VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

B. Tech– IIYearI Semester

S. No.	Category	Course Code	Title	L	T	P	Credits
1	BS	23BTHS08T	Discrete Mathematics & Graph Theory	3	0	0	3
2	HS	23BTBA01T	Universal Human Values-Understanding Harmony and Ethical Human Conduct	2	1	0	3
3	ES	23BTEC07T	Digital Logic & Computer Organization	3	0	0	3
4	PC	23BTCS03T	Advanced Data Structures & Algorithm Analysis	3	0	0	3
5	PC	23BTCS04T	Object Oriented Programming Through Java	3	0	0	3
6	PC	23BTCS04P	Advanced Data Structures and Algorithm Analysis Lab	0	0	3	1.5
7	PC	23BTCS05P	Object Oriented Programming Through Java Lab	0	0	3	1.5
8	SC	23BTCS01S	**Skill Oriented Course– I** Python Programming	0	1	2	2
9	MC	23BTCE01A	**Mandatory noncredit course - I** Environmental Science	2	0	0	-
Total				16	2	8	20

B.Tech– IIYearIISemester

S. No.	Category	Course Code	Title	L	T	P	Credits
1	ES	23BTME10T	Optimization Techniques	2	0	0	2
2	BS	23BTHS12T	Probability & Statistics	3	0	0	3
3	PC	23BTCS05T	Operating Systems	3	0	0	3
4	PC	23BTCS06T	Database Management Systems	3	0	0	3
5	PC	23BTCS07T	Software Engineering	3	0	0	3
6	PC	23BTCS06P	Operating Systems Lab	0	0	3	1.5
7	PC	23BTCS07P	Database Management Systems Lab	0	0	3	1.5
8	SC	23BTCS02S	**Skill Oriented Course– II** Full Stack Development –I	0	1	2	2
9	MC	23BTME09T	Design Thinking &Innovation	1	0	2	2
Total				**15**	**1**	**10**	**21**
Mandatory Community Service Project Internship of 08 weeks duration during summer vacation							

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

IIIB.TechISemester

S.No	CourseCode	Title	L	T	P	Credits
1	23BTIT01T	Advanced Java	3	0	0	3
2	23BTCS09T	Computer Networks & Internet Protocols	3	0	0	3
3	23BTCS10T	AutomataTheoryandCompilerDesign	3	0	0	3
4	23BTCS11T	IntroductionToQuantumTechnologiesAnd Applications	3	0	0	3
5	23BTCS12a 23BTCS18b 23BTCS08T 23BTEC11T 23BTIT02T	**ProfessionalElective-I** 1. ObjectOrientedAnalysisandDesign 2. Cyber Security 3. Artificial Intelligence 4. Microprocessors&Microcontrollers 5. Data Warehousing & Data Mining	3	0	0	3
6		**OpenElective-I**	3	0	0	3
7	23BTIT01P	Advanced Java Lab	0	0	3	1.5
8	23BTCS09P	Computer Networks & Internet Protocols Lab	0	0	3	1.5
9	23BTCS02S	**SkillEnhancementcourse** FullStackDevelopment –I	0	1	2	2
10	23BTEC09P	TinkeringLab	0	0	2	1
11	23BTCS13	EvaluationofCommunityService Internship	-	-	-	2
		Total	**18**	**1**	**10**	**26**

OpenElective–I

S.No.	CourseCode	CourseName	Offeredbythe Dept.
1	23BTCE13a	Green Buildings	CIVIL
2	23BTCE13b	ConstructionTechnologyandManagement	
3	23BTEE12T	ElectricalSafetyPracticesandStandards	EEE
4	23BTME15T	SustainableEnergyTechnologies	ME
5	23BTEC13T	ElectronicCircuits	ECE
6	23BTCS14T	QuantumTechnologiesAndApplications	CSE& Allied
7	23BTHS15T	MathematicsforMachineLearningandAI	Mathematics
8	23BTHS16T	MaterialsCharacterizationTechniques	Physics
9	23BTHS17T	ChemistryofEnergySystems	Chemistry
10	23BTHS18T	EnglishforCompetitiveExaminations	Humanities
11	23BTHS19T	EntrepreneurshipandNewVentureCreation	

Note:

1. A student is permitted to register for Honours or a Minor in IV semester after the results of III Semester are declared and students may be allowed to take maximum two subjects per semester pertaining to their Minor from V Semester onwards.
2. A student shall not be permitted to take courses as Open Electives/Minor/Honours with content substantially equivalent to the courses pursued in the student's primary major.
3. A student is permitted to select a Minor program only if the institution is already offering a Major degree program
in that discipline.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

III B.TechIISemester

S.No	CourseCode	Title	L	T	P	Credits
1	23BTCS16T	CloudComputing	3	0	0	3
2	23BTCS17T	Cryptography&NetworkSecurity	3	0	0	3
3	23BTAI02T	MachineLearning	3	0	0	3
4	23BTCS18a 23BTCS21c 23BTCS18c	**ProfessionalElective-II** 1. SoftwareTestingMethodologies 2. AugmentedReality&VirtualReality 3. DevOps 4. EmbeddedSystemsDesign	3	0	0	3
5	23BTCS19a 23BTCS19b 23BTCS19c 23BTIT03T	**ProfessionalElective-III** 1. SoftwareProjectManagement 2. MobileAdhoc Networks 3. NaturalLanguageProcessing 4. Distributed Operating Systems	3	0	0	3
6		**OpenElective–II**	3	0	0	3
7	23BTIT02P	Cloud Computing Lab	0	0	3	1.5
8	23BTCS10P	MachineLearningLab	0	0	3	1.5
9		**SkillEnhancementcourse** Softskills	0	1	2	2
10		**AuditCourse** TechnicalPaperWriting&IPR	2	0	0	-
11		Workshop	0	0	0	0
		Total	**20**	**1**	**08**	**23**
MandatoryIndustryInternshipof6to8weeksdurationduringsummer vacation						

OpenElective–II

S.No.	CourseCode	CourseName	Offeredbythe Dept.
1	23BTCE19a	DisasterManagement	CIVIL
2	23BTCE19b	SustainabilityInEngineeringPractices	
3	23BTEE18T	RenewableEnergySources	EEE
4	23BTME14T	AutomationandRobotics	ME
5	23BTEC20T	DigitalElectronics	ECE
6	23BTHS20T	OptimizationTechniquesforEngineers	Mathematics
7		MathematicalFoundationOfQuantumTechnologies	
8	23BTHS21T	PhysicsOfElectronicMaterialsAndDevices	Physics
9	23BTHS22T	ChemistryOfPolymersAnd Applications	Chemistry
10	23BTHS23T	AcademicWritingandPublicSpeaking	Humanities

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Discrete Mathematics & Graph Theory		L	T	P	C
23BTHS08T	(Common to CSE, CS& IT and CSE(AI))		3	0	0	3
Year	II	Semester	I			

Course Objectives:

Introduce the concepts of mathematical logic and gain knowledge in sets, relations and functions and Solve problems using counting techniques and combinatorics and to introduce generating functions and recurrence relations. Use Graph Theory for solving real world problems.

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

CO1: Apply mathematical logic to solve problems.
CO2: Applythe concepts and perform the operations related to sets, relations and functions. Also identify structures of algebraic nature.
CO3: Apply basic counting techniques to solve combinatorial problems.
CO4: Formulate problems and solve recurrence relations.
CO5: Apply Graph Theory in solving computer science problems.

UNIT I:Mathematical Logic

Introduction, Statements and Notation, Connectives, Well-formed formulas, Tautology, Duality law, Equivalence, Implication, Normal Forms, Functionally complete set of connectives, Inference Theory of Statement Calculus, Predicate Calculus, Inference theory of Predicate Calculus.

UNIT II: Set theory

The Principle of Inclusion- Exclusion, Pigeonhole principle and its application, Functions composition of functions, Inverse Functions, Recursive Functions, Lattices and its properties. Algebraic structures: Algebraic systems-Examples and General Properties, Semi groups and Monoids, groups, sub groups, homomorphism, Isomorphism.

UNIT III: Elementary Combinatorics

Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutations with Constrained Repetitions, Binomial Coefficients, The Binomial and Multinomial Theorems.

UNIT IV: Recurrence Relations

Generating Functions of Sequences, Calculating Coefficients of Generating Functions, Recurrence relations, Solving Recurrence Relations by Substitution and Generatingfunctions, The Method of Characteristic roots, Solutions of Inhomogeneous, Recurrence Relations.

UNIT V:Graphs

Basic Concepts, Isomorphism and Sub graphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs.

Textbooks:

1. J.P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science, Tata McGraw Hill, 2002.
2. Kenneth H. Rosen, Discrete Mathematics and its Applications with Combinatorics and Graph Theory, 7th Edition, McGraw Hill Education (India) Private Limited.

ReferenceBooks:

1. Joe L. Mott, Abraham Kandel and Theodore P.Baker, Discrete Mathematics for Computer Scientists & Mathematicians, 2nd Edition, Pearson Education.
2. NarsinghDeo, Graph Theory with Applications to Engineering and Computer Science.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Universal Human Values- Understanding Harmony and Ethical Human Conduct (Common to All Branches of Engineering)		L	T	P	C
23BTBA01T			2	1	0	3
Year	II	Semester	I			

Course Objectives:

- To help the students appreciate the essential complementary between 'VALUES' and 'SKILLS' to ensure sustained happiness and prosperity which are the core aspirations of all human being.
- To facilitate the development of a Holistic perspective among students towards life and profession as well as towards happiness and prosperity based on a correct understanding of the Human reality and the rest of existence.
- Forming basis of Universal Human Values and movement towards value-based living in a natural way.
  To highlight plausible implications of such a Holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behaviour and mutually enriching interaction with Nature.

Course Outcomes: After completion of the course, students will be able to
CO1: Understand the concepts of value education and human aspirations.
CO2: Understand the harmony in the human beings
CO3: Understand the harmony in the family and society
CO4: Understand the harmony in the nature
CO5: Apply the ethics towards value-based life and profession.

UNIT – I: Introduction to Value Education

Introduction to Value Education (6 lectures and 3 tutorials for practice session)
Lecture 1: Right Understanding, Relationship and Physical Facility (Holistic Development and the Role of Education)
Lecture 2: Understanding Value Education
Tutorial 1: Practice Session PS1 Sharing about Oneself
Lecture 3: self-exploration as the Process for Value Education
Lecture4: Continuous Happiness and Prosperity – the Basic Human Aspirations
Tutorial 2: Practice Session PS2 Exploring Human Consciousness
Lecture 5: Happiness and Prosperity – Current Scenario
Lecture 6: Method to Fulfill the Basic Human Aspirations
Tutorial 3: Practice Session PS3 Exploring Natural Acceptance

UNIT – II: Harmony in the Human Being

Harmony in the Human Being (6 lectures and 3 tutorials for practice session)
Lecture 7: Understanding Human being as the Co-existence of the self and the body.
Lecture 8: Distinguishing between the Needs of the self and the body
Tutorial 4: Practice Session PS4 Exploring the difference of Needs of self and body.
Lecture 9: The body as an Instrument of the self
Lecture 10: Understanding Harmony in the self
Tutorial 5: Practice Session PS5 Exploring Sources of Imagination in the self
Lecture 11: Harmony of the self with the body
Lecture 12: Programme to ensure self-regulation and Health
Tutorial 6: Practice Session PS6 Exploring Harmony of self with the body

UNIT – III: Harmony in the Family and Society

Harmony in the Family and Society (6 lectures and 3 tutorials for practice session)
Lecture 13: Harmony in the Family – the Basic Unit of Human Interaction
Lecture 14: 'Trust' – the Foundational Value in Relationship
Tutorial 7: Practice Session PS7 Exploring the Feeling of Trust
Lecture 15: 'Respect' – as the Right Evaluation
Tutorial 8: Practice Session PS8 Exploring the Feeling of Respect
Lecture 16: Other Feelings, Justice in Human-to-Human Relationship
Lecture 17: Understanding Harmony in the Society
Lecture 18: Vision for the Universal Human Order
Tutorial 9: Practice Session PS9 Exploring Systems to fulfill Human Goal

UNIT – IV: Harmony in the Nature/Existence

Harmony in the Nature/Existence (4 lectures and 2 tutorials for practice session)
Lecture 19: Understanding Harmony in the Nature
Lecture 20: Interconnectedness, self-regulation and Mutual Fulfillments among the Four Orders of Nature
Tutorial 10: Practice Session PS10 Exploring the Four Orders of Nature
Lecture 21: Realizing Existence as Co-existence at All Levels
Lecture 22: The Holistic Perception of Harmony in Existence
Tutorial 11: Practice Session PS11 Exploring Co-existence in Existence

UNIT – V: Implications of the Holistic Understanding

Implications of the Holistic Understanding – a Look at Professional Ethics (6 lectures and 3 tutorials for practice session)
Lecture 23: Natural Acceptance of Human Values
Lecture 24: Definitiveness of (Ethical) Human Conduct
Tutorial 12: Practice Session PS12 Exploring Ethical Human Conduct
Lecture 25: A Basis for Humanistic Education, Humanistic Constitution and Universal Human Order
Lecture 26: Competence in Professional Ethics
Tutorial 13: Practice Session PS13 Exploring Humanistic Models in Education
Lecture 27: Holistic Technologies, Production Systems and Management Models-Typical Case Studies
Lecture 28: Strategies for Transition towards Value-based Life and Profession
Tutorial 14: Practice Session PS14 Exploring Steps of Transition towards Universal Human Order

Practice Sessions for UNIT I – Introduction to Value Education
PS1 Sharing about Oneself
PS2 Exploring Human Consciousness
PS3 Exploring Natural Acceptance

Practice Sessions for UNIT II – Harmony in the Human Being
PS4 Exploring the difference of Needs of self and body
PS5 Exploring Sources of Imagination in the self
PS6 Exploring Harmony of self with the body

Practice Sessions for UNIT III – Harmony in the Family and Society
PS7 Exploring the Feeling of Trust

PS8 Exploring the Feeling of Respect
PS9 Exploring Systems to fulfil Human Goal

Practice Sessions for UNIT IV – Harmony in the Nature (Existence)
PS10 Exploring the Four Orders of Nature
PS11 Exploring Co-existence in Existence

Practice Sessions for UNIT V – Implications of the Holistic Understanding – a Look at Professional Ethics
PS12 Exploring Ethical Human Conduct
PS13 Exploring Humanistic Models in Education
PS14 Exploring Steps of Transition towards Universal Human Order

Textbooks:

- R R Gaur, R Asthana, G P Bagaria, A Foundation Course in Human Values and Professional Ethics, 2nd Revised Edition, Excel Books, New Delhi, 2019. ISBN 978-93-87034-47-1. JeevanVidya: EkParichaya, A Nagaraj, JeevanVidyaPrakashan, Amar kantak, 1999.
- Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
  The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi.

Reference Books:
1. JeevanVidya: EkParichaya, ANagaraj, JeevanVidyaPrakashan, Amarkantak, 1999.
2. Human Values, A.N. Tripathi, New Age Intl. Publishers, New Delhi, 2004.
3. The Story of Stuff (Book).
4. The Story of My Experiments with Truth - by Mohandas Karamchand Gandhi
5. Small is Beautiful - E. F Schumacher.
6. Slow is Beautiful - Cecile Andrews
7. Economy of Permanence - J C Kumarappa
8. Bharat Mein Angreji Raj – PanditSunderlal
9. Rediscovering India - by Dharampal
10.HindSwaraj or Indian Home Rule - by Mohandas K. Gandhi

Online Learning Resources:

- https://fdpsi.aicteindia.org/UHVII%20Class%20Notes%20&%20Handouts/UHV%20Handout%201-Introduction%20to%20Value%20Education.pdf
- https://fdpsi.aicteindia.org/UHVII%20Class%20Notes%20&%20Handouts/UHV%20Handout%20-Harmony%20in%20the%20Human%20Being.pdf
- https://fdpsi.aicteindia.org/UHVII%20Class%20Notes%20&%20Handouts/UHV%20Handout%203-Harmony%20in%20the%20Family.pdf
- https://fdpsi.aicteindia.org/UHV%201%20Teaching%20Material/D3S2%20Respect%20July%2023.pdf
- https://fdpsi.aicteindia.org/UHVII%20Class%20Notes%20&%20Handouts/UHV%20Handou%205-Harmony%20in%20the%20Nature%20and%20Existence.pdf
- https://fdpsi.aicteindia.org/download/FDPTeachingMaterial/3days%20FDPSI%20UHV%20Tehing%20Material/Day203%20Handouts/UHV%203D%20D3S2A%20Und%20Nature Existence.pdf
- https://fdpsi.aicteindia.org/UHV%20II%20Teaching%20Material/UHV%20II%20Lecture%202325%20Ethics%20v1.pdf
- https://www.studocu.com/in/document/kietgroupofinstitutions/universalhumanvalues/chapter5-holistic-understanding-of-harmony-on-professional-ethics/62490385
- https://onlinecourses.swayam2.ac.in/aic22_ge23/preview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Digital Logic & Computer Organization		L	T	P	C
23BTEC07T			3	0	0	3
Year	II	Semester	I			

Course Objectives:

The main objectives of the course is to

- Provide students with a comprehensive understanding of digital logic design principles and computer organization fundamentals
- Describe memory hierarchy concepts
- Explain input/output (I/O) systems and their interaction with the CPU, memory, and peripheral devices

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

CO1: Analyze and implement basic logic functions using logic gates, including universal gates.
CO2: Design sequential circuits including flip-flops, binary counters, registers, shift registers, and ripple counters.
CO3: Apply fundamental concepts of processor organization and perform related arithmetic operations.
CO4: Analyze cache memory structures and performance considerations.
CO5: Analyze processor examples in the context of I/O operations.

UNIT – I:

Data Representation: Binary Numbers,Fixed Point Representation.Floating Point Representation. Number base conversions, Octal and Hexadecimal Numbers, components, Signed binary numbers, Binary codes

Digital Logic Circuits-I: Basic Logic Functions, Logic gates, universal logic gates, Minimization of Logic expressions. K-Map Simplification, Combinational Circuits, Decoders, Multiplexers

UNIT – II:

Digital Logic Circuits-II: Sequential Circuits, Flip-Flops, Binary counters, Registers, Shift Registers, Ripple counters

Basic Structure of Computers: Computer Types, Functional units, Basic operational concepts, Bus structures, Software, Performance, multiprocessors and multi computers, Computer Generations,Von- Neumann Architecture

UNIT – III:

Computer Arithmetic: Addition and Subtraction of Signed Numbers, Design of Fast Adders, Multiplication of Positive Numbers, Signed-operand Multiplication, Fast Multiplication, Integer Division, Floating-Point Numbers and Operations

Processor Organization: Fundamental Concepts, Execution of a Complete Instruction, Multiple-Bus Organization, Hardwired Control and Multi programmed Control

UNIT – IV:

The Memory Organization: Basic Concepts, Semiconductor RAM Memories, Read-Only Memories, Speed, Size and Cost, Cache Memories, Performance Considerations, Virtual Memories, Memory Management Requirements, Secondary Storage

UNIT – V:

Input/Output Organization: Accessing I/O Devices, Interrupts, Processor Examples, Direct Memory Access, Buses, Interface Circuits, Standard I/O Interfaces

Textbooks:

1. Computer Organization, Carl Hamacher, ZvonkoVranesic, SafwatZaky, 6th edition, McGraw Hill
2. Digital Design, 6th Edition, M. Morris Mano, Pearson Education.

Reference Books:

1. Computer Organization and Architecture, William Stallings, 11thEdition, Pearson.
2. Computer Systems Architecture, M.Moris Mano, 3rdEdition, Pearson
3. Computer Organization and Design, David A. Paterson, John L.Hennessy, Elsevier
4. Fundamentals of Logic Design, Roth, 5thEdition, Thomson

Online Learning Resources:

1. https://nptel.ac.in/courses/106/103/106103068/

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Advanced Data Structures & Algorithm Analysis		L	T	P	C
23BTCS03T	(Common to CSE, CS & IT and CSE(AI))		3	0	0	3
Year	II	Semester	I			

Course Objectives:

The main objectives of the course is to

- provide knowledge on advance data structures frequently used in Computer Science domain
- Develop skills in algorithm design techniques popularly used
- Understand the use of various data structures in the algorithm design

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

CO1: Analyze the complexity of algorithms, apply asymptotic notations and apply advanced tree data structures and their operations.

CO2: Apply divide-and-conquer algorithms effectively for solving complex computational problems.

CO3: Develop greedy method and dynamic programming algorithms for solving various real-time applications.

CO4: Illustrate how Backtracking and branch-and-bound algorithms are applied to solve complex computational problems.

CO5: Solve NP Hard and NP Complete problems in graph theory and scheduling domains.

UNIT – I:

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.

AVL Trees – Creation, Insertion, Deletion operations and Applications

Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications

UNIT – II:

Graphs – Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications, Topological Sort.

Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen's matrix multiplication, Convex Hull Problem.

UNIT – III:

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths, General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees

Dynamic Programming: General Method, All pairs shortest paths.

UNIT – IV:

Dynamic Programming-II: 0/1 Knapsack, String Editing, Travelling Salesperson problem

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem

UNIT – V:

Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem

NP Hard and NP Complete Problems: Basic Concepts, Cook's theorem (Without Proof)

NP Hard Graph Problems: Clique Decision Problem (CDP), Traveling Salesperson Decision Problem (TSP)
NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling

Textbooks:

1. Fundamentals of Data Structures in C++, Horowitz, Ellis; Sahni, Sartaj; Mehta, Dinesh, 2nd Edition Universities Press.
2. Fundamentals of Computer Algorithms, Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2nd Edition University Press.

Reference Books:

1. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
2. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill
3. The Art of Computer Programming, Vol.1: Fundamental Algorithms, Donald E Knuth, Addison-Wesley, 1997.
4. Data Structures using C & C++: Langsam, Augenstein&Tanenbaum, Pearson, 1995
5. Algorithms + Data Structures & Programs:, N.Wirth, PHI
6. Fundamentals of Data Structures in C++: Horowitz Sahni& Mehta, Galgottia Pub.
7. Data structures in Java:, Thomas Standish, Pearson Education Asia

Online Learning Resources:

1. https://www.tutorialspoint.com/advanced_data_structures/index.asp
2. http://peterindia.net/Algorithms.html
3. https://www.youtube.com/playlist?list=PLDN4rrl48XKpZkf03iYFl-O29szjTrs_O

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Object Oriented Programming through Java		L	T	P	C
23BTCS04T	(Common to CSE, CS & IT and CSE(AI))		3	0	0	3
Year	II	Semester	I			

Course Objectives:

The learning objectives of this course are to:

- identify Java language components and how they work together in applications
- Learnthe fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- understand how to design applications with threads in Java
- understand how to use Java APIs for program development

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

CO1: Apply fundamental concepts including Java data types, variables, operators, and control statements to develop robust and efficient Java programs.

CO2: Apply object oriented programming features and concepts for solving real world problem.

CO3: Develop Java programs using the concepts of inheritance and interfaces.

CO4: Build Java applications using packages, exceptions and I/O streams.

CO5: Implement multithreaded programming and to developGUI using JavaFX.

UNIT I

Object Oriented Programming: Basic concepts, Principles.

Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

Data Types, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, **Introduction to Operators**, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

Control Statements:Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.

UNIT II

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.

UNIT III

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays,

Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three-dimensional Arrays, Arrays as Vectors.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance.

Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

UNIT IV

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

Java I/O and File: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java (Text Book 2).

UNIT V

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings,Comparison, Modifying, Searching; Class String Buffer.

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Java FX GUI: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)

Text Books:

1) JAVA one step ahead, Anitha Seth, B.L.Juneja, Oxford.
2) Joy with JAVA, Fundamentals of Object Oriented Programming, DebasisSamanta, MonalisaSarma, Cambridge, 2023.
3) JAVA for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

References Books:

1) The complete Reference Java, 11thedition, Herbert Schild,TMH
2) Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

Online Resources:

1) https://nptel.ac.in/courses/106/105/106105191/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816347_shared/overview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Advanced Data Structures & Algorithm Analysis		L	T	P	C
23BTCS04P	**Lab**(Common to CSE, CS & IT and CSE(AI))		0	0	3	1.5
Year	II	Semester	I			

Course Objectives:

The objectives of the course is to

- acquire practical skills in constructing and managing Data structures
- apply the popular algorithm design methods in problem-solving scenarios

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

CO1: Implement heap and various tree structures like AVL, B-Tree and graphs.

CO2: Implement various Sorting Techniques.

CO3: Develop greedy method, dynamic programming, backtracking and branch & bound algorithms for various real-time applications.

Experiments covering the Topics:

- Operations on AVL trees, B-Trees, Heap Trees
- Graph Traversals
- Sorting techniques
- Finding Biconnected components in a graph
- Shortest path algorithms using greedy Method
- 0/1 Knapsack Problem using Dynamic Programming and Backtracking
- Travelling Salesperson problem using Branch and Bound
- N-Queens Problem using Backtracking
- Job Sequencing using Branch and Bound

Sample Programs:

1. Construct an AVL tree for a given set of elements which are stored in a file. And implement insert and delete operation on the constructed tree. Write contents of tree into a new file using in-order.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array. Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by
   a) Adjacency Matrix b) Adjacency Lists
5. Write a program for finding the biconnected components in a given graph.
6. Implement Quick sort and Merge sort and observe the execution time for various input sizes (Average, Worst and Best cases).
7. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
8. Implement Job sequencing with deadlines using Greedy strategy.
9. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
10. Implement N-Queens Problem Using Backtracking.
11. Use Backtracking strategy to solve 0/1 Knapsack problem.
12. Implement Travelling Sales Person problem using Branch and Bound approach.

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, SahniSartaj, Mehta, Dinesh, 2nd Edition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, 2nd Edition, University Press
3. Data Structures and program design in C, Robert Kruse, Pearson Education Asia
4. An introduction to Data Structures with applications, Trembley& Sorenson, McGraw Hill

Online Learning Resources:

1. http://cse01-iiith.vlabs.ac.in/
2. http://peterindia.net/Algorithms.html

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Object Oriented Programming through Java Lab		L	T	P	C
23BTCS05P	(Common to CSE, CS & IT and CSE(AI))		0	0	3	1.5
Year	II	Semester	I			

Course Objectives:

The aim of this course is to

- Practice object-orientedprogramming in the Java programming language
- ImplementClasses, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrateinheritance, Exception handling mechanism,JDBC connectivity
- Construct Threads, Event Handling, implement packages, Java FX GUI

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

CO1: Demonstrate a comprehensive understanding of Java syntax and utilize it effectively to solve programming problems.

CO2: Apply fundamental Object-Oriented Programming (OOP) principles to design and implement software solutions.

CO3: Develop graphical user interface (GUI) applications using JavaFX and apply algorithmic thinking to solve computer science problems efficiently.

Experiments covering the Topics:

- Object Oriented Programming fundamentals- data types, control structures
- Classes, methods, objects, Inheritance, polymorphism,
- Exception handling, Threads, Packages, Interfaces
- Files, I/O streams, JavaFX GUI

Sample Experiments:

Exercise – 1:

a) Write a JAVA program to display default value of all primitive data type of JAVA

b) Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.

Exercise - 2

a) Write a JAVA program to search for an element in a given list of elements using binary search mechanism.

b) Write a JAVA program to sort for an element in a given list of elements using bubble sort

c) Write a JAVA program using StringBuffer to delete, remove character.

Exercise - 3

a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.

b) Write a JAVA program implement method overloading.

c) Write a JAVA program to implement constructor.

d)Write a JAVA program to implement constructor overloading.

Exercise - 4

a) Write a JAVA program to implement Single Inheritance

b) Write a JAVA program to implement multi-level Inheritance

c) Write a JAVA program for abstract class to find areas of different shapes

Exercise - 5

a) Write a JAVA program give example for "super" keyword.
b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?
c) Write a JAVA program that implements Runtime polymorphism

Exercise - 6

a) Write a JAVA program that describes exception handling mechanism
b) Write a JAVA program Illustrating Multiple catch clauses
c) Write a JAVA program for creation of Java Built-in Exceptions
d) Write a JAVA program for creation of User Defined Exception

Exercise - 7

a) Write a JAVA program that creates threads by extending Thread class.First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds,(Repeat the same by implementing Runnable)
b) Write a program illustrating **is Alive** and **join ()**
c) Write a Program illustrating Daemon Threads.
d) Write a JAVA program Producer Consumer Problem

Exercise – 8

a) Write a JAVA program that import and use the user defined packages
b) Without writing any code, build a GUI that display text in label and image in an ImageView (use JavaFX)
c) Build a Tip Calculator app using several JavaFX components and learn how to respond to user interactions with the GUI
d) Write a Java program that demonstrates how to connect to a database using JDBC

References Books:

1. P. J. Deitel, H. M. Deitel, "Java for Programmers", Pearson Education, PHI, 4th Edition, 2007.
2. P. Radha Krishna, "Object Oriented Programming through Java", Universities Press, 2nd Edition, 2007
3. Bruce Eckel, "Thinking in Java", Pearson Education, 4th Edition, 2006.
4. SachinMalhotra, SaurabhChaudhary, "Programming in Java", Oxford University Press, 5th Edition, 2010.

Online Learning Resources:

https://java-iitd.vlabs.ac.in/
http://peterindia.net/JavaFiles.html

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Python Programming(Skill Enhancement Course)		L	T	P	C
23BTCS01S	(Common to All Branches of Engineering)		0	1	2	2
Year	II	Semester	I			

CourseObjectives:

The main objectives of the course are to

- Introduce core programming concepts of Python programming language.
- Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
- Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these

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

CO1: Design solutions to mathematicalproblems

CO2: Develop, run and manipulate python programs using core data structures, files andobject-oriented programming (OOP) principles.

CO3: Develop various applications using Python libraries and frameworks such as JSON, XML, NumPy, and pandas for efficient data manipulation and processing.

UNTI-I:

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupyter Notebook.

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions, Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments, Reading Input, Print Output, Type Conversions, the type () Function and Is Operator, Dynamic and Strongly Typed Language.

Control Flow Statements: if statement, if-else statement, if...elif…else, Nested if statement, while Loop, for Loop, continue and break Statements, Catching Exceptions Using try and except Statement.

Sample Experiments:

1. Write a program to find the largest element among three Numbers.
2. Write a Program to display all prime numbers within an interval
3. Write a program to swap two numbers without using a temporary variable.
4. Demonstrate the following Operators in Python with suitable examples.
   i) Arithmetic Operators ii) Relational Operators iii) Assignment Operatorsiv) Logical Operators v) Bit wise Operators vi) Ternary Operator vii) Membership Operatorsviii) Identity Operators.
5. Write a program to add and multiply complex numbers
6. Write a program to print multiplication table of a given number.

UNIT-II:

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used on Lists, List Methods, del Statement.

Sample Experiments:

7. Write a program to define a function with multiple return values.
8. Write a program to define a function using default arguments.

9. Write a program to find the length of the string without using any library functions.
10. Write a program to check if the substring is present in a given string or not.
11. Write a program to perform the given operations on a list:
    i. Additionii. Insertioniii. slicing
12. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III:

Dictionaries: Creating Dictionary, Accessing and Modifying key:value Pairs in Dictionaries, Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.

Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple() Function, Indexing and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

Sample Experiments:

13. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the tuples and print the concatenated tuples.
14. Write a program to count the number of vowels in a string (No control flow allowed).
15. Write a program to check if a given key exists in a dictionary or not.
16. Write a program to add a new key-value pair to an existing dictionary.
17. Write a program to sum all the items in a given dictionary.

UNIT-IV:

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data, Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Python os and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

Sample Experiments:

18. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
19. Python program to print each line of a file in reverse order.
20. Python program to compute the number of characters, words and lines in a file.
21. Write a program to create, display, append, insert and reverse the order of the items in the array.
22. Write a program to add, transpose and multiply two matrices.
23. Write a Python program to create a class that represents a shape. Include methods to calculate its area and perimeter. Implement subclasses for different shapes like circle, triangle, and square.

UNIT-V:

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with Python, Pandas.

Sample Experiments:

24. Python program to check whether a JSON string contains complex object or not.
25. Python Program to demonstrate NumPy arrays creation using array () function.
26. Python program to demonstrate use of ndim, shape, size, dtype.
27. Python program to demonstrate basic slicing, integer and Boolean indexing.
28. Python program to find min, max, sum, cumulative sum of array
29. Create a dictionary with at least five keys and each key represent value as a list where this list contains at least ten values and convert this dictionary as a pandas data frame and

explore the data through the data frame as follows:

a) Apply head () function to the pandas data frame
b) Perform various data selection operations on Data Frame

30. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

Reference Books:

1. Gowrishankar S, Veena A., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2nd Edition, Pearson, 2024
3. Introduction to Programming Using Python, Y. Daniel Liang, Pearson.

Online Learning Resources/Virtual Labs:

1. https://www.coursera.org/learn/python-for-applied-data-science-ai
2. https://www.coursera.org/learn/python?specialization=python#syllabus

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Environmental Science		L	T	P	C
23BTCE01A	(Common to All Branches of Engineering)		2	0	0	0
Year	II	Semester	I			

CourseObjectives:

- Tomakethestudents togetawarenesson environment.
- Tounderstandtheimportanceofprotectingnaturalresources,ecosystemsforfuturegeneratio nsandpollutioncausesduetotheday-to-dayactivitiesofhumanlife
- Tosaveearthfromtheinventionsbytheengineers.
- Assess the sustainability of various human practices in terms of resource use and waste management.
- To apply theoretical knowledge to real-world environmental challenges through experiential learning opportunities

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

CO1: Comprehend the concepts of environment and its importance in our daily life and develop and apply various water conservation methods and conservation of other natural resources also.
CO2: Identify the importance of environmental education for protection of life cycles of various bio systems which are essential for bio sphere.
CO3: Develop new innovative methods for controlling of environmental pollution which may affect the human health.
CO4: Analyse environmental issues related to society and find solutions for environmental problems.
CO5: Analyse the effects of increasing human population as well as health associated problems and develops measures to be taken to protect human health.

UNITI

Multidisciplinary Nature of Environmental Studies: – Definition,Scope, and Importance – NeedforPublicAwareness.

Natural Resources : Renewable and non-renewable resources – Natural resources and associatedproblems – Forest resources – Use and over – exploitation, deforestation, case studies – Timberextraction – Mining, dams and other effects on forest and tribal people – Water resources – Use andover utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefitsand problems – Mineral resources: Use and exploitation, environmental effects of extracting and usingmineralresources,casestudies– Foodresources:Worldfoodproblems,changescausedbyagricultureandovergrazing,effectsofmodern agriculture,fertilizer-pesticideproblems,waterlogging,salinity,casestudies.–Energyresources:

UNITII

Ecosystems: Conceptofanecosystem.–Structureand functionofanecosystem – Producers,consumers, and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains,foodwebsandecologicalpyramids– Introduction,types,characteristicfeatures,structure,andfunctionofthe followingecosystem:

a. Forestecosystem.
b. Grasslandecosystem
c. Desertecosystem.
d. Aquaticecosystems(ponds,streams,lakes,rivers,oceans,estuaries)

BiodiversityanditsConservation:

Introduction, Definition, genetic,speciesandecosystemdiversity–Bio-geographicalclassificationofIndia–Valueofbiodiversity:consumptiveuse,Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and

Locallevels – India as a mega-diversity nation – Hot-sports of biodiversity – Threats to biodiversity: habitatloss,poaching ofwildlife,man-wildlife conflicts– Endangered and endemic speciesof India–Conservationof biodiversity:In-situandEx-situconservationofbiodiversity.

UNITIII

EnvironmentalPollution: Definition,Cause,effects,andcontrolmeasuresof:

a. AirPollution.
b. Waterpollution
c. Soilpollution
d. Marinepollution
e. Noisepollution
f. Thermalpollution
g. Nuclearhazards

Solid Waste Management: Causes**,** effects, and control measures of urban and industrial wastes –Role of an individual in prevention of pollution – Pollution case studies – Disaster management:floods,earthquake,cycloneandlandslides.

UNITIV

Social Issues and the Environment: From Unsustainable to Sustainable development– Urbanproblems related to energy – Water conservation, rain water harvesting, and watershed management –Resettlement and rehabilitation of people; its problems and concerns.Case studies – Environmentalethics:Issues and possible solutions – Climate change,global warming, acid rain,ozone layerdepletion, nuclear accidents, and holocaust.Case Studies – Wasteland reclamation. – Consumerismand waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution)Act. –Water(PreventionandcontrolofPollution)Act– WildlifeProtectionAct–ForestConservationAct– Issuesinvolvedinenforcementofenvironmentallegislation–Publicawareness.

UNITV

Human Population and the Environment: Population growth, variation among nations.Populationexplosion – Family Welfare Programmes. – Environment and human health – Human Rights – ValueEducation–HIV/AIDS–WomenandChildWelfare– RoleofinformationTechnologyinEnvironmentandhumanhealth–Casestudies.

FieldWork: Visittoa localareatodocumentenvironmentalassetsRiver/forestgrassland/hill/mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study ofcommonplants,insects,andbirds– river,hillslopes,etc.

Textbooks:

1. TextbookofEnvironmentalStudiesforUndergraduateCoursesErachBharuchaforUniversity GrantsCommission, UniversitiesPress.
2. Palaniswamy,“EnvironmentalStudies”,Pearsoneducation
3. S.AzeemUnnisa,“EnvironmentalStudies”AcademicPublishingCompany
4. K.RaghavanNambiar,“TextbookofEnvironmentalStudiesforUndergraduateCoursesasperUGCmodelsyllabus”,ScitechPublications (India),Pvt.Ltd.

References:

1. DeekshaDaveandE.SaiBabaReddy,“TextbookofEnvironmentalScience”,CengagePublicatio

ns.

2. M.AnjiReddy, “TextbookofEnvironmentalSciencesandTechnology”, BS Publication.
3. J.P.Sharma,ComprehensiveEnvironmentalstudies,Laxmipublications.
4. J.GlynnHenryandGaryW.Heinke,“EnvironmentalSciencesandEngineering”, PrenticeHallofIndiaPrivatelimited
5. G.R.Chatwal, “AText BookofEnvironmentalStudies”HimalayaPublishingHouse
6. GilbertM.MastersandWendellP.Ela,“IntroductiontoEnvironmentalEngineeringandScience, Prentice HallofIndiaPrivatelimited.

Web Materials:

1. https://www.coursera.org/learn/python-for-applied-data-science-ai
2. https://www.coursera.org/learn/python?specialization=python#syllabus

Course Code	OptimizationTechniques		L	T	P	C
23BTME10T	(Common to All Branches of Engineering)		2	0	0	2
Year	II	Semester	II			

Course Objectives:

The objectives of the course are to

- To provide the basic knowledge about Optimization, importance, application areas ofLinear Programmingin the industry.
- To impart different optimization models under typical situations in the businessorganization like transportation, assignment.
- To understand the process of sequencing in a typical industry.
- To describe different game strategies under cut-throat competitive businessenvironment
- To develop networks of activities of projects and to find out optimal modes ofcompleting projects using network modelling evaluation techniques.

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

CO1: Develop mathematical models of the real life situations and capable of formulating and obtaining best solutions for linear programming problems.

CO2: Formulate and Solve Transportation & Assignment Models.

CO3: Solve Job shop and Flow shop scheduling problems.

CO4: Appreciate various game models and determine optimum strategies and game value.

CO5: Construct project network and differentiate CPM and PERT techniques.

UNIT - I

Introduction: Meaning, Nature, Scope & Significance of Optimization – Typical applications. The Linear Programming Problem – Introduction, Formulation of Linear Programming problem, Limitations of L.P.P, Graphical method, Simplex method: Maximization and Minimization model(exclude Duality problems), Big-M method and Two Phase method.

UNIT - II

Transportation Problem: Introduction, Transportation Model, Finding initial basic feasible solutions, moving towards optimality, Unbalanced Transportation problems, Transportation problems with maximization, Degeneracy.

Assignment Problem – Introduction, Mathematical formulation of the problem, Solution of an Assignment problem, Hungarian Algorithm, Multiple Solution, Unbalanced Assignment problems, Maximization in Assignment Model.

UNIT - III

Sequencing – Job sequencing, Johnsons Algorithm for n Jobs and Two machines, n Jobs and Three Machines, n jobs through m machines, Two jobs and m Machines Problems.

UNIT - IV

Game Theory: Concepts, Definitions and Terminology, Two Person Zero Sum Games, Pure Strategy Games (with Saddle Point), Principal of Dominance, Mixed Strategy Games (Game without Saddle Point), Significance of Game Theory in Managerial Application.

UNIT - V

Project Management: Network Analysis – Definition –objectives -Rules for constructing Network diagram- Determining Critical Path – Earliest & Latest Times – Floats – Application of CPM and PERT techniques in Project Planning and Control – PERT Vs CPM. (exclude Project Crashing).

Textbooks:

1. Operations Research / R.Pannerselvam, PHI Publications.
2. Operations Research / S.D.Sharma-Kedarnath
3. Operations Research /A.M.Natarajan,P.Balasubramani,A. Tamilarasi/Pearson Education.
4. Engineering Optimization: Theory and practice / S.S.Rao, New Age International (P) Limited

Reference Books:

1. Quantitative Techniques in Management / ND Vohra, Tata McGraw Hill, 4th Edition,2011.
2. Introduction to O.R/Hiller &Libermann (TMH).
3. Operations Research: Methods & Problems / Maurice Saseini, ArhurYaspan&Lawrence Friedman. Pearson
4. Quantitative Analysis For Management/ Barry Render, Ralph M. Stair, Jr and Michael E. Hanna/
5. Operations Research / Wagner/ PHI Publications.

Online Learning Sources

https://onlinecourses.swayam2.ac.in/cec20_ma10/preview
https://onlinecourses.nptel.ac.in/noc20_ma23/preview
https://onlinecourses.nptel.ac.in/noc19_ma29/preview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Probability & Statistics		L	T	P	C
23BTHS12T	(Common to CSE,CS & IT and CSE(AI))		3	0	0	3
Year	II	Semester	II			

Course Objective:

This course aims at providing the student with the knowledge on the theory of Probability and random variables and usage of statistical techniques like testing of hypothesis, testing of significance, chi-square test and basic concepts of Least square methods.

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

CO1: Analyze the data quantitatively or categorically using various statistical elementarytools.

CO2: Design mathematical models involving probability, random variables and the critical thinking in the theory of probability and its applications in real life problems.

CO3: Apply the theoretical probability distributions like binomial, Poisson, and Normal in the relevant application areas.

CO4: Analyze to test various hypotheses included in theory and types of errors for large samples.

CO5: Apply the different testing tools like t-test, F-test, chi-square test to analyze the relevant real life problems.

UNIT I: Descriptive statistics

Statistics Introduction, Population vs Sample, Collection of data, primary and secondary data, Measures of Central tendency, Measures of Variability (spread or variance) Skewness, Kurtosis, correlation, correlation coefficient, rank correlation, regression coefficients, method of least squares, regression lines.

UNIT II:Probability

Probability, probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, random variables (discrete and continuous), probability density functions, properties, mathematical expectation**.**

UNIT III: Probability distributions

Probability distributions: Binomial, Poisson and Normal-their properties (Chebyshevs inequality). Approximation of the binomial distribution to normal distribution.

UNITIV: Estimation and Testing of hypothesis, large sample tests

Estimation-parameters, statistics, sampling distribution, point estimation, Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test. Large Sample Tests: Test for single proportion, difference of proportions, test for single mean and difference of means. Confidence interval for parameters in one sample and two sample problems

UNIT V: Small sample tests

Student-distribution (test for single mean, two means and paired t-test),testing of equality of variances (F-test), $\chi 2$ - test for goodness of fit, $\chi 2$ - test for independence of attributes.

Textbooks:

1. Millerand Freunds, **Probability and Statistics for Engineers**,7/e,Pearson,2008.
2. S.C. Gupta and V.K. Kapoor, **Fundamentals of Mathematical Statistics**, 11/e, Sultan Chand & Sons Publications, 2012.

Reference Books:

1. S.Ross, A **First Course in Probability**, Pearson EducationIndia,2002.
2. W.Feller, **An Introduction to Probability Theory and its Applications**, 1/e, Wiley, 1968.
3. B.V.Ramana, **Higher Engineering Mathematics**, McGrawHillEducation.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Operating Systems		L	T	P	C
23BTCS05T	(Common to CSE and CS& IT)		3	0	0	3
Year	II	Semester	II			

Course Objectives:

The main objectives of the course is to make student

- Understand the basic concepts and principles of operating systems, including process management, memory management, file systems, and Protection
- Make use of process scheduling algorithms and synchronization techniques to achieve better performance of a computer system.
- Illustrate different conditions for deadlock and their possible solutions.

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

CO1: Understand functional architecture of operating systems.
CO2: Analyze process coordination and Distinguish CPU scheduling algorithms.
CO3: Evaluate the problems related to synchronization mechanismsand deadlock handling
CO4: Evaluate various memory management techniques.
CO5: Evaluate File System and directory implementations.

UNIT - I

Operating Systems Overview: Introduction, Operating system functions, Operating systems operations, Computing environments, Free and Open-Source Operating Systems
System Structures: Operating System Services, User and Operating-System Interface, system calls, Types of System Calls, system programs, Operating system Design and Implementation, Operating system structure, Building and Booting an Operating System, Operating system debugging

UNIT - II

Processes: Process Concept, Process scheduling, Operations on processes, Inter-process communication.
Threads and Concurrency: Multithreading models, Thread libraries, Threading issues.
CPU Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling.

UNIT – III

Synchronization Tools: The Critical Section Problem, Peterson's Solution, Mutex Locks, Semaphores, Monitors, Classic problems of Synchronization.
Deadlocks: system Model, Deadlock characterization, Methods for handling Deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from Deadlock.

UNIT - IV

Memory-Management Strategies: Introduction, Contiguous memory allocation, Paging, Structure of the Page Table, Swapping.
Virtual Memory Management: Introduction, Demand paging, Copy-on-write, Page replacement, Allocation of frames, Thrashing
Storage Management: Overview of Mass Storage Structure, HDD Scheduling.

UNIT - V

File System: File System Interface: File concept, Access methods, Directory Structure; File system Implementation: File-system structure, File-system Operations, Directory

implementation, Allocation method, Free space management; File-System Internals: File-System Mounting, Partitions and Mounting, File Sharing.

Protection: Goals of protection, Principles of protection, Protection Rings, Domain of protection, Access matrix.

TextBooks:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 9th Edition, Wiley, 2023.
2. Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2023

Reference Books:

1. Operating Systems -Internals and Design Principles, Stallings W, 9thEdition, Pearson.
2. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, Global Edition, Wiley, 2023.

Online Learning Resources:

1. https://nptel.ac.in/courses/106/106/106106144/
2. http://peterindia.net/OperatingSystems.html

Course Code	Database Management Systems		L	T	P	C
23BTCS06T	(Common to CSE, CS & IT and CSE(AI))		3	0	0	3
Year	II	Semester	II			

Course Objectives:

The main objectives of the course is to

- Introduce database management systems and to give a good formal foundation on the relational model of data and usage of Relational Algebra
- Introduce the concepts of basic SQL as a universal Database language
- Demonstrate the principles behind systematic database design approaches by covering conceptual design, logical design through normalization
- Provide an overview of physical design of a database system, by discussing Database indexing techniques and storage techniques

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

CO1: Analyse a given database application scenario to use ER model for conceptual design of the database.

CO2: Apply the concepts of structured query language to create, query and manipulate database with the given constraint.

CO3: Design relational databases using SQL, optimizing data handling and retrieval through schema design, complex querying, and performance techniques.

CO4: Apply the concept of functional dependencies and normalization techniques to refine databases.

CO5: Apply transaction processing, concurrency control, and database recovery protocols within databases.

UNIT I:

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

Entity Relationship Model: Introduction, Representation of entities, attributes, entity set, relationship, relationship set, constraints, sub classes, super class, inheritance, specialization, generalization using ER Diagrams.

UNIT II:

Relational Model: Introduction to relational model, concepts of domain, attribute, tuple, relation, importance of null values, constraints (Domain, Key constraints, integrity constraints) and their importance, Relational Algebra, Relational Calculus. BASIC SQL:Simple Database schema, data types, table definitions (create, alter), different DML operations (insert, delete, update).

UNIT III:

SQL:Basic SQL querying (select and project) using where clause, arithmetic & logical operations, SQL functions(Date and Time, Numeric, String conversion).Creating tables with relationship, implementation of key and integrity constraints, nested queries, sub queries, grouping, aggregation, ordering, implementation of different types of joins, view(updatable and non-updatable), relational set operations.

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

UNIT V:
Transaction Concept:Transaction State, ACID properties, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for Serializability, lock based, time stamp based, optimistic, concurrency protocols, Deadlocks, Failure Classification, Storage, Recovery and Atomicity, Recovery algorithm.
Introduction to Indexing Techniques: B+ Trees, operations on B+Trees, Hash Based Indexing:

Text Books:
1) Database Management Systems, 3rd edition, Raghurama Krishnan, Johannes Gehrke, TMH (For Chapters 2, 3, 4)
2) Database System Concepts,5th edition, Silberschatz, Korth, Sudarsan,TMH (For Chapter 1 and Chapter 5)

Reference Books:
1) Introduction to Database Systems, 8thedition, C J Date, Pearson.
2) Database Management System, 6th edition, RamezElmasri, Shamkant B. Navathe, Pearson
3) Database Principles Fundamentals of Design Implementation and Management, 10th edition, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

Web-Resources:
1) https://nptel.ac.in/courses/106/105/106105175/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012758066672820224 56_shared/overview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Software Engineering		L	T	P	C
23BTCS07T	(Common to CSE and CS & IT)		3	0	0	3
Year	II	Semester	II			

CourseObjectives:

The objectives of this course are to introduce

- Software life cycle models, Software requirements and SRS document.
- Project Planning, quality control and ensuring good quality software.
- Software Testing strategies, use of CASE tools, Implementation issues,validation&verificationprocedures.

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

CO1: Analyze various software engineering models and apply for design and development of software projects.

CO2: Analyze requirements using formal methods and estimation techniques effectively.

CO3: Design software systems by integrating principles of software design, agility, function oriented design methodologies, and user interface design principles.

CO4: Apply various testing techniques for a software project to improve quality of a project.

CO5: Apply CASE tools and techniques for engineering software projects.

UNIT I:

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT II:

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis And Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT III:

Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling.Approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)

Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT IV:Coding and Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing**,** Smoke testing, and some general issues associated with testing.

Software Reliability And Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system, ISO 9000.SEI Capability maturity model.Few other important quality standards and Six Sigma**.**

UNIT V:

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

Software Reuse: reuse- definition, introduction, reason behind no reuse so far, Basic issues in any reuse program.

Text Books:

1. Fundamentals of Software Engineering, Rajib Mall, 5th Edition,PHI.
2. Software Engineering A practitioner's Approach, Roger S. Pressman, 9th Edition, Mc-Graw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10thEdition, Pearson.
2. SoftwareEngineering, PrinciplesandPractices, Deepak Jain, Oxford University Press.

E-Resources:

1) https://nptel.ac.in/courses/106/105/106105182/
2) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012605895063871488827_shared/overview
3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_013382690411003904735_shared/overview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Operating Systems & Software Engineering Lab		L	T	P	C
23BTIT01P			0	0	3	1.5
Year	II	Semester	II			

Course Objectives:

The main objectives of the course are to

- Provide insights into system calls, file systems, semaphores,
- Develop and debug CPU Scheduling algorithms, page replacement algorithms, thread implementation
- Implement Bankers Algorithms to Avoid the Dead Lock
- acquire the generic software development skill through various stages of software life cycle
- generate test cases for software testing

Experiments covering the Topics:

- UNIX fundamentals, commands & system calls
- CPU Scheduling algorithms, thread processing
- IPC, semaphores, monitors, deadlocks
- Page replacement algorithms, file allocation strategies
- Memory allocation strategies
- Software Requirement Specification, DFD, CFD
- Software estimation, UML diagrams, test case design

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

CO1: Evaluate scheduling algorithms, synchronization techniques, and memory management principles to enhance the performance and efficiency of computer systems.

CO2: Design solutions for deadlock conditions, and evaluate memory management strategies and allocation policies in operating systems.

CO3: Design UML diagrams, E-R diagrams, DFD and CFD for various applications.

Sample Experiments in Operating Systems:

1. Practicing of Basic UNIX Commands.
2. Write programs using the following UNIX operating system calls
   fork, exec, getpid, exit, wait, close, stat, opendir and readdir
3. Simulate UNIX commands like cp, ls, grep, etc.,
4. Simulate the following CPU scheduling algorithms
   a) FCFS b) SJF c) Priority d) Round Robin
5. Control the number of ports opened by the operating system with
   a) Semaphore b) Monitors.
6. Write a program to illustrate concurrent execution of threads using pthreads library.
7. Write a program to solve producer-consumer problem using Semaphores.
8. Implement the following memory allocation methods for fixed partition
   a) First fit b) Worst fit c) Best fit
9. Simulate the following page replacement algorithms
   a) FIFO b) LRU c) LFU
10. Simulate Paging Technique of memory management.
11. Implement Bankers Algorithm for Dead Lock avoidance and prevention
12. Simulate the following file allocation strategies
    a) Sequential b) Indexed c) Linked

Sample Experiments in Software Engineering:

1) Perform the following, for the following experiments:
    i. Do the Requirement Analysis and Prepare SRS
    ii. Draw E-R diagrams, DFD, CFD and structured charts for the project.
    a. Course Registration System
    b. Students Marks Analyzing System
    c. Online Ticket Reservation System
    d. Stock Maintenance
2) Consider any application, using COCOMO model, estimate the effort.
3) Consider any application, Calculate effort using FP oriented estimation model.
4) Draw the UML Diagrams for the problem a, b, c, d.
5) Design the test cases for e-Commerce application (Flipcart, Amazon)
6) Design the test cases for a Mobile Application (Consider any example from Appstore)
7) Design and Implement ATM system through UML Diagrams.

Reference Books:

1. Operating System Concepts, Silberschatz A, Galvin P B, Gagne G, 9th Edition, Wiley, 2023.
2.Modern Operating Systems, Tanenbaum A S, 4th Edition, Pearson, 2023
3. Operating Systems -Internals and Design Principles, Stallings W, 9th Edition, Pearson.
4. Software Engineering? A Practitioner‟ s Approach, Roger S. Pressman, MGH.
5. Software Engineering by Ian Sommerville, Pearson Edu, 5th edition.

Online Learning Resources:

1. https://www.cse.iitb.ac.in/~mythili/os/
2. http://peterindia.net/OperatingSystems.html
3.Software Engineering Virtual Lab — IIT Kharagpur (iitkgp.ac.in)

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Database Management Systems Lab		L	T	P	C
23BTCS07P	(Common to CSE, CS & IT and CSE(AI))		0	0	3	1.5
Year	II	Semester	II			

Course Objectives:

This Course will enable students to

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

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

CO1: Apply SQL proficiently to address diverse query challenges and demonstrate effective database management.

CO2: Design database application scenarios using the ER model for conceptual design and apply normalization methods to optimize database structure.

CO3: Implement transaction processing, concurrency control, and database recovery protocols to ensure data integrity and system reliability.

Experiments covering the topics:

- DDL, DML, DCL commands
- Queries, nested queries, built-in functions,
- PL/SQL programming- control structures
- Procedures, Functions, Cursors, Triggers,
- Database connectivity- ODBC/JDBC

Sample Experiments:

1. Creation, altering and dropping of tables and inserting rows into a table (use constraints while creating tables) examples using SELECT command.
2. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSET, Constraints. Example:- Select the roll number and name of the student who secured fourth rank in the class.
3. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.
4. Queries using Conversion functions (to_char, to_number and to_date), string functions (Concatenation, lpad, rpad, ltrim, rtrim, lower, upper, initcap, length, substr and instr), date functions (Sysdate, next_day, add_months, last_day, months_between, least, greatest, trunc, round, to_char, to_date)
5.
   i. Create a simple PL/SQL program which includes declaration section, executable section and exception –Handling section (Ex. Student marks can be selected from the table and printed for those who secured first class and an exception can be raised if no records were found)
   ii. Insert data into student table and use COMMIT, ROLLBACK and SAVEPOINT in PL/SQL block.
6. Develop a program that includes the features NESTED IF, CASE and CASE expression. The program can be extended using the NULLIF and COALESCE functions.
7. Program development using WHILE LOOPS, numeric FOR LOOPS, nested loops using ERROR Handling, BUILT –IN Exceptions, USE defined Exceptions, RAISE-APPLICATION ERROR.

8. Programs development using creation of procedures, passing parameters IN and OUT of PROCEDURES.
9. Program development using creation of stored functions, invoke functions in SQL Statements and write complex functions.
10. Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.
11. Develop Programs using BEFORE and AFTER Triggers, Row and Statement Triggers and INSTEAD OF Triggers
12. Create a table and perform the search operation on table using indexing and non-indexing techniques.
13. Write a Java program that connects to a database using JDBC
14. Write a Java program to connect to a database using JDBC and insert values into it
15. Write a Java program to connect to a database using JDBC and delete values from it

Reference Books:

1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI.
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education.
4. RamezElmasri, Shamkant, B. Navathe, "Database Systems", Pearson Education, 6th Edition, 2013.
5. Database Principles Fundamentals of Design Implementation and Management, 10th edition, Corlos Coronel, Steven Morris, Peter Robb, Cengage Learning, 2022

Online Learning Resources:

1. http://www.scoopworld.in
2. http://vlabs.iitb.ac.in/vlabs-dev/labs/dblab/index.php

Course Code	Python with DJango		L	T	P	C
23BTIT01S	(Skill Enhancement Course)		0	1	2	2
Year	II	Semester	II			

Course Objectives:

The main objectives of the course are to

- Design and build static as well as dynamic web pages and interactive web-based applications
- Web development using Django framework.
- Analyze and create functional website in Django and deploy Django Web Application on Cloud

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

CO1: Develop simple applications using Python libraries such as Tkinter, CherryPy, Web2Py and BeautifulSoup4.

CO2: Develop web applications using Django MVC, models, views and templates.

CO3: Implement user registration, login, and authentication functionalities in Django.

CO4: Perform CRUD operations on SQLite databases using Django's ORM

CO5: Develop Django applications on cloud platforms like Heroku, configuring environment variables, handling static files, and managing database connections for seamless deployment and scalability.

UNIT-I: Python libraries for web development :

Collections-Container datatypes, Tkinter-GUI applications, Requests-HTTP requests, BeautifulSoup4-web scraping, Scrapy, Zappa, Dash, CherryPy, Turbo Gears, Flask, Web2Py, Bottle, Falcon, Cubic Web, Quixote, Pyramid.

Sample Experiments:

1. Write a Python GUI program to import Tkinter package and create a window. Set its title and add a label to the window.
2. Write a Python program that designs a simple login form with labels and Entry widgets, arranging them in a grid using the Grid geometry manager.
3. Write a program using BeautifulSoup4 library for web scraping for a given URL
4. Develop a sample Hello World page using Flask framework
5. Develop a sample web page using CherryPy / Web2Py / Bottle Framework

UNIT-II: Introduction to Django Framework

Understanding Django environment, Features of Django and Django architecture, MVC and MTV, Urls and Views, Mapping the views to URLs, Django Template, Template inheritance Django Models, Creating model for site, Converting the model into a table, Fields in Models, Integrating Bootstrap into Django, Creating tables, Creating grids, Creating carousels.

Sample Experiments:

6. Create a Sample “Hello World” Application using Django
7. Create a Login and Registration Page using MVC architecture in Django Framework
8. Create a sample page in Django by integrating BootStrap.
9. Create an application with Tables, grids in Django
10. Create a Django App with Carousels feature.

UNIT-III: Integrating Accounts& Authentication on Django

Introduction to Django Authentication System, SecurityProblem&Solution with Django Creating Registration Form using Django, Adding Email Field in Forms, Configuring email settings, Sending emails with Django, Adding Grid Layout On Registration Page, Adding Page Restrictions, Login Functionality Test and Logout.

Sample Experiments:

11. Create a registration page using Authentication System
12. Create an application in Django to send emails using email settings and Grid Layout
13. Create an application in Django using page restriction / authentication with Login and Logout Functionality
14. Create a sample form using Django Forms

UNIT-IV: Connecting SQLite with Django

DatabaseMigrations,FetchDataFromDatabase,DisplayingDataOnTemplates,AddingConditionOnData,Sending data from url to view, Sending data from view to template, Saving objects into database, Sorting objects, Filtering objects, Deleting objects, Difference between session and cookie, Creating sessions and cookies in Django.

Sample Experiments:

15. Create an app in Django which fetches data from database and show as list and also save objects in database
16. Create an app in Django for performing CRUD operations on records in a database
17. Create an app in Django which uses session management and cookies to store and manage user sessions.

UNIT-V: Deploying Django Web Application on Cloud

Creating a functional website in Django, Four Important Pillars to Deploy, registering on Heroku and GitHub, Push project from Local System to GitHub, working with DjangoHeroku, Working with Static Root, Handling WSGI with gunicorn, setting up Database & adding users.

Sample Experiments:

18. Create a website in Django with login, and registration page.
19. Register on GitHub, and Heroku and deploy the website on Heroku with all the functionalities developed.
20. Configure Django to handle static files.

Text books:

1. MartinC.Brown,"Python:TheCompleteReferencePaperback",4th Edition2018,McGrawHillEducation.
2. ReemaThareja, "Python Programming: Using Problem Solving Approach", 3rd Edition 2017,Oxford.
3. Daniel Rubio, Apress, "Beginning Django Web Application Development and Deployment with Python", 2nd Edition2017,Apress.

Reference Books:

1. Tom Aratyn, "Building Django 2.0 Web Applications: Create enterprise-grade, scalable Python web applications easilywithDjango 2.0",2nd Edition 2018, Packt Publishing.
2. Harry Percival, "Test-Driven Development with Python: Obey the Testing Goat: Using Django, Selenium and JavaScript",2nd Edition 2019, Kindle Edition.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Design Thinking & Innovation		L	T	P	C
23BTME09T	(Common to All branches of Engineering)		1	0	2	2
Year	II	Semester	II			

Course Objectives:

The objective of this course is to familiarize students with design thinking process as a tool for breakthrough innovation. It aims to equip students with design thinking skills and ignite the minds to create innovative ideas, develop solutions for real-time problems.

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

CO1: Apply the basic principles of design thinking.

CO2: Apply the design thinking techniques for solving problems in various sectors.

CO3: Appreciate the difference between innovation and creativity.

CO4: Apply the techniques of product design to develop a new product.

CO5: Apply design thinking techniques for business models and start-ups.

UNIT I: Introduction to Design Thinking

Introduction to elements and principles of Design, basics of design-dot, line, shape, form as fundamental design components. Principles of design.Introduction to design thinking, history of Design Thinking, New materials in Industry.

UNIT II: Design Thinking Process

Design thinking process (empathize, analyze, idea & prototype), implementing the process in driving inventions, design thinking in social innovations. Tools of design thinking - person, costumer, journey map, brainstorming, product development

Activity: Every student presents their idea in three minutes, Every student can present design process in the form of flow diagram or flow chart etc. Every student should explain about product development.

UNIT III: Innovation

Art of innovation, Difference between innovation and creativity, role of creativity and innovation in organizations- Creativity to Innovation- Teams for innovation- Measuring the impact and value of creativity.

Activity: Debate on innovation and creativity, Flow and planning from idea to innovation, Debate on value-based innovation.

UNIT IV:Product Design

Problem formation, introduction to product design, Product strategies, Product value, Product planning, product specifications- Innovation towards product design- Case studies

Activity: Importance of modeling, how to set specifications, Explaining their own product design.

UNIT V: Design Thinking in Business Processes

Design Thinking applied in Business &Strategic Innovation, Design Thinking principles that redefine business – Business challenges: Growth, Predictability, Change, Maintaining Relevance, Extreme competition, Standardization. Design thinking to meet corporate needs- Design thinking for Startups- Defining and testing Business Models and Business Cases- Developing & testing prototypes.

Activity: How to market our own product, About maintenance, Reliability and plan for startup.

Textbooks:

1. Tim Brown, Change by design, Harper Bollins(2009)
2. IdrisMootee, Design Thinking for Strategic Innovation, 2013, JohnWiley&Sons.

Reference Books:

1. David Lee, Design Thinking in the Classroom, Ulysses press
2. Shruti N Shetty, Design the Future, Norton Press
3. William Lidwell, Universal Principles of Design-Kritinaholden, Jill Butter.
4. Chesbrough.H, The Era of Open Innovation – 2013

Online Learning Resources:

1. https://nptel.ac.in/courses/110/106/110106124/
2. https://nptel.ac.in/courses/109/104/109104109/
3. https://swayam.gov.in/nd1_noc19_mg60/pre

Course Code	Advanced Java		L	T	P	C
23BTIT01T			3	0	0	3
Year	III	Semester	I			

COURSE OBJECTIVES: ·

- To understand the fundamentals of web programming and client side scripting. ·
- To learn server side development using servlets, web sockets. ·
- To learn the Spring framework and build applications using Spring. ·
- To learn and implement the concept of Java Persistence API. ·
- To learn the advanced client side scripting and framework.

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

CO1: Understand Web Technologies and JavaScript Fundamentals
CO2: Apply Server-Side Programming Concepts using Servlets and Web Components
CO3: Develop Web Applications using Spring Framework and Spring Boot
CO4: Implement Data Persistence using JPA and Hibernate
CO5: Build Secure and Scalable RESTful Applications using Advanced Spring Features

UNIT I INTRODUCTION TO WEB & JAVASCRIPT

Introduction to Web: Server - Client - Communication Protocol (HTTP), Javascript Prototypes - Classes - Modules – Fetch API – JS Canvas - Storage: LocalStorage, Cookies, IndexedDB, JSON

UNIT II SERVER SIDE PROGRAMMING

Web Server: Web Containers - Web Components, Servlet: Lifecycle - Request - Servlet Context - Response - Filter - Session - Dispatching Requests, WebSocket, Logging - Log4j2, Build tool - Gradle. Introduction to Spring: IoC Container and Dependency Injection (DI)

UNIT III SPRING

Spring Configuration and Spring Boot, Spring MVC, Spring Bean Lifecycle - Dispatcher Servlet and Configuration - Interceptors – Annotations, Controllers - Views - Input Validation -File Upload Container,Dependency and IOC .

UNITI IV AOP,JAVA PERSISTENCE API AND HIBERNATE

Aspect Oriented Programming(AOP) - Entity: Basic, Embeddable and Collection Types - Identifiers - Entity Relationship - Inheritance, Persistence Context and Entity Manager, JPQL, Criteria API, Spring Data JPA - Specification and Projection.

UNIT V ADVANCED SPRING PROGRAMMING

Spring Boot JDBC - Spring Boot Actuator - Spring Cloud -Spring Boot Testing - Spring Security Architecture, Spring Cache - Building RESTful Web Services

REFERENCES

1. David Flanagan, “Java Script: The Definitive Guide”, O’Reilly Media, Inc, 7th Edition, 2020
2. Matt Frisbie, "Professional JavaScript for Web Developers", Wiley Publishing, Inc, 4th Edition, ISBN: 978-1-119-36656-0, 2019
3. Alex Banks, Eve Porcello, "Learning React", O’Reilly Media, Inc, 2nd Edition, 2020
4. David R. Heffelfinger, "Java EE 8 Application Development", Packt Publishing, First edition2017
5. Benjamin Muschko, "Gradle in Action", Manning Publications, First edition2014
6. IulianaCosmina, Rob Harrop, Chris Schaefer, Clarence Ho, "Pro Spring 5: An In-Depth Guide to the Spring Framework and Its Tools", Apress, Fifth edition2017

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	COMPUTER NETWORKS & INTERNET PROTOCOLS		L	T	P	C
23BTCS09T			3	0	0	3
Year	III	Semester	I			

Course Objectives: The course is designed to:
- Understand the basic concepts of Computer Networks.
- Introduce the layered approach for design of computer networks
- Expose the network protocols used in Internet environment
- Explain the format of headers of IP, TCP and UDP
- Familiarize with the applications of Internet
- Elucidate the design issues for a computer network

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

CO1: Describe the structure and components of the Internet, including the network edge, core, transmission media, and performance metrics such as delay, loss, and throughput.

CO2: Analyze data link layer protocols and techniques, including error detection/correction, multiple access protocols, and LAN switching technologies.

CO3: Apply routing algorithms and explain the structure and operation of the network layer, including IP addressing and internetworking.

CO4: Compare transport layer protocols like UDP and TCP, and analyze flow control, error handling, and congestion control mechanisms.

CO5: Describe the architecture and protocols of network applications including HTTP, DNS, email, peer-to-peer systems, and content distribution.

UNIT I: Computer Networks and the Internet **Lecture: 8 Hrs**

What Is the Internet? Network Edge, The Network Core, Delay, Loss, and Throughput inPacket Switched Networks (Textbook 2), Reference Models, Multimedia Networks, GuidedTransmission Media, Wireless Transmission (Textbook 1)

UNIT II:The Data Link Layer, Access Networks, and LANs **Lecture: 10 Hrs**

Data Link Layer Design Issues, Error Detection and Correction, Elementary Data LinkProtocols, Sliding Window Protocols (Textbook 1)

Introduction to the Link Layer, Error-Detection and -Correction Techniques, Multiple AccessLinks and Protocols, Switched Local Area Networks, Link Virtualization: A Network as a LinkLayer, Data Center Networking, Retrospective: A Day in the Life of a Web Page (Packet)(Textbook 2)

UNIT III: The Network Layer **Lecture: 8 Hrs**

Routing Algorithms, Internetworking, The Network Layer in The Internet (Textbook 1)

UNIT IV:The Transport Layer **Lecture: 9 Hrs**

Connectionless Transport: UDP (Textbook 2), The Internet Transport Protocols: TCP, Congestion Control (Textbook 1)

UNIT V: The Application Layer **Lecture: 8 Hrs**

Principles of Network Applications, The Web and HTTP, Electronic Mail in the Internet, DNS—The Internet‘s Directory Service, Peer-to-Peer Applications, Video Streaming and Content Distribution Networks (Textbook 2)

Textbooks:

1. Andrew S. Tanenbaum, David J. Wetherall, *Computer Networks*, 6th Edition, PEARSON.

2. James F. Kurose, Keith W. Ross, *Computer Networking: A Top-Down Approach*, 6th Edition, Pearson, 2019.

Reference Books:

1. Forouzan, *Data Communications and Networking*, 5th Edition, McGraw HillPublication.
2. YouluZheng, ShakilAkhtar, *Networks for Computer Scientists and Engineers*, OxfordPublishers, 2016.

Online Learning Resources:

https://nptel.ac.in/courses/106105183/25
https://www.nptelvideos.in/2012/11/computer-networks.html
https://nptel.ac.in/courses/106105183/3

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	AUTOMATA THEORY AND COMPILER DESIGN		L	T	P	C
23BTCS10T			3	0	0	3
Year	III	Semester	I			

Course Objectives:

1. Able to understand the concept of abstract machines, construct FA, Regular Expressions for theregular languages and equivalent FSMs.
2. Able to construct pushdown automata equivalent to Context free Grammars, construct TuringMachines and understand undecidability.
3. Emphasize the concepts learnt in phases of compiler, lexical analyser and Top-down parser.
4. Able to understand the concepts of Bottom-up parser, Intermediate Code Generation.
5. Able to understand the concepts of Code optimizer and Code Generation.

Course Outcomes:

CO1: Demonstrate knowledge on Automata Theory, Regular Expression and Analyze and Design of finite automata, and prove equivalence of various finite automata.
CO2: Demonstrate knowledge on context free grammar, Analyze and design of PDA and TM.
CO3: Understand the basic concept of compiler design, and its different phases which will be helpful toconstruct new tools like LEX, YACC, etc.
CO4: Ability to implement semantic rules into a parser that performs attribution while parsing and applyerror detection and correction methods. s
CO5: Apply the code optimization techniques to improve the space and time complexity of programswhile programming and Ability to design a compiler.

Unit-I: Introduction to Automata and Regular Expressions 12 Hrs

Introduction, Alphabets, Strings and Languages, Chomsky Hierarchy, Automata and Grammars,Regular Grammar and Language, Finite Automata, Deterministic finite Automata (DFA), Nondeterministic finite Automata (NFA), Equivalence of NFA and DFA, Minimization of Finite Automata, Regular Expressions, Converting Regular Grammar and Expression into Finite Automata, Pumping lemma for regular sets, Closure properties of regular sets (Without proof).

UNIT-II: Context Free Grammars and Pushdown Automata 12 Hrs

Context Free Language, Context Free Grammar, Derivation and Parse tree, Ambiguity, Simplificationof CFG‘s, Chomsky Normal Form, Greibach Normal Form, Push Down Automat (PDA), Design of PDA, Equivalence of PDA and CFL/CFG

UNIT-III: Turing Machines and Introduction to Compilers 12 Hrs

Turing Machine, TM Model, Language acceptance, Design of Turing Machine, Compilers, Phases ofCompiler, The role of Lexical Analyzer, Input Buffering.

UNIT-IV: Parsers and Intermediate Code Generation 12 Hrs

Top-Down parsers: Parsing, Recursive Descent Parsers, Predictive Parsers
Bottom–up Parsers: Shift-Reduce Parsing, LR parsers, Intermediate Code Generation: Three addresscodes.

UNIT-V: Code Optimization and Code Generation 12 Hrs

Code Optimization: Peephole optimization, Basic blocks and flow graphs, DAG, Principles of Source Code Optimization.
Code Generation: Issues in Design of Code Generation, Simple Code Generator.

Text Books:

1. Introduction to Automata theory languages and Computation, Hopcroft H.E. and Ullman Jeffrey.D,3/e, 2006, Pearson Education, New Delhi, India.
2. Mishra K L P and Chandrasekaran N, ―Theory of Computer Science - Automata, Languages and Computation‖, 2/e, 2007, PHI, New Delhi, India.
3. Compilers: Principles, Techniques, and Tools, Updated 2e July 2023 Alfred V. Aho , Monica S.Lam, Ravi Sethi , Jeffrey D. Ullman , SoravBansal

Reference Books:

1. Introduction to Languages and Theory of Computation, John C Martin, 1/e, 2009, Tata McGraw HillEducation, Hyderabad, India.
2. Introduction to Theory of Computation, Sipser, 2/e, 2005, Thomson, Australia.
3. Compiler Construction: Principles And Practice, Kenneth C. Louden, Thomson/ Delmar CengageLearning, 2006.
4. Lex &yacc, Doug Brown, John Levine and Tony Mason, 2 nd Edition, O‘reilly Media
5. Engineering a compiler, Keith Cooper and Linda Torczon, 2 nd Edition, MorganKaufmann, 2011.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	INTRODUCTION TO QUANTUM TECHNOLOGIES AND APPLICATIONS (Qualitative Treatment)		L	T	P	C
23BTCS11T			3	0	0	3
Year	III	Semester	I			

Course Objectives (COBJ):
- Introduce fundamental quantum concepts like superposition and entanglement.
- Understand theoretical structure of qubits and quantum information.
- Explore conceptual challenges in building quantum computers.
- Explain principles of quantum communication and computing.
- Examine real-world applications and the future of quantum technologies.

Course Outcomes (CO):
CO1: Explain core quantum principles in a non-mathematical manner.
CO2: Compare classical and quantum information systems.
CO3: Identify theoretical issues in building quantum computers.
CO4: Discuss quantum communication and computing concepts.
CO5: Recognize applications, industry trends, and career paths in quantum technology.

Unit 1: Introduction to Quantum Theory and Technologies
The transition from classical to quantum physics, Fundamental principles explained conceptually:Superposition, Entanglement, Uncertainty Principle, Wave-particle duality, Classical vs Quantummechanics – theoretical comparison,Quantum states and measurement: nature of observation,Overview of quantum systems: electrons, photons, atoms,The concept of quantization: discrete energy levels,Why quantum? Strategic, scientific, and technological significance,Asnapshot of quantum technologies: Computing, Communication, and Sensing,National and global quantum missions: India‘s Quantum Mission, EU, USA, China

Unit 2: Theoretical Structure of Quantum Information Systems
What is a qubit? Conceptual understanding using spin and polarization, Comparison: classical bitsvs quantum bits, Quantum systems: trapped ions, superconducting circuits, photons (nonengineering view),Quantum coherence and decoherence – intuitive explanation, Theoretical concepts: Hilbert spaces, quantum states, operators – only interpreted in abstract,The role of entanglement and non-locality in systems, Quantum information vs classical information: principles and differences, Philosophical implications: randomness, determinism, and observer role

Unit 3: Building a Quantum Computer – Theoretical Challenges and Requirements
What is required to build a quantum computer (conceptual overview)?,Fragility of quantum systems: decoherence, noise, and control, Conditions for a functional quantum system: Isolation, Error management, Scalability, Stability, Theoretical barriers: Why maintaining entanglement is difficult,Error correction as a theoretical necessity, Quantum hardware platforms (brief conceptual comparison),Superconducting circuits, Trapped ions, Photonics, Visionvs reality: what‘s working and what remains elusive,The role of quantum software in managing theoretical complexities

Unit 4: Quantum Communication and Computing – Theoretical Perspective
Quantum vs Classical Information, Basics of Quantum Communication, Quantum Key Distribution(QKD),Role of Entanglement in Communication, The Idea of the Quantum Internet – Secure Global Networking, Introduction to Quantum Computing, Quantum Parallelism (Many States at Once),Classical vs Quantum Gates, Challenges: Decoherence and Error Correction, Real-World Importance and Future Potential

Unit 5: Applications, Use Cases, and the Quantum Future
Real-world application domains: Healthcare (drug discovery),Material science, Logistics and optimization, Quantum sensing and precision timing, Industrial case studies: IBM, Google, Microsoft, PsiQuantum, Ethical, societal, and policy considerations, Challenges to adoption: cost, skills,

standardization, Emerging careers in quantum: roles, skillsets, and preparation pathways, Educational and research landscape – India's opportunity in the global quantum race

Textbooks:

1. Michael A. Nielsen, Isaac L. Chuang, Quantum Computation and Quantum Information, Cambridge University Press, 10th Anniversary Edition, 2010.
2. Eleanor Rieffel and Wolfgang Polak, Quantum Computing: A Gentle Introduction, MIT Press,2011.
3. Chris Bernhardt, Quantum Computing for Everyone, MIT Press, 2019.

Reference Books:

1. David McMahon, Quantum Computing Explained, Wiley, 2008.
2. Phillip Kaye, Raymond Laflamme, Michele Mosca, An Introduction to Quantum Computing, Oxford University Press, 2007.
3. Scott Aaronson, Quantum Computing Since Democritus, Cambridge University Press, 2013.
4. **Alastair I.M. Rae**, Quantum Physics: A Beginner's Guide, Oneworld Publications, Revised Edition, 2005.
5. **Eleanor G. Rieffel, Wolfgang H. Polak**, Quantum Computing: A Gentle Introduction, MITPress, 2011.
6. **Leonard Susskind, Art Friedman**, Quantum Mechanics: The Theoretical Minimum, Basic Books, 2014.
7. **Bruce Rosenblum, Fred Kuttner**, Quantum Enigma: Physics Encounters Consciousness, Oxford University Press, 2nd Edition, 2011.
8. **GiulianoBenenti, GiulioCasati, GiulianoStrini**, Principles of Quantum Computation and Information, Volume I: Basic Concepts, World Scientific Publishing, 2004.
9. **K.B. Whaley et al.**, Quantum Technologies and Industrial Applications: European Roadmap and Strategy Document, Quantum Flagship, European Commission, 2020.
10. **Department of Science & Technology (DST), Government of India**, National Mission on Quantum Technologies & Applications – Official Reports and Whitepapers, MeitY/DST Publications, 2020 onward.

Online Learning Resources:

- IBM Quantum Experience and Qiskit Tutorials
- Coursera – Quantum Mechanics and Quantum Computation by UC Berkeley
- edX – The Quantum Internet and Quantum Computers
- YouTube – Quantum Computing for the Determined by Michael Nielsen
- Qiskit Textbook – IBM Quantum

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	OBJECT ORIENTED ANALYSIS AND DESIGN		L	T	P	C
23BTCS12a	(Professional Elective –I)		3	0	0	3
Year	III	Semester	I			

Course Objectives:

1. Describe the activities in the different phases of the object-oriented development lifecycle.
2. Understand the concepts of object-oriented model with the E-R and EER models.
3. Model a real-world application by using UML diagram.
4. Design architectural modelling.
5. Describing an application of UML.

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

CO1: Understand the importance and principles of modeling in software development and explain the conceptual model and architecture of UML within the software development life cycle.

CO2: Create and interpret basic and advanced structural models, including class and object diagrams, using appropriate UML construct and relationships.

CO3: Develop use case, interaction, and activity diagrams to model and visualize the dynamic behavior of a system.

CO4: Model complex system behaviors using state machines, events, and signals, and represent system architecture with component and deployment diagrams.

CO5: Apply UML modeling techniques to real-world problems using patterns, frameworks, and artifact diagrams through a comprehensive case study.

UNIT – I **9 Hrs**

Introduction to UML: Importance of modelling, principles of modelling, object-oriented modelling, conceptual model of the UML, Architecture, Software Development Life Cycle.

UNIT – II **9 Hrs**

Basic Structural Modelling: Classes, Relationships, common Mechanisms, and diagrams.

AdvancedStructural Modelling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages.

Class & Object Diagrams: Terms, concepts, modelling techniques for Class & Object Diagrams.

UNIT – III **9 Hrs**

Basic Behavioural Modelling-I: Interactions, Interaction diagrams.

Basic Behavioural Modelling-II: Use cases, Use case Diagrams, Activity Diagrams.

UNIT – IV **9 Hrs**

Advanced Behavioural Modelling: Events and signals, state machines, processes and Threads, time and
space, state chart diagrams.

Architectural Modelling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT – V **9 Hrs**

Patterns and Frameworks, Artifact Diagrams. Case Study: The Unified Library application.

Text Books:

1. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modelling Language User Guide, Pearson Education 2nd Edition.
2. Object-Oriented Analysis and Design with the Unified Process By John W. Satzinger, Robert BJackson and Stephen D Burd, Cengage Learning.

Reference Books:

1. Meilir Page-Jones: Fundamentals of Object-Oriented Design in UML, Pearson Education.
2. Pascal Roques: Modelling Software Systems Using UML2, WILEY-Dreamtech India Pvt. Ltd.
3. AtulKahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.

4. Mark Priestley: Practical Object-Oriented Design with UML, TMH.
5. Appling UML and Patterns: An introduction to Object – Oriented Analysis and Design andUnified Process, Craig Larman, Pearson Education.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	CYBER SECURITY		L	T	P	C
23BTCS18b	Professional Elective-I		3	0	0	3
Year	III	Semester	I			

Course Objectives:
The course is designed to provide awareness on different cyber-crimes, cyber offenses, tools and methods used in cybercrime.

Course Outcomes: After completion of the course, students will be able to
CO1: Classify the cybercrimes and understand the Indian ITA 2000
CO2: Analyse the vulnerabilities in any computing system and find the solutions
CO3: Predict the security threats of the future
CO4: Investigate the protection mechanisms
CO5: Design security solutions for organizations

UNIT I Introduction to Cybercrime **Lecture 8Hrs**
Introduction, Cybercrime, and Information Security, Who are Cybercriminals, Classifications of Cybercrimes, And Cybercrime: The legal Perspectives and Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes.

UNIT II Cyber Offenses: How Criminals Plan Them **Lecture 9Hrs**
Introduction, How Criminals plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector, Cloud Computing

UNIT III Cybercrime: Mobile and Wireless Devices **Lecture 9Hrs**
Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies an Measures in Mobile Computing Era, Laptops.

UNIT IV Tools and Methods Used in Cybercrime **Lecture 8Hrs**
Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Keyloggers and Spywares,Virus and Worms, Trojan Horse and Backdoors, Steganography, DoS and DDoS attacks, SQL Injection, Buffer Overflow.

UNIT V Cyber Security: Organizational Implications **Lecture 8Hrs**
Introduction, Cost of Cybercrimes and IPR issues, Web threats for Organizations, Security and Privacy Implications, Social media marketing: Security Risks and Perils for Organizations, Social Computing and the associated challenges for Organizations.

Text books:
1. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Nina Godbole and Sunil Belapure, Wiley INDIA.

Reference Books:
1. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.
2. Introduction to Cyber Security, Chwan- Hwa(john) Wu,J. David Irwin.CRC Press T&F Group

Online Learning Resources:
http://nptel.ac.in/courses/106105031/40
http://nptel.ac.in/courses/106105031/39
http://nptel.ac.in/courses/106105031/38

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Artificial Intelligence (Professional Elective-I)		L	T	P	C
23BTCS08T			3	0	0	3
Year	III	Semester	I			

Pre-requisite:
- Knowledge in Computer Programming.
- A course on ―Mathematical Foundations of Computer Science‖.
- Background in linear algebra, data structures and algorithms, and probability.

Course Objectives:
- The student should be made to study the concepts of Artificial Intelligence.
- The student should be made to learn the methods of solving problems using Artificial Intelligence.
- The student should be made to introduce the concepts of Expert Systems.
- To understand the applications of AI, namely game playing, theorem proving, and machine learning.
- To learn different knowledge representation techniques

Course Outcomes:

CO1: Understand and explain the foundational concepts, historical evolution, and agent-based architecture in Artificial Intelligence.

CO2: Apply uninformed and informed search strategies, including game-playing algorithms, to solve AI-based problems efficiently.

CO3: Demonstrate various knowledge representation techniques and apply reasoning methods under uncertainty using probabilistic models.

CO4: Analyze logical inference techniques and apply fundamental machine learning approaches for problem-solving and decision-making.

CO5: Describe the structure, functionality, and applications of expert systems and evaluate real-world expert systems like MYCIN and XCON.

UNIT-I **Lecture 9Hrs**

Introduction: AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

UNIT-II **Lecture 9Hrs**

Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A* ,AO* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

UNIT–III: **Lecture 8Hrs**

Representation of Knowledge: Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability, Baye's probabilistic interferences and dempstershafer theory.

UNIT-IV **Lecture 8Hrs**

Logic concepts: First order logic, Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning..

UNIT-V **Lecture 10Hrs**

Expert Systems: Architecture of expert systems, Roles of expert systems – Knowledge Acquisition Meta knowledge Heuristics. Typical expert systems – MYCIN, DART, XCON: Expert systems shells.

Textbooks:

1. S. Russel and P. Norvig, ―Artificial Intelligence – A Modern Approach‖, Second Edition, Pearson Education.
2. Kevin Night and Elaine Rich, Nair B., ―Artificial Intelligence (SIE)‖, McGraw Hill

Reference Books:

1. David Poole, Alan Mack worth, Randy Goebel,‖ Computational Intelligence: a logical approach‖, Oxford University Press.
2. G. Luger, ―Artificial Intelligence: Structures and Strategies for complex problem solving‖, Fourth Edition, Pearson Education.
3. J. Nilsson, ―Artificial Intelligence: A new Synthesis‖, Elsevier Publishers.
4. Artificial Intelligence, SarojKaushik, CENGAGE Learning.

Online Learning Resources:

1. https://ai.google/
2. https://swayam.gov.in/nd1_noc19_me71/preview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	MICROPROCESSORS ANDMICROCONTROLLERS		L	T	P	C
23BTEC11T	(Professional Elective-I)		3	0	0	3
Year	III	Semester	I			

Course Objectives:
1. To comprehend the architecture, operation, and configurations of the 8086 microprocessors.
2. To get familiar with 8086 programming concepts, instruction set, and assembly language development tools.
3. To study the interfacing of 8086 with memory, peripherals, and controllers for various applications.
4. To learn the architecture, instruction set, and programming of the 8051 microcontrollers.
5. To understand microcontroller interfacing techniques, peripheral programming, and processor comparisons.

Course Outcomes: At the end of this course, the students will be able to
CO1: Gain knowledge on the architecture, operation, and configurations of the 8086 microprocessors.
CO2: Get familiar with 8086 programming concepts, instruction set, and assembly languagedevelopment tools.
CO3: Know the interfacing of 8086 with memory, peripherals, and controllers for variousapplications.
CO4: Learn the architecture, instruction set, and programming of the 8051 microcontrollers.
CO5: Understand microcontroller interfacing techniques, peripheral programming, and processorcomparisons.

UNIT-I
8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internalarchitecture, bus interfacing unit, execution unit, interrupts and interrupt response, 8086 system timing, minimum mode and maximum mode configuration.

UNIT-II
8086 Programming: Program development steps, instructions, addressing modes, assembler directives,writing simple programs with an assembler, assembly language program development tools.

UNIT-III
8086 Interfacing: Semiconductor memories interfacing (RAM, ROM), Intel 8255 programmable peripheral interface, Interfacing switches and LEDS, Interfacing seven segment displays, software and hardware interrupt applications, Intel 8251 USART architecture and interfacing, Intel 8237a DMA controller, stepper motor, A/D and D/A converters, Need for 8259 programmable interrupt controllers.

UNIT-IV
Microcontroller - Architecture of 8051 – Special Function Registers (SFRs) - I/O Pins Ports and Circuits - Instruction set - Addressing modes - Assembly language programming.

UNIT-V
Interfacing Microcontroller - Programming 8051 Timers - Serial Port Programming - Interrupts Programming – LCD & Keyboard Interfacing - ADC, DAC & Sensor Interfacing - External Memory Interface- Stepper Motor and Waveform generation - Comparison of Microprocessor, Microcontroller, PIC and ARM processors

Textbooks:
1. Microprocessors and Interfacing – Programming and Hardware by Douglas V Hall, SSSP Rao, TataMcGraw Hill Education Private Limited, 3rdEdition,1994.
2. K M Bhurchandi, A K Ray, Advanced Microprocessors and Peripherals, 3rd edition, McGraw HillEducation, 2017.
3. Raj Kamal, Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd Edition,
Pearson, 2012.

References:

1. Ramesh S Gaonkar, Microprocessor Architecture Programming and Applications with the 8085, 6thedition, Penram International Publishing, 2013.
2. Kenneth J. Ayala, The 8051 Microcontroller, 3rd edition, Cengage Learning, 2004.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	DATA WAREHOUSING &DATA MINING (Professional Elective-I)		L	T	P	C
23BTIT02T			3	0	0	3
Year	III	Semester	I			

Course Objective:
• Familiarize with mathematical foundations of data mining tools.
• Introduce classical models and algorithms in data warehouses and data mining.
• Investigate the kinds of patterns that can be discovered by association rule mining, classification and clustering.
• Explore data mining techniques in various applications like social, scientific and environmental context.

Course Outcomes: Upon completion of the course, the students should be able to:
CO1: Design a Data warehouse system and perform business analysis with OLAP tools.
CO2: Apply suitable pre-processing and visualization techniques for data analysis.
CO3: Apply frequent pattern and association rule mining techniques for data analysis.
CO4: Design appropriate classification and clustering techniques for data analysis.
CO5: Infer knowledge from raw data.

UNIT- I: **Lecture 9Hrs**
Basic Concepts – Data Warehousing Components – Building a Data Warehouse – DatabaseArchitectures for Parallel Processing – Parallel DBMS Vendors – Multidimensional Data Model – Data Warehouse Schemas for Decision Support, Concept Hierarchies -Characteristics of OLAP Systems – Typical OLAP Operations, OLAP and OLTP.

UNIT- II: **Lecture 9Hrs**
Introduction to Data Mining Systems – Knowledge Discovery Process – Data Mining Techniques – Issues – applications- Data Objects and attribute types, Statistical description of data, Data Preprocessing – Cleaning, Integration, Reduction, Transformation and discretization, DataVisualization, Data similarity and dissimilarity measures.

UNIT- III: **Lecture 8 Hrs**
Mining Frequent Patterns, Associations and Correlations – Mining Methods- Pattern Evaluation Method – Pattern Mining in Multilevel, Multi-Dimensional Space – Constraint Based Frequent Pattern Mining, Classification using Frequent Patterns.

UNIT- IV: **Lecture 9Hrs**
Decision Tree Induction – Bayesian Classification – Rule Based Classification – Classification by Back Propagation – Support Vector Machines — Lazy Learners – Model Evaluation and SelectionTechniques to improve Classification Accuracy. Clustering Techniques – Cluster analysis-Partitioning Methods – Hierarchical Methods – Density Based Methods – Grid Based Methods – Evaluation ofclustering – Clustering high dimensional data- Clustering with constraints, Outlier analysis-outlier detection methods.

UNIT- V: WEKA TOOL **Lecture 8Hrs**
Datasets – Introduction, Iris plants database, Breast cancer database, Auto imports database – Introduction to WEKA, The Explorer – Getting started, Exploring the explorer, Learning algorithms, Clustering algorithms, Association–rule learners.

TEXT BOOK:
1.Jiawei Han and MichelineKamber, ―Data Mining Concepts and Techniques, Third Edition,Elsevier, 2012.
REFERENCES:
1.Alex Berson and Stephen J.Smith, ―Data Warehousing, Data Mining & OLAP‖, Tata McGraw – Hill

Edition, 35th Reprint 2016.
2.K.P. Soman, ShyamDiwakar and V. Ajay, ―Insight into Data Mining Theory and Practice, EasternEconomy Edition, Prentice Hall of India, 2006.
3.IanH.Witten and Eibe Frank, ―Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, Second Edition.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	ADVANCED JAVA LAB		L	T	P	C
23BTIT01P			0	0	3	1.5
Year	III	Semester	I			

COURSE OBJECTIVES: ·
- To understand the fundamentals of web programming and client side scripting. ·
- To learn server side development using servlets, web sockets. ·
- To learn the Spring framework and build applications using Spring. ·
- To learn and implement the concept of Java Persistence API. ·
- To learn the advanced client side scripting and framework.

LIST OF EXPERIMENTS:

1. Create an event registration application using java script. It should implement different widgets for registration form and registered records view using tabs. It should perform the form validation.
2. Create a java script application in an Object Oriented way using Classes and Modules. It should also use browser storage for persistence.
3. Build a web application using Gradle. The server side of the application should implement RESTful APIs using Servlet and do necessary logging. The client side of the application should be a single page application which consumes the RESTful APIs through AJAX.
4. Build a chat application using WebSocket.
5. Create a Spring MVC application. The application should handle form validation, file upload, session tracking.
6. Implement a RESTful Spring Boot application using Spring REST, Spring Security and Spring Cache.
7. Design a system using JPA and Hibernate. The system should have multiple entities and relationships between the entities. The database schema should be generated through Hibernate. Provide RESTful endpoints for CRUD operations for the defined entities. Also, support pagination and searching using JPA's JPQL and Criteria API.
8. Create a Spring RESTful Application with Spring Data JPA. Support pagination and searching using Specifications.
9. Create a React application with different components and interactions between the components.
10. Develop a full-stack application using React and Spring. Make use of Spring REST, Spring Security, Spring Data JPA, Hibernate, Spring Boot, Gradle and ReactJS state and component mechanism.

REFERENCES

1. David Flanagan, "Java Script: The Definitive Guide", O'Reilly Media, Inc, 7th Edition, 2020
2. Matt Frisbie, "Professional JavaScript for Web Developers", Wiley Publishing, Inc, 4th Edition, ISBN: 978-1-119-36656-0, 2019
3. Alex Banks, Eve Porcello, "Learning React", O'Reilly Media, Inc, 2nd Edition, 2020
4. David R. Heffelfinger, "Java EE 8 Application Development", Packt Publishing, First edition2017
5. Benjamin Muschko, "Gradle in Action", Manning Publications, First edition2014
6. IulianaCosmina, Rob Harrop, Chris Schaefer, Clarence Ho, "Pro Spring 5: An In-Depth Guide to the Spring Framework and Its Tools", Apress, Fifth edition2017

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	COMPUTER NETWORKS & INTERNETPROTOCOLS LAB		L	T	P	C
23BTCS09P			0	0	3	1.5
Year	III	Semester	I			

Course Objectives:
- To understand the working principle of various communication protocols.
- To understand the network simulator environment and visualize a network topologyand observe its performance
- To analyze the traffic flow and the contents of protocol frames.
- Familiarize with the applications of Internet.

Course Outcomes:
CO1: To understand the working principle of various communication protocols.
CO2: To understand the network simulator environment and visualize a network topologyand observe its performance.
CO3: To analyze the traffic flow and the contents of protocol frames.
CO4: Critique the existing routing protocols

List of Experiments:
1. Implement the data link layer framing methods such as character, character-stuffing and bitstuffing.
2. Write a program to compute CRC code for the polynomials CRC-12, CRC-16 and CRC CCIP
3. Develop a simple data link layer that performs the flow control using the sliding windowprotocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkastra‘s algorithm to compute the shortest path through a network
5. Take an example subnet of hosts and obtain a broadcast tree for the subnet.
6. Implement distance vector routing algorithm for obtaining routing tables at each node.
7. Implement data encryption and data decryption
8. Write a program for congestion control using Leaky bucket algorithm.
9. Write a program for frame sorting technique used in buffers.
10. Programs using Wireshark
i. Packet Capture Using Wire shark
ii. Starting Wire shark
iii. Viewing Captured Traffic
iv. Analysis and Statistics & Filters.
11. How to run Nmap scan
12. Operating System Detection using Nmap
13. Do the following using NS2 Simulator
i. NS2 Simulator-Introduction
ii. Simulate to Find the Number of Packets Dropped
iii. Simulate to Find the Number of Packets Dropped by TCP/UDP
iv. Simulate to Find the Number of Packets Dropped due to Congestion
v. Simulate to Compare Data Rate& Throughput.
vi. Simulate to Plot Congestion for Different Source/Destination
vii. Simulate to Determine the Performance with respect to transmission of Packets

Text Books:
1. Andrew S.Tanenbaum, David J.Wetherall, Computer Networks, 6th Edition, PEARSON.
2. James F.Kurose, Keith W. Ross, Computer Networking: A Top-Down 6th edition, Pearson, 2019.

3. Computer Networks: A Systems Approach-Bruce Davie, VMware-Larry Peterson, Princeton University-2019.

Reference Books:

1. Computer Networks–B. K. MathanNagan, T. Mahalakshmi- Charulatha PublicationsPrivate Limited-2019.
2. Computer Networks-Dr.Amol V. DhumaneNitin N. Sakhare-NiraliPrakashan Publishers-2024
3. Data Communications and Networking with TCPIP Protocol Suite-Behrouz A. Forouzan McGraw Hill-6th Edition

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	FULL STACK DEVELOPMENT – I		L	T	P	C
23BTCS02S	(Skill Enhancement Course)		0	1	2	2
Year	III	Semester	I			

Course Objectives: The main objectives of the course are to
- Make use of HTML elements and their attributes for designing static web pages
- Build a web page by applying appropriate CSS styles to HTML elements
- Experiment with JavaScript to develop dynamic web pages and validate forms

Course Outcomes: At the end of the Course, the student will be able to
CO1: Design Websites.
CO2: Apply Styling to web pages.
CO3: Make Web pages interactive.
CO4: Design Forms for applications.
CO5: Choose Control Structure based on the logic to be implemented.

Experiments covering the Topics:
- Lists, Links and Images
- HTML Tables, Forms and Frames
- HTML 5 and Cascading Style Sheets, Types of CSS
- Selector forms
- CSS with Color, Background, Font, Text and CSS Box Model
- Applying JavaScript - internal and external, I/O, Type Conversion
- JavaScript Conditional Statements and Loops, Pre-defined and User-defined Objects
- JavaScript Functions and Events

Sample Experiments:

1. Lists, Links and Images

a. Write a HTML program, to explain the working of lists.
   Note: It should have an ordered list, unordered list, nested lists and ordered list in an unordered list and definition lists.
b. Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.
c. Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.
d. Write a HTML program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a full sized version of the image. Create an image gallery using this technique

2. HTML Tables, Forms and Frames

a. Write a HTML program, to explain the working of tables. (use tags: <table>, <tr>, <th>, <td> and attributes: border, rowspan, colspan)
b. Write a HTML program, to explain the working of tables by preparing a timetable. (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.).
c. Write a HTML program, to explain the working of forms by designing Registration form. (Note: Include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select>&<option>

tags, <text area> and two buttons ie: submit and reset. Use tables to provide a better view).

d. Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame → image, second frame → paragraph, third frame → hyperlink. And also make sure of using "no frame" attribute such that frames to be fixed).

3. HTML 5 and Cascading Style Sheets, Types of CSS

a. Write a HTML program, that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, , tags.

b. Write a HTML program, to embed audio and video into HTML web page.

c. Write a program to apply different types (or levels of styles or style specification formats) - inline, internal, external styles to HTML elements. (identify selector, property and value).

4. Selector forms

a. Write a program to apply different types of selector forms

i. Simple selector (element, id, class, group, universal)
ii. Combinator selector (descendant, child, adjacent sibling, general sibling)
iii. Pseudo-class selector
iv. Pseudo-element selector
v. Attribute selector

5. CSS with Color, Background, Font, Text and CSS Box Model

a. Write a program to demonstrate the various ways you can reference a color in CSS.

b. Write a CSS rule that places a background image halfway down the page, tilting it horizontally. The image should remain in place when the user scrolls up or down.

c. Write a program using the following terms related to CSS font and text:
i. font-size ii. font-weight iii. font-style iv. text-decoration v. text-transformation vi. text-alignment

d. Write a program, to explain the importance of CSS Box model using
i. Content ii. Border iii. Margin iv. padding

6. Applying JavaScript - internal and external, I/O, Type Conversion

a. Write a program to embed internal and external JavaScript in a web page.

b. Write a program to explain the different ways for displaying output.

c. Write a program to explain the different ways for taking input.

d. Create a webpage which uses prompt dialogue box to ask a voter for his name and age. Display the information in table format along with either the voter can vote or not

7. Java Script Pre-defined and User-defined Objects

a. Write a program using document object properties and methods.
b. Write a program using window object properties and methods.
c. Write a program using array object properties and methods.
d. Write a program using math object properties and methods.
e. Write a program using string object properties and methods.
f. Write a program using regex object properties and methods.
g. Write a program using date object properties and methods.
h. Write a program to explain user-defined object by using properties,methods, accessors, constructors and display.

8. Java Script Conditional Statements and Loops

a. Write a program which asks the user to enter three integers, obtains the numbers from the

user and outputs HTML text that displays the larger number followed by the words "LARGER NUMBER" in an information message dialog. If the numbers are equal, output HTML text as "EQUAL NUMBERS".

b. Write a program to display week days using switch case.

c. Write a program to print 1 to 10 numbers using for, while and do-while loops.

d. Write a program to print data in object using for-in, for-each and for-of loops

e. Develop a program to determine whether a given number is an 'ARMSTRONG NUMBER' or not. [Eg: 153 is an Armstrong number, since sum of the cube of the digits is equal to the number i.e.,13 + 53+ 33 = 153]

f. Write a program to display the denomination of the amount deposited in the bank in terms of 100's, 50's, 20's, 10's, 5's, 2's & 1's. (Eg: If deposited amount is Rs.163, the output should be 1-100's, 1-50's, 1- 10's, 1-2's & 1-1's)

9. Java Script Functions and Events

a. Design a appropriate function should be called to display
   i. Factorial of that number
   ii. Fibonacci series up to that number
   iii. Prime numbers up to that number
   iv. Is it palindrome or not

b. Design a HTML having a text box and four buttons named Factorial, Fibonacci, Prime, and Palindrome. When a button is pressed an appropriate function should be called to display
   i. Factorial of that number
   ii. Fibonacci series up to that number
   iii. Prime numbers up to that number
   iv. Is it palindrome or not

c. Write a program to validate the following fields in a registration page
   i. Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
   ii. Mobile (only numbers and length 10 digits)
   iii. E-mail (should contain format like xxxxxxx@xxxxxx.xxx)

10. JavaScript Database Connectivity

a. Introduction to server-side JavaScript with Node.js

b. Connecting JavaScript applications to MySQL and Mongo DB databases

Text Books:

1. John Dean, Web Programming with HTML5, CSS and JavaScript, Jones & Bartlett Learning, 2019.

Reference Books:

1. Programming the World Wide Web, 7th Edition, Robet W Sebesta, Pearson, 2013.
2. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasan Subramanian, 2nd edition, APress, O'Reilly.

Online Learning Resources:

1. https://www.w3schools.com/html
2. https://www.w3schools.com/css
3. https://www.w3schools.com/js/
4. https://www.w3schools.com/nodejs
5. https://www.w3schools.com/typescript

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	TINKERING LAB		L	T	P	C
23BTEC09P			0	0	2	1
Year	III	Semester	I			

The aim of tinkering lab for engineering students is to provide a hands-on learning environment where students can explore, experiment, and innovate by building and testing prototypes. These labs are designed to demonstrate practical skills that complement theoretical knowledge.

Course objectives: The objectives of the course are to
CO1: Encourage Innovation and Creativity
CO2: Provide Hands-on Learning and Impart Skill Development
CO3: Foster Collaboration and Teamwork
CO4: Enable Interdisciplinary Learning, Prepare for Industry andEntrepreneurship
CO5: Impart Problem-Solving mind-set

These labs bridge the gap between academia and industry, providing students with the practicalexperience. Some students may also develop entrepreneurial skills, potentially leading to startups or innovation-driven careers. Tinkering labs aim to cultivate the next generation of engineers by giving them the tools, space, and mind-set to experiment, innovate, and solve real-world challenges.

List of experiments:
1) Make your own parallel and series circuits using breadboard for any application of your choice.
2) Design and 3D print a Walking Robot
3) Design and 3D Print a Rocket.
4) Temperature & Humidity Monitoring System (DHT11 + LCD)
5) Water Level Detection and Alert System
6) Automatic Plant Watering System
7) Bluetooth-Based Door Lock System
8) Smart Dustbin Using Ultrasonic Sensor
9) Fire Detection and Alarm System
10) RFID-Based Attendance System
11) Voice-Controlled Devices via Google Assistant
12) Heart Rate Monitoring Using Pulse Sensor
13) Soil Moisture-Based Irrigation
14) Smart Helmet for Accident Detection
15) Milk Adulteration Detection System
16) Water Purification via Activated Carbon
17) Solar Dehydrator for Food Drying
18) Temperature-Controlled Chemical Reactor
19) Ethanol Mini-Plant Using Biomass
20) Smart Fluid Flow Control (Solenoid + pH Sensor)
21) Portable Water Quality Tester
22) AI Crop Disease Detection
23) AI-based Smart Irrigation
24) ECG Signal Acquisition and Plotting
25) AI-Powered Traffic Flow Prediction
26) Smart Grid Simulation with Load Monitoring
27) Smart Campus Indoor Navigator

28) Weather Station Prototype
29) Firefighting Robot with Sensor Guidance
30) Facial Recognition Dustbin
31) Barcode-Based Lab Inventory System
32) Growth Chamber for Plants
33) Biomedical Waste Alert System
34) Soil Classification with AI
35) Smart Railway Gate
36) Smart Bin Locator via GPS and Load Sensors
37) Algae-Based Water Purifier
38) Contactless Attendance via Face Recognition

- **Note:** The students can also design and implement their own ideas, apart from the listof experiments mentioned above.
- **Note:** A minimum of 8 to 10 experiments must be completed by the students.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	CLOUD COMPUTING		L	T	P	C
23BTCS16T			3	0	0	3
Year	III	Semester	II			

Course Objectives:
- To explain the evolving computer model called cloud computing.
- To introduce the various levels of services that can be achieved by cloud.
- To describe the security aspects in cloud.

Course Outcomes (CO): After completion of the course, students will be able to
CO1: Ability to create cloud computing environment
CO2: Ability to design applications for Cloud environment
CO3: Design & develop back up strategies for cloud data based on features.
CO4: Use and Examine different cloud computing services.
CO5: Apply different cloud programming model as perneed.

UNIT I Basics of Cloud computing **Lecture8Hrs**

Introduction to cloud computing: Introduction, Character is tics of cloud computing, Cloud Models, Cloud Services Examples, Cloud Based services and applications

Cloud concepts and Technologies: Virtualization, Load balancing, Scalability and Elasticity, Deployment, Replication, Monitoring, Software defined, Network function virtualization, Map Reduce, Identity and Access Management, services level Agreements, Billing.

Cloud Services and Plat forms: Compute Services, Storage Services, Data, base Services, Application services, Content delivery services, Analytics Services, Deployment and Management Services, Identity and Access Management services, Open Source Private Cloud software.

UNIT II Hadoop and Python **Lecture 9Hrs**

Hadoop Map Reduce: Apache Hadoop, Hadoop Map Reduce Job Execution, Hadoop Schedulers, Hadoop Clusterset up.

Cloud Application Design: Reference Architecture for Cloud Applications, Cloud Application Design Methodologies, Data Storage Approaches.

UNIT III Python for Cloud computing **Lecture 8Hrs**

Python for Cloud: Python for Amazon web services, Python for Google Cloud Platform, Python for windows Azure, Python for Map Reduce, Python packages of Interest, Python web Application Frame work, Designing a REST ful web API.

Cloud Application Developmentin Python: Design Approaches, Image Processing APP, Document Storage App, Map Reduce App, Social Media Analytics App.

UNIT IV Big data, Multimedia and Tuning **Lecture8Hrs**

Big Data Analytics: Introduction, Clustering Big Data, Classification of Big data Recommendation of Systems.

Multimedia Cloud: Introduction, Case Study: Live video Streaming App, Streaming Protocols, case Study: Video Trans coding App.

Cloud Application Bench marking and Tuning: Introduction, Work load Character is tics,Application Performance Metrics, Design Considerations for a Bench marking Methodology, Bench marking Tools, Deployment Prototyping, Load Testing &Bottleneck Detection case Study, Hadoop bench marking case Study.

UNIT V Applications and Issues in Cloud **Lecture 9Hrs**

Cloud Security: Introduction, CSA Cloud Security Architecture, Authentication, Authorization, Identity Access Management, Data Security, Key Management, Auditing.

Cloud for Industry, Health care & Education: Cloud Computing for Health care, Cloud computing for Energy Systems, Cloud Computing for Transportation Systems, Cloud Computing for

Manufacturing Industry, Cloud computing for Education.

Migrating in to a Cloud: Introduction, Broad Approaches to migrating into the cloud, the seven– step model of migration in to a cloud.

Organizational readiness and Change Management in The Cloud Age: Introduction, Basic concepts of Organizational Readiness, Drivers for changes: A frame work to comprehend the competitive environment, common change management models, change management maturity models, Organizational readiness self– assessment.

Legal Issues in Cloud Computing: Introduction, Data Privacy and security Issues, cloud contracting models, Jurisdictional issues raised by virtualization and at a location, commercial and business considerations, Special Topics.

Text books:

1. Cloud Computing Ahands - on Approach ǁBy ArshdeepBahga, Vijay Madisetti, Universities Press, 2016
2. Cloud Computing Principles and Paradigms: By RajKumarBuyya, James Broberg, AndrzejGoscinski, Wiley, 2016

Reference Books:

1. Mastering Cloud Computing by RajkumarBuyya, Christian Vecchiola, S ThamaraiSelvi, TMH
2. Cloud computing AHands-On Approach by ArshdeepBahga and Vijay Madisetti.
3. Cloud Computing: A Practical Approach, Anthony T.Velte, To by J.Velte, Robert Elsenpeter, Tata McGraw Hill, rp 2011.
4. Enterprise Cloud Computing, GautamShroff, Cambridge University Press, 2010.
5. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, George Reese,O_Reilly, SPD, rp 2011.
6. Essentials of Cloud Computing by K.Chandrasekaran. CRC Press.

Online Learning Resources:

Cloud computing – Course (nptel.ac.in)

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	CRYPTOGRAPHY & NETWORKSECURITY		L	T	P	C
23BTCS17T			3	0	0	3
Year	III	Semester	II			

Course Objectives: This course aim sat training students to master the
- The concepts of classical encryption techniques and concepts of finite fields and number theory
- Working principles and utilities of various crypto graphic algorithms including secret keycryptography, hashes, and message digests, and public key algorithms
- Design issues and working principles of various authentication protocols, PKIstandards
- Various secure communication standards including Kerberos, IPsec, TLS and email
- Concepts of crypto graphic utilitiesand authentication mechanismsto design secure applications

Course Outcomes: After completion of the course, students will beable to
CO1: Identify information security goals, classical encryption techniques and acquire fundamental knowledge on the concepts of in it fields and number theory
CO2: Compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication
CO3: Apply the knowledge of cryptographic check sums and evaluate the performance of different message digestalgorithms for verifying the integrity of varying message sizes.
CO4: Demonstrate the ability to apply **user authentication principles** including **Kerberos** for secure authentication
CO5: Gain proficiency in securing web communications using **TLS** and **HTTPS**, manage secure remote access with **SSH**, and design **firewall policies**

UNIT-I **Lecture9Hrs**
Computer and Network Security Concepts: Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security, **Classical Encryption Techniques:** SymmetricCipher Model, Substitution Techniques, Transposition Techniques, Steganography, **Block Ciphers:** Traditional Block Cipher Structure, The Data Encryption Standard, Advanced Encryption Standard: AES Structure, AES Transformation Functions

UNIT II **Lecture 9Hrs**
Number Theory: The Euclidean Algorithm, Modular Arithmetic, Fermat‘s and Euler‘s Theorems, The Chinese Remainder Theorem, Discrete Logarithms, Finite Fields: Finite Fields of the Form GF(p), Finite Fields of the Form GF(2n) .
Public Key Cryptography: Principles, Public Key Cryptography Algorithms, RSA Algorithm, Diffie Hellman Key Exchange, Elliptic Curve Cryptography.

UNIT-III **Lecture9Hrs**
Cryptographic Hash Functions: Application of Cryptographic Hash Functions, Requirements&Security, Secure Hash Algorithm, Message Authentication Functions, Requirements &Security,HMAC &CMAC.
Digital Signatures: NIST Digital Signature Algorithm, Distribution of Public Keys, X.509 Certificates, Public- Key Infrastructure

UNITIV **Lecture9Hrs**
UserAuthentication:RemoteUserAuthenticationPrinciples,Kerberos.
ElectronicMailSecurity: PrettyGoodPrivacy(PGP) andS/MIME.
IPSecurity:IPSecurityOverview, IP SecurityPolicy, EncapsulatingSecurityPayload,CombiningSecurityAssociations,InternetKeyExchange.

UNIT V **Lecture8Hrs**

Transport Level Security: Web Security Requirements, Transport Layer Security (TLS), HTTPS,Secure Shell (SSH)

Fire walls: Fire wall Characteristics and Access Policy, Types of Fire walls, Fire wall Location and Configurations.

Text books:

1) Cryptography and Network Security – William Stallings, Pearson Education, 8 thEdition.

2) Cryptography, Network Security and Cyber Laws–Bernard Menezes, Cengage Learning, 2010 edition.

Reference Books:

1) Cryptography and Network Security-BehrouzA Forouzan, DebdeepMukhopadhyaya, Mc- Graw Hill, 3rdEdition, 2015.

2) Network Security Illustrated, Jason Albanese and Wes Sonnenreich, MGH Publishers, 2003.

Online Learning Resources:

1) https://nptel.ac.in/courses/106/105/106105031/lecture

2)https://nptel.ac.in/courses/106/105/106105162/ lecture by Dr. SouravMukhopadhyayIIT Kharagpur[VideoLecture]

3) https://www.mitel.com/articles/web-communication-cryptography-and-networksecurity web articles by Mitel Power Connections

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	MACHINE LEARNING		L	T	P	C
23BTAI02T			3	0	0	3
Year	III	Semester	II			

Course Objectives: The objectives of the course are

- Define machine learning and its different types (supervised and unsupervised) and understand their applications.
- Apply supervised learning algorithms including decision trees and k-nearest neighbors (k-NN).
- Implement unsupervised learning techniques, such as K-means clustering.

Course Outcomes:

CO1: Identify machine learning techniques suitable for a given problem. (L3)
CO2: Solve real-world problems using various machine learning techniques. (L3)
CO3: Apply Dimensionality reduction techniques for data preprocessing. (L3)
CO4: Explain what is learning and why it is essential in the design of intelligent machines. (L2)
CO5: Evaluate Advanced learning models for language, vision, speech, decision making etc. (L5)

UNIT-I: Introduction to Machine Learning: Evolution of Machine Learning, Paradigms for ML, Learning by Rote, Learning by Induction, Reinforcement Learning, Types of Data, Matching, Stages in Machine Learning, Data Acquisition, Feature Engineering, Data Representation, Model Selection, Model Learning, Model Evaluation, Model Prediction, Search and Learning, Data Sets.

UNIT-II: Nearest Neighbor-Based Models: Introduction to Proximity Measures, Distance Measures, Non-Metric Similarity Functions, Proximity Between Binary Patterns, Different Classification Algorithms Based on the Distance Measures ,K-Nearest Neighbor Classifier, Radius Distance Nearest Neighbor Algorithm, KNN Regression, Performance of Classifiers, Performance of Regression Algorithms.

UNIT-III: Models Based on Decision Trees: Decision Trees for Classification, Impurity Measures, Properties, Regression Based on Decision Trees, Bias–Variance Trade-off, Random Forests for Classification and Regression. The Bayes Classifier: Introduction to the Bayes Classifier, Bayes'Rule and Inference, The Bayes Classifier and its Optimality, Multi-Class Classification | Class Conditional Independence and Naive Bayes Classifier (NBC)

UNIT-IV: Linear Discriminants for Machine Learning: Introduction to Linear Discriminants, Linear Discriminants for Classification, Perceptron Classifier, Perceptron Learning Algorithm, Support Vector Machines, Linearly Non-Separable Case, Non-linear SVM, Kernel Trick, Logistic Regression, Linear Regression, Multi-Layer Perceptrons (MLPs), Back propagation for Training an MLP.

UNIT-V: Clustering : Introduction to Clustering, Partitioning of Data, Matrix Factorization | Clustering of Patterns, Divisive Clustering, Agglomerative Clustering, Partitional Clustering, K-Means Clustering, Soft Partitioning, Soft Clustering, Fuzzy C-Means Clustering, RoughClustering, Rough K-Means Clustering Algorithm, Expectation Maximization-Based Clustering, Spectral Clustering.

Textbooks:

1. Machine Learning Theory and Practice‖, M N Murthy, V S Ananthanarayana, Universities Press (India), 2024

Reference Books:

1. Machine Learning‖, Tom M. Mitchell, McGraw-Hill Publication, 2017
2. Machine Learning in Action‖,Peter Harrington, DreamTech
3. Introduction to Data Mining‖, Pang-Ning Tan, Michel Stenbach, Vipin Kumar, 7th Edition, 2019.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	SOFTWARE TESTING METHODOLOGIES		L	T	P	C
23BTCS18a	(Professional Elective-II)		3	0	0	3
Year	III	Semester	II			

Course Objectives:

- To study the fundamental concepts of software testing which includes objectives, process, criteria, strategies, and methods.
- To discuss various software testing types and levels of testing like black and white box testing along with levels unit test, integration, regression, and system testing.
- It also helps to learn the types of bugs, testing levels with which the student can very well identify a bug and correct as when it happens.
- It provides knowledge on transaction flow testing and data flow testing techniques so that the flow of the program is tested as well.
- To learn the domain testing, path testing and logic based testing to explore the testing process easier.

Course Outcomes:

CO1: Understand the purpose of software testing, classify types of bugs, and apply path testing techniques using flow graphs and path predicates.
CO2: Apply transaction flow and dataflow testing techniques to identify defects based on variable usage and control flow.
CO3: Perform domain testing by analyzing input domains, domain boundaries, and their impact on software behavior and testability.
CO4: Utilize path expressions, regular expressions, and logic-based techniques such as decision tables and KV charts to derive test cases.
CO5: Apply state-based testing using state graphs and transition testing, and use graph matrices for analyzing software control structures.

UNIT-I **Lecture9Hrs**

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

UNIT-II **Lecture 8Hrs**

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

UNIT-III **Lecture8Hrs**

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

UNIT-IV **Lecture9Hrs**

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

UNIT-V **Lecture9Hrs**

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

TEXT BOOKS

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

REFERENCES BOOKS

1. The craft of software testing – Brian Marick, Pearson Education.
2. Software Testing Techniques – SPD(Oreille)
3. Software Testing in the Real World – Edward Kit, Pearson.
4. Effective methods of Software Testing, Perry, John Wiley.
5. Art of Software Testing – Meyers, John Wiley.

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	Augmented Reality And VirtualReality		L	T	P	C
23BTCS21c	(Professional Elective –IV)		3	0	0	3
Year	III	Semester	II			

Course Objective:
The primary objective of this course is to introduce students to the foundational principles and technologies of Virtual Reality (VR) and Augmented Reality (AR), along with the key devices, modelling techniques, and interaction mechanisms involved in creating immersive environments. The course will cover the essentials of VR and AR, including hardware, software, and human perception, as well as advanced concepts such as 3D modelling, interaction design, and audio rendering. Students will gain hands-on experience in the use of VR/AR systems and explore the challenges and methodologies for building interactive virtual environments.

Course Outcomes: At the end of the Course the student will be able to:
CO1: Understand the core concepts of Virtual Reality and Augmented Reality, and their differences.
CO2: Learn about the hardware and software components required for VR and AR systems, as well asthe impact of human physiology and perception on the virtual experience.
CO3: Gain knowledge of input devices (trackers, navigation, and gesture interfaces) and outputdevices (graphics, sound displays, and haptic feedback).
CO4: Develop skills in modelling techniques, including geometric, kinematics, physical, and behavior modelling for VR and AR environments.
CO5: Explore the technologies and methodologies used to create Augmented Reality systems, including marker-based AR and AR software development.

UNIT – I **(10 Lectures)**
INTRODUCTION TO VIRTUAL REALITY (VR): Defining Virtual Reality, Key elements of virtual reality experience, Virtual Reality, Telepresence, Augmented Reality and Cyberspace.
Bird's-Eye View: Hardware, Software, Human Physiology and Perception.

UNIT-II **(10 Lectures)**
Input Devices: (Trackers, Navigation, and Gesture Interfaces): Three-dimensional position trackers, navigation and manipulation, interfaces and gesture interfaces**.**
Output Devices: Graphics displays, sound displays & haptic feedback.

UNIT-III **(10 Lectures)**
Modelling: Geometric modelling, Kinematics modelling, Physical modelling, Behaviour modelling, Model management.

UNIT-IV **(10 Lectures)**
Augmented Reality (AR): Taxonomy, Technology and Features of Augmented Reality, AR Vs VR, Challenges with AR, AR systems and functionality, Augmented Reality Methods, Visualization Techniques for Augmented Reality, Enhancing interactivity in AR Environments, Evaluating AR systems
AR software development: AR software, Camera parameters and camera calibration, Marker-based augmented reality, AR Toolkit.

UNIT-V **(10 Lectures)**
Interaction & Audio: Interaction - Motor Programs and Remapping, Locomotion, Manipulation, Social Interaction. Audio -The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering.
Interaction - Motor Programs and Remapping, Locomotion, Manipulation, Social Interaction. Audio - The Physics of Sound, The Physiology of Human Hearing, Auditory Perception, Auditory Rendering. (From Text Book2)

TEXT BOOKS:

1. Virtual Reality Technology, Second Edition, Gregory C. Burdea& Philippe Coiffet, John Wiley & Sons, Inc, 2017.
2. Virtual Reality, Steven M. LaValle, Cambridge University Press, 2016.

REFERENCES:

1. RajeshK.Maurya,*Computer Graphics with Virtual Reality System*, 3rd Edition, Wiley Publication, 2018.
2. William R. Sherman and Alan B. Craig, *Understanding Virtual Reality Interface, Application, and Design*, 2nd Edition, Morgan Kaufmann Publishers, Elsevier, 2019.
3. GrigoreC.Burdea,PhilippeCoiffet,*Virtual Reality Technology*, 2nd Edition, Wiley,2017.
4. K.S. Hale and K. M. Stanney, *Handbook on Virtual Environments*, 2nd Edition, CRC Press, 2015.

WEB REFERENCES:

1. http://vr.cs.uiuc.edu/vrbook.pdf
2. https://nptel.ac.in/courses/106/106/106106138/

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	DevOps		L	T	P	C
23BTCS18c	Professional Elective-II		3	0	0	3
Year	III	Semester	II			

Pre-requisite:
Fundamentals of software development and maintenance

Course Objectives:
- Understand collaboration and productivity by automating infrastructure and workflows
- Familiarize with continuous measuring applications performance

Course Outcomes: After completion of the course, students will be able to
CO1: Understand the origins, culture, and foundational practices of DevOps, and develop a business case and playbook for DevOps adoption.
CO2: Apply the Business Model Canvas framework and implement DevOps plays to optimize the software delivery pipeline and value stream.
CO3: Utilize DevOps strategies to foster innovation through platforms, microservices architecture, APIs, and organizational alignment.
CO4: Design scalable DevOps practices for large organizations, including culture building, team models, tool standardization, and security integration.
CO5: Lead DevOps transformation initiatives in the enterprise by promoting collaboration, trust, and structured adoption roadmaps through pilot projects.

UNIT I **Lecture 8 Hrs**

DevOps: An Overview, DevOps: Origins, DevOps: Roots, DevOps: Practices DevOps: Culture.
Adopting Dev Ops: Developing the Playbook.
Developing a Business Case for a DevOps: Developing the Business Case

UNIT II **Lecture 9 Hrs**

Completing the Business Model Canvas, Customer Segments, Value Segments, Value Propositions,Channels, Customer Relationships, Revenue Streams, Key Resources, Key Activities, Key Partnerships, Cost Structures. DevOps Plays for Optimizing the delivery Pipeline: DevOps as an optimization Exercise, Core Themes, The DevOps Plays, Specializing Core Plays

UNIT III **Lecture 8Hrs**

DevOps Plays for Driving Innovation: Optimize to Innovate, The Uber Syndrome, Innovation and the Role of Technology, Core Themes, play: Build a DevOps Platform, play: Deliver Micro services Architectures, play: DevOps an API Economy, play: Organizing for Innovation.

UNIT IV **Lecture 10 Hrs**

Scaling DevOps for the Enterprise: Core Themes, play: DevOps Center of Competency, play: Developing Culture of Innovation at Scale, play: Developing a Culture of continuous Improvement, play: Team Models for DevOps, play: Standardization of Tools and Process, play: Security Considerations for DevOps, Play: DevOps and Outsourcing.

UNIT V **Lecture 10 Hrs**

Leading DevOps Adoption in the Enterprise: Play: DevOps as a transformation Exercise, play: Developing a Culture of Collaboration and Trust, play: DevOps Thinking for the Line of Business, play: starting with Pilot Projects, Play: Rearing Unicorns on an Aircrafts Carrier. Appendix Case Study: Example DevOps Adoption Roadmap Organization Background, Roadmap Structure, Adoption Roadmap.

Text books:

1. Sanjeev Sharma, The DevOps Adoption Playbook, Published by John Wiley & Sons, Inc.2017

Reference Books:

1. Sanjeev Sharma & Bernie Coyne, DevOps for Dummies, Published by John Wiley & Sons, Inc.
2. Michael Huttermann, DevOps for Developers, Apress publishers,2012.

Online Learning Resources:

Learning Dev Ops with Terra form Infrastructure Automation Course | Udemy

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	EMBEDDED SYSTEM DESIGN		L	T	P	C
	Professional Elective-II		3	0	0	3
Year	III	Semester	II			

Course Objectives:
1. To understand the history, classification, and design process of embedded systems.
2. To explore the core components of embedded systems, including processors, memory, and I/O components.
3. To introduce onboard and external communication interfaces used in embedded systems.
4. To explain different firmware design approaches and programming techniques for embedded systems.
5. To provide an understanding of real-time operating systems and task management in embedded systems.

Course Outcomes: After completing the course, the student will be able to,
1. Classify embedded systems based on their purpose, generation, and complexity.
2. Identify and select appropriate hardware components for an embedded system design.
3. Differentiate and implement various communication protocols like I2C, SPI, and CAN.
4. Develop firmware using assembly and high-level programming languages.
5. Analyze and apply RTOS-based task scheduling and synchronization techniques.

UNIT I Introduction to Embedded Systems
History of embedded systems, Classification of embedded systems based on generation and complexity, Purpose of embedded systems, The embedded system design process-requirements, specification, architecture design, designing hardware and software, components, system integration, Applications of embedded systems, and characteristics of embedded systems.

UNIT II Typical Embedded System
Core of the embedded system-general purpose and domain specific processors, ASICs, PLDs, COTs; Memory-ROM, RAM, memory according to the type of interface, memory shadowing, memory selection for embedded systems, Sensors, actuators, I/O components: seven segment LED, relay, piezo buzzer, push button switch, other sub-systems: reset circuit, brownout protection circuit, oscillator circuit real time clock, watch dog timer.

UNIT III Communication Interface
On board communication interfaces-I2C, SPI, CAN, parallel interface; External communication interfaces-RS232 and RS485, USB, infrared, Bluetooth, Wi-Fi, ZigBe, GPRS, GSM.

UNIT IV Embedded Firmware Design and Development
Embedded firmware design approaches-super loop based approach, operating system based approach; embedded firmware development languages-assembly language based development, high level language based development.

UNIT V RTOS based Embedded System Design
Operating system basics, types of operating systems, tasks, process and threads, multiprocessing and multitasking, task scheduling: non-pre-emptive and pre-emptive scheduling; task communicationshared memory, message passing, Remote Procedure Call and Sockets, Task Synchronization: Task Communication/ Synchronization Issues, Task Synchronization Techniques

Text books:
1. Introduction to Embedded Systems - Shibu KV, McGraw Hill Education.
2. Computers as Components –Wayne Wolf, Morgan Kaufmann (second edition).

References:
1. Embedded System Design -Frank Vahid, Tony Grivargis, john Wiley.

2. Embedded Systems- An integrated approach - Lyla b das, Pearson education 2012.
3. Embedded Systems – Raj Kamal, TMH

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	SOFTWARE PROJECT MANAGEMENT		L	T	P	C
23BTCS19a	(Professional Elective-III)		3	0	0	3
Year	III	Semester	II			

Course Objective:
This course is designed to enable the students to understand the fundamental principles of SoftwareProject management &will also have a good knowledge of the responsibilities of a project manager and how to handle them.

Course Out comes: After completion of the course, students will be able to

CO1: Describe the fundamentals of Project Management

CO2: Recognize and use Project Scheduling Techniques

CO3: Familiarize with Project Control Mechanisms

CO4: Understand Team Management

CO5: Recognize the importance of Project Documentation and Evaluation

UNIT-I **Lecture9Hrs**

Conventional Software Management: The water fall model, conventional software Management performanceEvolution of Software Economics: software Economics. Pragmatic Software Cost Estimation Improving Software Economics: Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation, Achieving Required Quality, Peer Inspections.

UNIT-II **Lecture9Hrs**

The old way and the new: The principles of convention al software Engineering, principles of modernsoftware management, transitioning to an iterative process.

Lifecycle phases: Engineering and production stages, inception, Elaboration, construction, transitionphases.

Artifacts of the process: The Artifact sets, Management Artifacts, Engineering Artifacts, programmaticArtifacts

UNIT-III **Lecture 9Hrs**

Work Flows of the process: Software process work flows, Inter Trans work flows.**Check points of theProcess:** Major Mile Stones, Minor Milestones, Periodic status assessments. Iterative Process Planning: work break down structures, planning guidelines, cost and schedule estimating, Iteration planning process, Pragmatic planning

UNIT-IV **Lecture9Hrs**

Process Automation: Automation Building Blocks,The Project Environment.

Project Control and Process instrumentation: The sevencore Metrics, Management indicators, qualityindicators

Tailoring the Process: Process discriminants. Managing people and organizing teams.

UNIT-V **Lecture9Hrs**

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

Future Software Project Management: modern Project Profiles, Next generation Software economics,modern process transitions.

Case Study: The Command Center Processing and Display System-Replacement (CCPDS-R)

Text books:

1. Software Project Management, Walker Royce,Pearson Education,2012
2. BobHughes,MikeCotterellandRajibMall—SoftwareProjectManagement‖,6thEdition, McGraw Hill Edition, 2017

Reference Books:

1. PankajJalote,—SoftwareProjectManagementinpractice‖,5thEdition,PearsonEducation, 2017.
2. MuraliK.Chemuturi,Thomas M.CagleyJr.‖Mastering Software Project Management: BestPractices, Tools and Techniques‖, J.Ross Publishing, 2010
3. SanjayMohapatra,—SoftwareProjectManagement‖,CengageLearning,2011

Online Learning Resources:

http://nptel.ac.in/courses/106101061/29

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	MOBILE ADHOC NETWORKS		L	T	P	C
23BTCS19b	(Professional Elective-III)		3	0	0	3
Year	III	Semester	II			

Course Objective:

- Knowledge of mobile ad hoc networks, design and implementation issues, and available solutions.
- Knowledge of routing mechanisms and the three classes of approaches: proactive, on-demand, and hybrid.
- Knowledge of clustering mechanisms and the different schemes that have been employed, e.g., hierarchical, flat, and leaderless.
- Knowledge of the 802.11 Wireless Lan (WiFi) and Bluetooth standards.

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

CO1: Describe the unique issues in ad-hoc/sensor networks.

CO2: Describe current technology trends for the implementation and deployment of wireless adhoc/sensor networks.

CO3: Discuss the challenges in designing MAC, routing and transport protocols for wireless adhoc/sensor networks.

CO4: Discuss the challenges in designing routing and transport protocols for wireless Adhoc/sensornetworks.

CO5: Comprehend the various sensor network Platforms, tools and applications

UNIT- I Introduction to Ad Hoc Networks:

Characteristics of MANETs, Applications of MANETs and challenges of MANETs -Routing in MANETs: Criteria for classification, Taxonomy of MANET routing algorithms, Topology based routing algorithms, Position based routing algorithms, Other routing algorithms.

UNIT -II Data Transmission:

Broadcast storm problem, Broadcasting, Multicasting and Geocasting -TC Pover Ad Hoc: TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

UNIT- III Basics of Wireless, Sensors and Applications:

Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer.

UNIT- IV Data Retrieval in Sensor Networks:

Routing layer, Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs, Sensor Networks and mobile robots-Security: Security in Ad Hoc networks, Key management, Secure routing, Cooperation in MANETs, Intrusion Detection systems.

UNIT- V

Sensor Network Platforms and Tools: Sensor Network Hardware, Berkeley motes, Sensor Network Programming Challenges, Node-Level Software Platforms -Operating System: Tiny OS -Imperative Language: nesC, Data flow style language: Tiny GALS, Node Level Simulators, ns- 2 and its sensor network extension.

TEXT BOOKS:

1. Ad Hoc and Sensor Networks –Theory and Applications, Carlos Corderio Dharma P. Aggarwal,World Scientific Publications, March 2006,ISBN –981-256-681-3

2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN –978-1-55860-914-3 (Morgan Kauffman)

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	NATURAL LANGUAGE PROCESSING		L	T	P	C
23BTCS19c	(Professional Elective-III)		3	0	0	3
Year	III	Semester	II			

Course Objective

- Explain and apply fundamental algorithms and techniques in the area of natural languageprocessing (NLP)
- Discuss approaches to syntax and semantics in NLP.
- Examine current methods for statistical approachesto machine translation.
- Teach machine learning techniques used in NLP.

Course Out comes: After completion of the course, students will be able to

CO1: Understand the various NLP Applications and Organization of Natural language, able to learn and implement realistic applications using Python.

CO2: Apply the various Parsing techniques, Bayes Rule, Shannon game, Entropy and Cross Entropy.

CO3: Understand the fundamentals of CFG and parsers and mechanismsin ATN‘s.

CO4: Apply Semantic Interpretation and Language Modelling.

CO5: Apply the concept of MachineTranslation and multilingual Information Retrieval systemsand Automatic Summarization.

UNIT- I Introduction to Natural language

The Study of Language, Applications of NLP, Evaluating Language Understanding Systems, Different Levels of Language Analysis, Representations and Understanding, Organization of Natural language Understanding Systems, Linguistic Back ground: Anoutline of English Syntax.

UNIT-II Grammars and Parsing

Grammars and Parsing – Top – Down and Bottom-Up Parsers, Transition Network Grammars, Feature Systems and Augmented Grammars, Morphological l Analysis and the Lexicon, Parsing with Features, Augmented Transition Networks, Bayes Rule, Shannon game, Entropy and Cross Entropy.

UNIT-III Grammars for Natural Language

Grammars for Natural Language, Movement Phenomenonin Language, Gap Threading, Human Preferences in Parsing, Shift Reduce Parsers, Deterministic Parsers.

UNIT-IV Semantic Interpretation

Semantic &Logical form, Word senses &ambiguity, The basic logical form language, Encoding ambiguity in the logical Form, Verbs &States in logical form, The maticroles, Speech acts & embedded sentences, Defining semantics structure model theory.

Language Modelling: Introduction,n-GramModels,LanguagemodelEvaluation,ParameterEstimation,LanguageModelAdaption,TypesofLanguageModels,Language-SpecificModellingProblems,MultilingualandCrosslingual Language Modelling.

UNIT-V Machine Translation

Survey:Introduction,ProblemsofMachineTranslation,IsMachineTranslationPossible,BriefHistory,PossibleApproaches,CurrentStatus.

AnusarakaorLanguageAccessor:

Background,CuttingtheGordianKnot,TheProblem,StructureofAnusarakaSystem,UserInterface,LinguisticArea,GivingupAgreementinAnusarsakaOutput,LanguageBridges.

Multilingual Information Retrieval

Introduction, Document Pre-processing, Monolingual Information Retrieval, CLIR, MLIR, Evaluationin Information Retrieval, Tools, Software and Resources.

Multilingual Automatic Summarization

Introduction, Approachesto Summarization, Evaluation, How to Builda Summarizer, Competitions and Datasets.

Textbooks:

1. James Allen, Natural Language Understanding, 2 ndEdition, 2003, Pearson Education.
2. Multilingual Natural Language Processing Applications: From Theory To Practice- DanielM.Bikel and ImedZitouni, Pearson Publications.
3. Natural Language Processing, A paninian perspective, AksharBharathi, Vineetchaitanya, Prentice – Hall of India.

Reference Books:

1. Charniack, Eugene, Statistical Language Learning, MITPress,1993.
2. Jurafsky, Dan and Martin, James, Speech and Language Processing, 2 ndEdition, PrenticeHall,2008.
3. Manning, Christopher and Hen rich, Schutze, Foundations of Statistical Natural Language Processing, MIT Press,1999.

Online Learning Resources:

https://nptel.ac.in/courses/106/105/106105158/
http://www.nptelvideos.in/2012/11/natural-languageprocessing.html

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	DISTRIBUTED OPERATING SYSTEMS		L	T	P	C
23BTIT	(Professional Elective-III)		3	0	0	3
Year	III	Semester	II			

Course Objectives

- To study, learns, and understands the main concepts of advanced operating systems (parallelprocessing systems, distributed systems, real time systems, network operating systems, and open source operating systems)
- Hardware and software features that support these systems.

Course Outcomes

CO1: Understand the design approaches of advanced operating systems
CO2: Analyze the design issues of distributed operating systems.
CO3: Evaluate design issues of multi-processor operating systems.
CO4: Identify the requirements Distributed File System and Distributed Shared Memory.
CO5: Formulate the solutions to schedule the real time applications.

UNIT - I

Architectures of Distributed Systems: System Architecture Types, Distributed Operating Systems, Issues in Distributed Operating Systems, Communication Primitives.
Theoretical Foundations: Inherent Limitations of a Distributed System, Lam port‘s Logical Clocks, Vector Clocks, Causal Ordering of Messages, Termination Detection.

UNIT - II

Distributed Mutual Exclusion: The Classification of Mutual Exclusion Algorithms, Non-Token – Based Algorithms: Lamport‘s Algorithm, The Ricart-Agrawalsa Algorithm, Maekawa‘s Algorithm,TokenBased Algorithms: Suzuki-Kasami‘s Broadcast Algorithm, Singhal‘s Heurisric Algorithm,Raymond‘s Heuristic Algorithm.

UNIT - III

Distributed Deadlock Detection: Preliminaries, Deadlock Handling Strategies in Distributed Systems,Issues in Deadlock Detection and Resolution, Control Organizations for Distributed Deadlock Detection,Centralized- Deadlock – Detection Algorithms, Distributed Deadlock Detection Algorithms,Hierarchical Deadlock Detection Algorithms

UNIT - IV

Multiprocessor System Architectures: Introduction, Motivation for multiprocessor Systems, BasicMultiprocessor System Architectures.
Multi-Processor Operating Systems: Introduction, Structures ofMultiprocessor Operating Systems, Operating Design Issues, Threads, Process Synchronization,Processor Scheduling.
Distributed File Systems: Architecture, Mechanisms for Building Distributed File Systems, Design Issues

UNIT - V

Distributed Scheduling: Issues in Load Distributing, Components of a Load Distributed Algorithm, Stability, Load Distributing Algorithms, Requirements for Load Distributing, Task Migration, Issues in task Migration Distributed Shared Memory: Architecture and Motivation, Algorithms for Implementing DSM, Memory Coherence, Coherence Protocols, Design Issues

TEXT BOOK:

1. Advanced Concepts in Operating Systems, MukeshSinghal, Niranjan G. Shivaratri, Tata McGrawHill Edition 2001

REFERENCE BOOK:

1. Distributed Systems: Andrew S. Tanenbaum, Maarten Van Steen, Pearson Prentice Hall, Edition – 2, 2007

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	CLOUD COMPUTING LAB		L	T	P	C
23BTIT02P			0	0	3	1.5
Year	III	Semester	II			

Course Objectives:
- Demonstrate application development using Cloud
- Explain features of Hadoop

Course Outcomes (CO): On completion of this course, the students will be able to:
CO1: Configure various virtualization tools such as Virtual Box, VMware workstation.
CO2: Design and deploy a web application in a PaaS environment.
CO3: Learn how to simulate a cloud environment to implement new schedulers.
CO4: Install and use a generic cloud environment that can be used as a private cloud.
CO5: Manipulate large data sets in a parallel environment.

List of Experiments:
1. Install Virtual Box/VMware Workstation with different flavours of Linux or windows OS on top of windows operating systems.
2. Install a C compiler in the virtual machine created using virtual box and executes Simple Programs
3. Install Google App Engine. Create hello world app and other simple web applications using python/java.
4. Use GAE launcher to launch the web applications.
5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using try stack (Online Open stack Demo Version)
8. Install Hadoop single node cluster and run simple applications like word count
9. Establish an AWS account. Use the AWS Management Console to launch an EC2 instance and connect to it.
10. Develop a Guestbook Application using Google App Engine
11. Develop a Server less Web App using AWS
12. Design a Content Recommendation system using AWS
13. Design a Cloud based smart traffic management system
14. Design Cloud based attendance management system
15. Design E-learning cloud-based system
16. Using Amazon Lex builds a Chatbot

References:
1. https://www.vmware.com/products/workstation-pro/workstation-pro-evaluation.html.
2. http://code.google.com/appengine/downloads.html
3. http://code.google.com/appengine/downloads.html

Online Learning Resources/Virtual Labs:
1. Google Cloud Computing Foundations Course - Course (nptel.ac.in)

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	MACHINE LEARNING LAB		L	T	P	C
23BTCS10P			0	0	3	1.5
Year	III	Semester	II			

Course Objectives:
- Make use of Data sets in implementing the machine learning algorithms
- Implement the machine learning concepts and algorithms in any suitable language of choice.

Course Outcomes (CO): After completion of the course, students will be able to

CO1: Understand the Mathematical and statistical prospective of machine learning algorithms through python programming

CO2: Appreciate the importance of visualization in the data analytics solution.

CO3: Derive insights using Machine learning algorithms

List of Experiments:

Note: a. The programs can be implemented in either JAVA or Python.
b. For Problems 1 to 6 and 10, programs are to be developed without using the built-in classes or APIs of
Java/Python.
c. Data sets can be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students.

1. Implement and demonstrate the FIND-S algorithm for finding the most specifichypothesis based on a given set of training data samples. Read the training data from a .CSV file.
2. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
3. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
4. Build an Artificial Neural Network by implementing the Back-propagation algorithm and test the same using appropriate data sets.
5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.
6. Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.
7. Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.

Projects

1. Predicting the Sale price of a house using Linear regression
2. Spam classification using Naïve Bayes algorithm
3. Predict car sale prices using Artificial Neural Networks
4. Predict Stock market trends using LSTM
5. Detecting faces from images

References:

1.Python Machine Learning Workbook for beginners, AI Publishing, 2020.

Online Learning Resources/ Virtual Labs:

1) Machine Learning A-Z (Python & R in Data Science Course) | Udemy
2) Machine Learning | Coursera

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	SOFTSKILLS		L	T	P	C
			1	0	2	2
Year	III	Semester	II			

Course Objectives:
- To encourage all round development of the students by focusing on soft skills
- To make the students aware of critical thinking and problem-solving skills
- To enhance healthy relationship and understanding within and outside an organization
- To function effectively with heterogeneous teams

Course Outcomes (CO): After completion of the course, students will be able to
CO1: List out various elements of soft skills
CO2: Describe methods for building professional image
CO3: Apply critical thinking skills in problem solving
CO4: Analyse the needs of an individual and team for well-being
CO5: Assess the situation and take necessary decisions
CO6: Create a productive work place atmosphere using social and work-life skills ensuringpersonal and emotional well-being

UNIT – I Soft Skills & Communication Skills
Soft Skills - Introduction, Need - Mastering Techniques of Soft Skills – Communication Skills - Significance, process, types - Barriers of communication - Improving techniques
Activities:
Intrapersonal Skills- Narration about self- strengths and weaknesses- clarity of thought – self-expression – articulating with felicity
(The facilitator can guide the participants before the activity citing examples from the lives of the great,anecdotes and literary sources)
Interpersonal Skills- Group Discussion – Debate – Team Tasks - Book and film Reviews by groups - Group leader presenting views (non- controversial and secular) on contemporary issues or on a giventopic.
Verbal Communication- Oral Presentations- Extempore- brief addresses and speeches-convincingnegotiating- agreeing and disagreeing with professional grace.
Non-verbal communication – Public speaking – Mock interviews – presentations with an objective toidentify non- verbal clues and remedy the lapses on observation

UNIT – II Critical Thinking
Active Listening – Observation – Curiosity – Introspection – Analytical Thinking – Open-mindedness – Creative Thinking - Positive thinking - Reflection
Activities:
Gathering information and statistics on a topic - sequencing – assorting – reasoning – critiquing issues – placing the problem – finding the root cause - seeking viable solution – judging with rationale – evaluating the views of others - Case Study, Story Analysis.

UNIT – III Problem Solving & Decision Making
Meaning & features of Problem Solving – Managing Conflict – Conflict resolution –Team building - Effective decision making in teams – Methods & Styles
Activities: Placing a problem which involves conflict of interests, choice and views – formulating the problem – exploring solutions by proper reasoning – Discussion on important professional, career and organizational decisions and initiate debate on the appropriateness of the decision.
Case Study & Group Discussion.

UNIT – IV **Emotional Intelligence & Stress Management**

Managing Emotions – Thinking before Reacting – Empathy for Others – Self-awareness – SelfRegulation – Stress factors – Controlling Stress – Tips

Activities:
Providing situations for the participants to express emotions such as happiness, enthusiasm, gratitude, sympathy, and confidence, compassion in the form of written or oral presentations.
Providing opportunities for the participants to narrate certain crisis and stress –ridden situations caused by failure, anger, jealousy, resentment and frustration in the form of written and oral presentation,Organizing Debates.

UNIT – V Corporate Etiquette
Etiquette- Introduction, concept, significance - Corporate etiquette - meaning, modern etiquette,benefits - Global and local culture sensitivity - Gender Sensitivity - Etiquette in interaction- Cell phoneetiquette - Dining etiquette - Netiquette - Job interview etiquette -Corporate grooming tips -Overcoming challenges
Activities
Providing situations to take part in the Role Plays where the students will learn about bad and good manners and etiquette - Group Activities to showcase gender sensitivity, dining etiquette etc. - Conducting mock job interviews - Case Study - Business Etiquette Games
NOTE-:

1. The facilitator can guide the participants before the activity citing examples from the lives of thegreat, anecdotes, epics, scriptures, autobiographies and literary sources which bear true relevance to theprescribed skill.
2. Case studies may be given wherever feasible for example for Decision Making- The decision of KingLear.

Prescribed Books:
1. MitraBarun K, *Personality Development and Soft Skills*, Oxford University Press, Pap/Cdredition 2012
2. Dr ShikhaKapoor, *Personality Development and Soft Skills: Preparing for Tomorrow*, K I2018, esuoHgnihsilbuPlanoitanretnI

Reference Books:
1. Sharma, Prashant, *Soft Skills: Personality Development for Life Success*, BPB Publications2018.
2. Alex K,*Soft Skills*S.Chand& Co, 2012 (Revised edition)
3. Gajendra Singh Chauhan&SangeethaSharma,*Soft Skills: An Integrated Approach toMaximise Personality*Published by Wiley, 2013
4. Pillai, Sabina & Fernandez Agna, *Soft Skills and Employability Skills*, Cambridge UniversityPress, 2018
5. Dr. Rajiv Kumar Jain, Dr. UshaJain,*Life Skills*(Paperback English)Publisher : Vayu Education of India, 2014

Online Learning Resources:
1. https://youtu.be/DUlsNJtg2L8?list=PLLy_2iUCG87CQhELCytvXh0Ey-bOO1_q
2. https://youtu.be/xBaLgJZ0t6A?list=PLzf4HHlsQFwJZel_j2PUy0pwjVUgj7KlJ
3. https://youtu.be/-Y-R9hDl7lU
4. https://youtu.be/gkLsn4ddmTs
5. https://youtu.be/2bf9K2rRWwo
6. https://youtu.be/FchfE3c2jzc
7.https://www.businesstrainingworks.com/training-resource/five-free-businessetiquette-training-games/
8. https://onlinecourses.nptel.ac.in/noc24_hs15/preview
9.https://onlinecourses.nptel.ac.in/noc21_hs76/preview

VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science and Information Technology

Course Code	TECHNICAL REPORT WRITING & IPR		L	T	P	C
			2	0	0	0
Year	III	Semester	II			

Course Objectives:
1. To enable the students to practice the basic skills of research paper writing
2. To make the students understand the importance of IP and to educate them on the basic conceptsof Intellectual Property Rights.
3. To practice the basic skills of performing quality literature review
4. To help them in knowing the significance of real life practice and procedure of Patents.
5. To enable them learn the procedure of obtaining Patents, Copyrights, & Trade Marks

Course Outcomes: At the end of the course, students will be able to
CO1: Identify key secondary literature related to their proposed technical paperwriting
CO2: Explain various principles and styles in technical writing
CO3: Use the acquired knowledge in writing a research/technical paper
CO4: Analyse rights and responsibilities of holder of Patent, Copyright,Trademark, International Trademark etc.
CO5: Evaluate different forms of IPR available at national & international level
CO6: Develop skill of making search of various forms of IPR by using modern tools and techniques.

UNIT – I:
Principles of Technical Writing: styles in technical writing; clarity, precision, coherence andlogicalsequence in writing-avoiding ambiguity- repetition, and vague language –highlighting yourfindingsdiscussing your limitations -hedging and criticizing -plagiarism and paraphrasing.

UNIT – II:
Technical Research Paper Writing: Abstract- Objectives-Limitations-Review of LiteratureProblemsand Framing Research Questions- Synopsis

UNIT – III:
Process of research: publication mechanism: types of journals- indexing-seminars- conferences- proof reading – plagiarism style; seminar & conference paper writing; Methodology-discussion-results-citation rules

UNIT – IV:
Introduction to Intellectual property: Introduction, types of intellectual property, Internationalorganizations, agencies and treaties, importance of intellectual property rights Trade Marks: Purpose and function of trademarks, acquisition of trade mark rights, protectable matter, selecting and
evaluating trade mark, trade mark registration processes.

UNIT – V:
Law of copy rights: Fundamentals of copy right law, originality of material, rights of reproduction, rights to perform the work publicly, copy right ownership issues, copy right registration, notice of copy right, international copy right law.
Law of patents: Foundation of patent law, patent searching process, ownership rights and transfer.
Patent law, intellectual property audits.

Textbooks:
1. Deborah. E. Bouchoux, *Intellectual Property Rights*, Cengage Learning India, 2013
2. Meenakshi Raman, Sangeeta Sharma. *Technical Communication:Principles and practices.*Oxford.
Reference Books:
1. R.Myneni, *Law of Intellectual Property*, 9th Ed, Asia law House, 2019.

2. PrabuddhaGanguli,*Intellectual Property Rights* Tata Mcgraw Hill, 2001
3. P.Naryan,*Intellectual Property Law*, 3rd Ed ,Eastern Law House, 2007.
4. Adrian Wallwork. *English for Writing Research Papers*Second Edition. Springer Cham Heidelberg New York ,2016
5. Dan Jones, Sam Dragga, *Technical Writing Style*

Online Resources

1. https://theconceptwriters.com.pk/principles-of-technical-writing/
2. https://www.ewh.ieee.org/soc/emcs/acstrial/newsletters/summer10/TechPaperWriting.html
3. https://www.ewh.ieee.org/soc/emcs/acstrial/newsletters/summer10/TechPaperWriting.html
4. https://www.manuscriptedit.com/scholar-hangout/process-publishing-research-paper-journal/
5. https://www.icsi.edu/media/website/IntellectualPropertyRightLaws&Practice.pdf
6. https://lawbhoomi.com/intellectual-property-rights-notes/
7. https://www.extension.purdue.edu/extmedia/ec/ec-723.pdf