop ecosystem, Hadoop distributions, Hadoop Vs SQL. Introduction to Hadoop: Why Hadoop? RDBMS Vs Hadoop, Distributed computing challenges, History of Hadoop, Hadoop overview, Usecase of Hadoop, HDFS, Processing data with Hadoop, Managing resources and applications with Hadoop YARN.

10 HOURS

UNIT-III
Introduction to MongoDB: Introduction, Terminologies used in RDBMS and MongoDB, Data types in MongoDB, MongoDB query language: insert, save, update, remove, find, count, limit, sort, skip, arrays, aggregate functions, mapreduce functions, java script programming, cursors in MongoDB, indexes, MongoImport and MongoExport. Introduction to Cassandra: Features of Cassandra, CQL data types, CRUD operations, Collections, Counter, TTL, Alter commands, Import and Export, Querying system table.

10 HOURS

UNIT-IV
Introduction to Map Reduce Programming: Introduction, Mapper, Reducer, Combiner, Partitioner, Searching, sorting, compression Introduction to HIVE: Introduction, HIVE architecture, HIVE data types, HIVE file formats, HIVE query language, RCFile implementation, SerDe, User Defined Functions (UDF)

10 HOURS

UNIT-V
Introduction to PIG: Anatomy of PIG, PIG on Hadoop, PIG philosophy, overview of PIG, Data types in PIG, Running and execution modes of PIG, HDFS commands, Relational operators, Eval function, Complex Data types JasperReport using Jaspersoft: Introduction, Connecting to MongoDB NoSQL database, Connecting to Cassandra NoSQL database.

12 HOURS

Text Books:

1. Seema Acharya, SubhasiniChellappan, Big Data and Analytics, Wiley Publications, 2015.

Reference Books:

1. Chris Eaton,Dirkderooset al., Understanding Big data, McGraw Hill, 2012.
2. Boris lublinsky, Kevin T. Smith, Alexey Yakubovich, Professional Hadoop Solutions, Wiley, ISBN: 9788126551071, 2015.
3. Tom White, HADOOP: The definitive Guide, O Reilly 2012.
4. Norman Matloff, The Art of R Programming: A Tour of Statistical Software Design, No Starch Press, USA, 2011.
5. http://www.bigdatauniversity.com/

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title:Machine Learning	Course Code:14ISE742
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Linear Algebra, Probability & Statistics, Calculus
- Any programming language C++, Python.
- Knowledge on Decision Tree

Course Outcomes:

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

1. Understand the preliminary concepts which form the basis of machine learning.
2. Understand how labelled can be used to construct supervised learning models such as regression and classification.
3. Analyze unsupervised learning application towards extracting clusters and predictive models.
4. Apply ensemble of learning techniques for obtaining optimal hypothesis.
5. Apply genetic algorithm for search space optimization and evolution/analytical learning.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Executable Codes/ Live Demonstration
4. Programming Assignment

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Online Certification (10M)
3. Rubrics for evaluating Programming Assignment-10 Marks
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	3													2
CO3	2	3												2
CO4	2	2	3	1	1				1	1				2
CO5	2	2	3	1	1				1	1				2
14ISE742	3	2	2	1	1				1	1				3

Course Content

UNIT1
Introduction-Learning problems, Designing a learning system, perspectives and issues in Machine Learning. Data Mining Concept Learning Task, Concept Learning as search, Find S, Version space and Candidate Elimination Algorithm.

10 HOURS

UNIT-II
Introduction with logistic regression; Logistic regression. Cost function. Artificial Neural Networks representations, appropriate problems for neural network learning, perceptrons, multilayer networks and the backpropagation algorithm, K-means clustering.

12 HOURS

UNIT-III
Bayesian Learning-Bayes theorem, concept learning, maximum likelihood, Bayes optimal classifier, Gibbs algorithm, nave Bayes classifier, Bayesian belief networks, the EM algorithm, PCA.

12 HOURS

UNIT-IV
Support Vector machine: Margins-Intuition, Notation, functional and geometric margins, optimal margin classifier, Lagrange duality, kernels, regularization and non-separable case, the SMO algorithm.

10 HOURS

UNIT-V
Genetic Algorithm, hypothesis space search, genetic programming, models of evolution and learning, Analytical Learning- learning with perfect domain theories: PROLOG-EBG

08 HOURS

Text Books:

1. Tom M. Mitchell, "Machine Learning", by McGraw Hill, 2013.
2. Christopher M. Bishop, "Pattern Recognition and Machine Learning", by Springer, 2007.

Reference Books:

1. Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani, "An Introduction to Statistical Learning: with Applications in R", Springer, 2016.
2. Trevor Hastie, Robert Tibshirani and Jerome Friedman, "The Elements of Statistical Learning: Data Mining, Inference, and Prediction", Springer, 2016
3. Andreas Muller, "Introduction to Machine Learning with Python: A Guide for Data Scientists", Shroff/O'Reilly; First edition (2016)
4. https://nptel.ac.in/courses/106106139/
5. Andrew NG's online Course

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title:Android Application Development and Version Control Repository	Course Code:14ISE743
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have completed least one programming language course on Object Oriented Programming (Preferably C++, Java, or C#).

Course Outcomes:

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

1. Understand concepts of Android Application Development Platform and its associated Environment.
2. Design rich User Interface for Android Application based on the user requirement.
3. Develop Android Application using APIs of Database and Sensors .
4. Illustrate the process involved in publishing an Android Application.
5. Use Version Control Repository for project management.

Teaching Methodology:

1. Blackboard teaching
2. Hands-on sessions through live demonstrations and PPTs.
3. Course Project.

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluating Course Project -20Marks.
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1					3							3		1
CO2	2	2	2	1	3	1	1		2	2	2	2	1	1
CO3	3	2	2	1	3	1	1		2	2	2	2	1	1
CO4					3							1	1	1
CO5					3					3	2	2		
14ISE743	3	2	2	1	3	2			2	3	2	2	2	1

Course Content

UNIT1
Mobility and Android: Introduction: Mobility Panorama, Mobile Platforms, App Development Approaches, Android Overview, Getting Started with Android, setting up Development Environment, Saying Hello to Android, Traversing an Android App Project Structure, Logical Components of an Android App, Android Tool Repository, Installing and Running App Devices Learning with an Application, Mobile App Development Challenges, Tenets of a Winning App.

10 HOURS

UNIT-II
Building Blocks, App User Interface, Activity, UI Resources, UI Elements and Events, Interaction among Activities, Fragments, Action Bar, App Functionality - Beyond UI: Threads, Async Task, Service, Notifications, Intents and Intent Resolution, Broadcast Receivers, Telephony and SMS

12 HOURS

UNIT-III
App Data - Persistence and Access: Flat Files, Shared Preferences, Relational Data, Data Sharing Across Apps, Enterprise Data, Location Services and Maps, Google Play Services, Location Services, Maps, Sensors: Sensors in Android, Android Sensor Framework, Motion Sensors, Position Sensors, Environment Sensors.

10 HOURS

UNIT-IV
Moving To Market: Testing Android Apps, Testing Android App Components, App Testing Landscape Overview, Publishing Apps, Groundwork, Configuring, Packaging, Distributing

10 HOURS

UNIT-V
Introducing Git, Installing Git, Ignoring Files, Adding Files, Forking and Cloning, Exploring Git Log, Branching, Developing on Branch, Git Commits and Branches, Git Remotes, Pull vs Push Model.

10 HOURS

Text Books:

1. Composing Mobile Apps: Learn, Explore, Apply using Android, 1st Edition, AnubhavPradhan, Anil V Deshpande, Wiley Publication 2017.
2. Learn Android Studio - Adam Gerber, 1st Edition, Clifton Craig, Apress Publications, 2015.

Reference Books:

1. Google Developer Training, "Android Developer Fundamentals Course Concept Reference, Google Developer Training Team, 2017. https://www.gitbook.com/book/google-developer-training/android-developerfundamentals-course-concepts/details
2. Android Developer Tools Essentials, 1st edition, Mike Wolfson - O'Reilly Media Publications, 2013.
3. https://in.udacity.com/course/new-android-fundamentals–ud851
4. https://in.udacity.com/course/advanced-android-app-development–ud855

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title:Cloud Computing	Course Code:14ISE744
L-T-P: 4-0-0	Credits: 04
Total Contact Hours: 52 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Fundamentals of Operating System, Networking and Unix.

Course Outcomes:

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

1. Understand virtualization mechanisms used in cloud platforms.
2. Understand service oriented architecture of distributed computing.
3. Analyse various cloud computing platforms and paradigms.
4. Apply concepts of ubiquitous cloud and Internet of Things concepts for solving problems.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Case Study
4. Demonstration of cloud platform

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for evaluation of Case study-10 marks.
3. Rubrics for evaluation of Demonstration of cloud platfroms-10 marks.
4. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	2	2	1						2	1	2	1	2	
CO2	1	2	1						2	1	2	1	2	
CO3	1	1	2						2	1	2	1	1	
CO4	2		3			2	3		3		3	1	3	3
14ISE744	2	2	2			1	2		3	1	2	1	3	2

Course Content

UNIT1
Virtual Machines and Virtualization of clusters and Data centers: Implementation Levels of Virtualization, Virtualization Structures/Tools and mechanisms, virtualization of cpu, Memory and I/O devices, Virtual clusters and resource Management, Virtualization for data center automation.

12 HOURS

UNIT-II
Cloud Platform architecture over virtual data centers: Cloud Computing and service models, Data center design and interconnection networks, Architectural design of compute and storage clouds, Public cloud platforms: GAE, AWS and Azure, Inter cloud Resource Management, Cloud security and Trust Management.

10 HOURS

UNIT-III
Service Oriented Architecture for Distributed Computing: Services and Service Oriented Architecture, Message Oriented middleware, Discovery, Registries, Metadata and Databases

10 HOURS

UNIT-IV
Cloud Programming and Software Environments: Features of cloud and grid platforms, Parallel and distributed programming paradigms, programming Support for Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud software Environments

12 HOURS

UNIT-V
Ubiquitous Clouds and the Internet of Things: Cloud Trends in supporting Ubiquitous Computing, Enabling Technologies for the Internet of Things, Innovative Applications of the Internet of Things

08 HOURS

Text Books:

1. Distributed and cloud computing by Kai Hwang, Geoffrey C Fox and Jack J Dongarra ,Elsevier, 2012

Reference Books:

1. Mastering Cloud Computing RajkumarBuyya, Christian Vecchiola, and Thamarai, Selvi McGraw Hill Education.
2. https://onlinecourses.nptel.ac.in/noc17_cs23/preview

Semester:VII

Department:Information Science and Engineering	Course Type:core
Course Title:Fundamentals of Java	Course Code:14ISO752
L-T-P: 3-0-0	Credits: 03
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Knowledge of any Object-oriented Programming concepts is helpful.

Course Outcomes:

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

1. Understand java programming Concepts.
2. Apply knowledge of Packaging and Exception Handling to make complete Java Projects.
3. Apply knowledge of Inheritance, Exception handling Multithreading to build Java applications.
4. Develop a programming solution to solve the given problem

Teaching Methodology:

1. specified sample teaching techniques her,please add your techniques.
2. Blackboard teaching
3. PowerPoint presentations (if needed)
4. Regular review of students by asking questions based on topics covered in the class
5. Program Assignment

Assessment Methods:

1. Three internals, 30 Marks each will be conducted and the average of best of two will be taken
2. Rubrics for evaluating Programming Assignment-20 marks
3. Final examination, of 100 Marks will be conducted and will be evaluated for 50 Marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													
CO2	3	3	2						1			1		
CO3	3	2	3						1			1		
CO4	3	2	3	1					1			1		
subjectcode	3	2	2	1					1			1		

Course Contents

UNIT1

The History and Evolution of Java: The Birth of Modern Programming: C . C++: The Creation of Java . How Java Changed the Internet . Java Applets. Security . Portability .. Javas Magic: The Bytecode . Servlets: Java on the Server Side . . . The Java Buzzwords Object-Oriented Programming. Two Paradigms Abstraction The Three OOP Principles . A First Simple Program Entering the Program . . First Sample Program . Second Short Program Java Is a Strongly Typed Language . The Primitive Types Integers. Floating-Point Types Characters . . Booleans . A Closer Look at Literals . Variables Declaring a Variable . Type Conversion and Casting .. Automatic Type Promotion in Expressions Arrays . One-Dimensional Arrays .. . Multidimensional Arrays Alternative Array Declaration Syntax .

08 HOURS

UNIT-II

Operators Arithmetic Operators The Bitwise Operators Relational Operators Boolean Logical Operators The Assignment Operator . The ? Operator Operator Precedence . Using Parentheses Control Statements . Javas Selection Statements . .Javas Selection Statements .Iteration Statements . Jump Statements .

08 HOURS

UNIT-III

Introducing Classes. Class Fundamentals Declaring Objects Assigning Object Reference Variables . Introducing Methods . Constructors . The this Keyword . Garbage Collection . The finalize( ) Method .A Closer Look at Methods and Classes Overloading Methods Overloading Constructors Using Objects as Parameters . A Closer Look at Argument Passing Returning Objects .Recursion Introducing Access Control . Understanding static . Introducing final

08 HOURS

UNIT-IV

Inheritance Inheritance Basics . Member Access and Inheritance . Using super .Creating a Multilevel Hierarchy When Constructors Are Called Method Overriding Dynamic Method Dispatch . Why Overridden Methods? . Applying Method Overriding Using Abstract Classes Using final with Inheritance Using final to Prevent Overriding Using final to Prevent Inheritance The Object Class. Packages and Interfaces . Packages . Access Protection . . Importing Packages .Interfaces

08 HOURS

UNIT-V

Exception Handling Exception-Handling Fundamentals Exception Types Uncaught Exceptions Using try and catch Displaying a Description of an Exception . . Multiple catch Clauses . Nested try Statements . throw throws. finally . Javas Built-in Exceptions Creating Your Own Exception Subclasses . Chained Exceptions Using Exceptions Multithreaded Programming The Java Thread Model Thread Priorities Synchronization Messaging The Thread Class and the Runnable Interface The Main Thread Creating a Thread . Implementing Runnable Extending Thread Choosing an Approach Creating Multiple Threads Using isAlive( ) and join( ) Thread Priorities

07 HOURS

Text Books:

1. Herbert Schildt: JavaTM: The Complete Reference Java, Eighth edition, Tata McGraw Hill Publications, 2011,

Reference Books:

1. Introduction to JAVA Programming Y.Daniel Liang, 6th Edition, Pearson Education, 2007
2. http://www.nptelvideos.com/java/java_video_lectures_tutorials.php

Semester:VII

Department:Information Science and Engineering	Course Type:Open elective
Course Title:Design and Development of Web Applications	Course Code: 14ISO753
L-T-P: 3-0-0	Credits: 03
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Course Outcomes:

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

1. Understand the concepts of web technology.
2. Design web pages using HTML/XHTML mark-up languages for Real World Scenarios.
3. Apply Cascading Style Sheets and JavaScript to develop the dynamic user interface.
4. Design server side script with PHP to generate the web pages dynamically using the database connectivity.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Regular review of students by asking questions based on topics covered in the class

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Designing a program and execution -20Marks.
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													1
CO2	2	1	2						2	2		1		2
CO3	2	1	2						2	2		1		3
CO4	2	1	2						2	2		1		3
14ISO753	3	1	2						1	1		1		2

Course Content

UNIT1

Fundamentals of web: Internet, The World Wide Web(WWW), Web Browsers and Web Servers, Uniform Resource Locators(URLs), The Hyper Text Transfer Protocol(HTTP), Security, The Web Programmers Toolbox. Introduction to HTML/XHTML: Origins and evolution of HTML and XHTML, Basic Syntax, Standard HTML document structure, Basic text mark-up, Images, Hyper Text Links, Lists, Tables, Forms, HTML5, Syntactic differences between HTML and XHTML

09 HOURS

UNIT-II

Cascading Style Sheets: Introduction, Levels of Style sheets, Style specification formats, Font properties, List Properties, Alignment of Text, Color, The Box Model, Background Images, The ¡spin¿ and ¡div¿ Tags, Conflict Resolution. Basics of JAVA SCRIPT: Overview of JavaScript, Object orientation and Java Script, General Syntactic characteristics, Primitives, Operations and Expressions, Screen output and keyboard input, Control Statements, Object Creation and Modification, Arrays, Functions, An Example, Constructors, Pattern matching by using regular expressions, Another Example, Errors in Scripts.

09 HOURS

UNIT-III

JavaScript and HTML Documents: The JavaScript execution environment, The Document Object Model, Element access in JavaScript, Events and Event Handling, Handling events from Body elements, handling events from Button elements, handling events from Text box and Password elements. Dynamic Documents with JavaScript: Introduction, Positioning Elements, Moving Elements, Element Visibility, Changing Colors and Fonts, Dynamic Content, Stacking Elements, Locating the mouse cursor, reacting to a Mouse Click, Slow movement of Elements, Dragging and Dropping Elements.

09 HOURS

UNIT-IV

Introduction to PHP: Origins and uses of PHP, Overview of PHP, General syntactic characters, Primitives operations and expressions, output, control statements, Arrays, Functions, Pattern matching, Form handling, Cookies, Session Tracking, comparative study of different technologies and its applications.

06 HOURS

UNIT-V

Database Access through the Web: Relational Databases, An Introduction to the Structured Query Language, Architectures for Database Access, The MySQL Database System, Database Access with PHP and MySQL.

06 HOURS

Text Books:

1. Programming the World Wide Web- Robert W. Sebesta, 7thEdition, Pearson Education, 2014.

Reference Books:

1. Internet & World Wide Web How to Program M.Dietel, P.JDeital, A.B.GoldBerg, 3rdEdition, Pearson Education/PHI 2004.

Semester:VII

Department:Information Science and Engineering	Course Type:Open Elective
Course Title:Mobile App Development	Course Code:14ISO754
L-T-P: 3-0-0	Credits: 03
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have completed least one programming language course on computer programming (Preferably C, C++, Java, or C#).

Course Outcomes:

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

1. Understand concepts of Android Application Development Platform and its associated Environment.
2. Design rich User Interface for Android Application based on the user requiremente
3. Develop Android Application using APIs of Database and Sensors . e
4. Illustrate the process involved in publishing an Android Application.

Teaching Methodology:

1. Blackboard teaching
2. Hands-on sessions through live demonstrations and PPTs.
3. Course Project.

Assessment Methods:

1. Three internals, 30 Marks each will be conducted and the average of best of two will be taken
2. Rubrics for evaluating Course Project -20Marks.
3. Final examination, of 100 Marks will be conducted and will be evaluated for 50 Marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1					3							3		
CO2	2	2	2	1	3	1	1		2	2	2	2		
CO3	3	2	2	1	3	1	1		2	2	2	2		
CO4					3					3	2	2		
subjectcode	3	2	2	1	3	2			2	3	2	2		

Course Content

UNIT1
Mobility and Android: Introduction: Mobility Panorama, Mobile Platforms, App Development Approaches, Android Overview, Getting Started with Android, setting up Development Environment, Saying Hello to Android, Traversing an Android App Project Structure, Logical Components of an Android App, Android Tool Repository, Installing and Running App Devices Learning with an Application, Mobile App Development Challenges, Tenets of a Winning App.

08 HOURS

UNIT-II
Building Blocks, App User Interface, Activity, UI Resources, UI Elements and Events, Interaction among Activities, Fragments, Action Bar

08 HOURS

UNIT-III
App Functionality - Beyond UI: Threads, Async Task, Service, Notifications, Intents and Intent Resolution, Broadcast Receivers, Telephony and SMS

08 HOURS

UNIT-IV
App Data - Persistence and Access: Flat Files, Shared Preferences, Relational Data, Data Sharing Across Apps, Enterprise Data, Location Services and Maps, Google Play Services, Location Services, Maps, Sensors: Sensors in Android, Android Sensor Framework

08 HOURS

UNIT-V
Moving To Market: Testing Android Apps, Testing Android App Components, App Testing Landscape Overview, Publishing Apps, Groundwork, Configuring, Packaging, Distributing

07 HOURS

Text Books:

1. Composing Mobile Apps: Learn, Explore, Apply using Android, 1st Edition, AnubhavPradhan, Anil V Deshpande, Wiley Publication 2017.l
2. Learn Android Studio - Adam Gerber, 1st Edition, Clifton Craig, Apress Publications, 2015

Reference Books:

1. Android Developer Tools Essentials, 1st edition, Mike Wolfson - O'Reilly Media Publications, 2013
2. Google Developer Training, "Android Developer Fundamentals Course Concept Reference, Google Developer Training Team, 2017. https://www.gitbook.com/book/google-developer-training/android-developerfundamentals-course-concepts/details (Download pdf file from the above link)
3. https://in.udacity.com/course/new-android-fundamentals–ud851
4. https://in.udacity.com/course/advanced-android-app-development–ud855

Semester:VII

Department:Information Science and Engineering	Course Type:Open Elective
Course Title:Python Programming	Course Code:14ISO755
L-T-P: 3-0-0	Credits: 03
Total Contact Hours: 39 hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Students should have basic knowledge of object oriented programming.

Course Outcomes:

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

1. Understand the fundamentals of python programming language.
2. Design Python programs utilizing arithmetic expressions, repetition, file Input and Output.
3. Demonstrate the use of the built-in data structures in Python.
4. Apply the knowledge of control structures, functions to solve a given problem.
5. Understand the concepts of object-oriented programming as used in Python.

Teaching Methodology:

1. Blackboard teaching
2. PowerPoint presentations
3. Demonstration
4. Programming Assignment

Assessment Methods:

1. Three MSEs of 30 marks of each. The average of best two performances will be considered to award 30 marks.
2. Rubrics for Programming assignments: 20Marks
3. SEE for 100 marks will be evaluated for50 marks.

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3								3	1				2
CO2	2	2	3						3	3				2
CO3	3	3	2						3	3				2
CO4	3	3	2						3	2				2
CO5	2								3	1				2
14ISO755	3	3	3						3	3				2

Course Content

UNIT1

The way of the program: The Python programming language, what is a program?; What is debugging?, Formal and natural languages, The First Program, Variables, expressions and statements: Values and types, variables, variable names and keywords, statements, evaluating expressions, operators and operands, order of operations, operations on strings, composition, comments; Functions:Function calls,type conversion, type coercion, Math functions, composition, adding new functions, definitions and use, flow of execution, parameters and arguments, variables and parameters are local, Stack diagrams, functions with results; Conditionals and recursion: The modulus operator, Boolean expressions, logical operators, conditional execution, alternative execution, chained conditionals, nested conditionals, the return statement, recursion, stack diagrams for recursive functions, infinite recursion, keyboard input.

08 HOURS

UNIT-II

Fruitful functions: Return values, program development, composition, Boolean functions, more recursion, leap of faith, checking types; Iteration: Multiple assignment, the while statement, tables, two-dimensional tables, encapsulation and generalization, local variables, functions; Strings: A compound data type, length, traversal and the for loop, string slices, string comparison, strings are immutable, a find function, looping and counting, the string module, character classification.

07 HOURS

UNIT-III

Lists: List values, accessing elements, list length, list membership, lists and for loops, list operations, list slices, lists are mutable, list deletion, objects and values, aliasing, cloning lists, list parameters, nested lists, matrices, strings and lists; Tuples: Mutability and tuples, tuple assignment, tuples as return values, random numbers, list of random numbers, counting, buckets, a single pass solution; Dictionaries: Operations and methods, Aliasing and copying, sparse matrices, long integers, counting letters.

08 HOURS

UNIT-IV

Files and exceptions: Text files,writing variables, directories, pickling; Exceptions: Exception, the raise statement, custom exception classes, Catching exceptions, Functions, the Zen of exceptions.

07 HOURS

UNIT-V

Classes and Objects: User-defined compound types, attributes, instances as arguments, sameness, rectangles, instances as return values, Objects are mutable, copying; Classes and Functions: Time, pure functions, modifiers, prototype development versus planning, Generalization, algorithms.

06 HOURS

Text Books:

1. Downey, A., Elkner, J., & Meyers, C. (2016). How to think like a computer scientist: learning with python. Green Tea Press, Wellesley, Massachusetts.
2. Magnus Lie Hetland , Beginning Python from Novice To Professional. APress 2nd Edition

Reference Books:

1. Michael Dawson Python programming for absolute beginners.
2. Introduction to Python:
   https://www.edx.org/course/introduction-to-python-absolute-beginner-0

semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Web Technology Lab	Course Code:14ISL76
L-T-P: 0-0-3	Credits:1.5
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

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

1. Understand the fundamental concepts of web programming.
2. Design web pages using HTML/XHTML and Cascading Style Sheets.
3. Develop a dynamic user interface with validation using JavaScript.
4. Develop client and server side scripts using PHP and Ajax for given real world senario.
5. Use Angular JS and Node JS for developing web applications.

Teaching Methodology:

1. Blackboard teaching
2. Hands-on session
3. Tutorial on Lab Programs
4. Course Project

Assessment Methods:

1. Rubrics for evaluating laboratory experiments 20 Marks
2. Rubrics for evaluating course project 20 Marks
3. MSE for 10 marks.
4. SEE examination will be evaluated for 50 marks .

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1	3													2
CO2	2	2	3	1	3	1			2	1	2	2		2
CO3	2	2	3	1	3	1			2	1	2	2		2
CO4	2	2	3	1	3	1			2	1	2	2		2
CO5	2	2	3	1	3	1			2	1	2	2		2
14ISL76	2	2	3	1	3	1			2	1	2	2		2

Course Content

PART-A

1. Design a static webpage for Pet Store / Comics / Flower Shop using HTML. The page should be implemented by using basic layout and tags, such as, div, p, h1, h2, h3, img, etc., and style attribute.
2. Design an online calculator including currency conversion webpage using JavaScript to dynamically provide the results. It should not take any non-number value.
3. Write an HTML page that has one input, which can take multi-line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with a white space and lines are separated with new line character.
4. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (colour, bold and font size).
5. Create a generic sign up page for including parameters like First Name, Last Name, Date of Birth, mobile number and email address and perform form validation on submit using JavaScript.

PART-B

1. Create a generic sign up page for including parameters like First Name, Last Name, Date of Birth, mobile number and email address and insert the values in MySQL Database using PHP.
2. Create display all button for populating the table contents using MySQL database and PHP. Also create a search field to search Student by Name or by Student ID and return the result in table.
3. Create a web application by dynamically populating the list of cities, populating states, populating country based on hierarchy from MySQL database using PHP and AJAX.
4. Develop a simple web application using angular JS and Node JS for employee details.

semester:III

Department:Information Science and Engineering	Course Type:core
Course Title:Distributed Computing Lab	Course Code:14ISL76
L-T-P: 0-0-3	Credits: 1.5
Total Contact Hours: 26hrs	Duration of SEE: 3 hrs
SEE Marks: 50	CIE Marks: 50

Pre-requisites:

- Student needs basic knowledge of operating system

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

1. Design parallel programmes using MPI library and will be able to apply appropriate MPI communication library based on the application needs.
2. Implement parallel applications using Pthread libraries.
3. Design parallel programming by applying openMP directives.
4. Apply map-reduce framework on Hadoop File System for the data driven applications.

Teaching Methodology:

1. Hands-on session and tutorial on Lab Programs
2. Blackboard Teaching

Assessment Methods:

1. MSE for 20 marks.
2. Rubrics for evaluating laboratory experiments for 30 marks
3. SEE examination will be evaluated for 50 marks .

Course Outcome to Programme Outcome Mapping

PO	1	2	3	4	5	6	7	8	9	10	11	12	PSO1	PSO2
CO1		2	3	3	3				3				3	
CO2		2	3	3	3				3				3	
CO3		2	3	3	3				3				3	
CO4		2	3	3	3				3				3	
14ISL76		2	3	3	3				3				3	

Course Content

1. A file includes a list of account numbers and the loans that have been taken by the account holders. Manager need to see the details in descending order of loan amount. Can you give a parallel solution for this using MPI.
2. A middleman company is providing the loan support services to the people. It asks users to enter their interest in the loans and the information is stored in matrix format. Also it has another matrix where in for each bank the interest rate for the loan type is stored. Can you give a solution regarding what will be the total interest rate for each person in each bank? The solution can be implemented using scatter gather and broadcast-reduce methods of MPI. Compare the performance in both the methods and explain with reason.
3. Design and Implement a twoway ring topology with wraparound and apply both forward and backward communication of each process using following types of communications. -Blocking Send and Receive -Non Blocking send and receive Compare the execution time.
4. Design and develop a C program using MPI to create Cartesian Topology of process group.
5. Design and Implement a parallel solution using MPI to find the factorial of any given number.
6. A middleman company is providing the loan support services to the people. It asks users to enter their interest in the loans and the information is stored in matrix format. Also it has another matrix where in for each bank the interest rate for the loan type is stored. Can you give a solution regarding what will be the total interest rate for each person in each bank? Provide a solution for this using FOR DIRECTIVE of OpenMp library.
7. For a given file which contains a novel, a reader wants to know how many times the names Jane and Elizabeth have been mentioned. Write a parallel code using OpenMP to parallelize this using either FOR xdirective or both of them.
8. Design and Implement producer-consumer program to synchronize threads using MUTEX.
9. Design and implement producer-consumer program to synchronize threads using conditional variable.
10. Perform Word Count using map-reduce framework on the data over Hadoop File System.
11. Shopping.com has a huge distributed database regarding previous purchase details of customers stored in HDFS. Use the SPARK framework to predict the next purchase of the customer based on data available.