



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

B. Tech– II Year I Semester

S. No.	Category	Course Code	Title	L	T	P	Credits
1	HS	23BTHS08T	Discrete Mathematics & Graph Theory	3	0	0	3
2	MB	23BTMB01T	Universal Human Values- Understanding Harmony and Ethical Human Conduct	2	1	0	3
3	EC	23BTEC07T	Digital Logic & Computer Organization	3	0	0	3
4	CS	23BTCS03T	Advanced Data Structures & Algorithm Analysis	3	0	0	3
5	CS	23BTCS04T	Object Oriented Programming Through Java	3	0	0	3
6	CS	23BTCS04P	Advanced Data Structures and Algorithm Analysis Lab	0	0	3	1.5
7	CS	23BTCS05P	Object Oriented Programming Through Java Lab	0	0	3	1.5
8	CS	23BTCS01S	Skill Oriented Course– I Python Programming	0	1	2	2
9	CE	23BTCE01A	Mandatory noncredit course - I Environmental Science	2	0	0	-
Total				16	2	8	20

B. Tech– II Year II Semester

S. No.	Category	Course Code	Title	L	T	P	Credits
1	MB	23BTMB02T 23BTMB03T 23BTMB04T	Humanities Elective-I Managerial Economics and Financial Analysis/ Organizational Behaviour/ Business Environment	2	0	0	2
2	HS	23BTHS12T	Probability & Statistics	3	0	0	3
3	CS	23BTCS05T	Operating Systems	3	0	0	3
4	CS	23BTCS06T	Database Management Systems	3	0	0	3
5	CS	23BTCS07T	Software Engineering	2	1	0	3
6	CS	23BTCS06P	Operating Systems Lab	0	0	3	1.5
7	CS	23BTCS07P	Database Management Systems Lab	0	0	3	1.5
8	CS	23BTCS02S	Skill Oriented Course– II Full Stack Development –I	0	1	2	2
9	ME	23BTME09T	Design Thinking & Innovation	1	0	2	2
Total				14	2	10	21

Mandatory Community Service Project Internship of 08 weeks duration during summer vacation

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B.Tech - Computer Science & Engineering****B. Tech – III Year I Semester**

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Machine Learning	3	0	0	3
2	Professional Core	Computer Networks	3	0	0	3
3	Professional Core	Formal Languages and Automata Theory	3	0	0	3
4	Professional Elective-I	1. Object Oriented Analysis and Design 2. Artificial Intelligence 3. Microprocessors & Microcontrollers 4. Data Warehousing & Data Mining 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective-I		3	0	0	3
6	Professional Core	Machine Learning Lab	0	0	3	1.5
7	Professional Core	Computer Networks Lab	0	0	3	1.5
8	Skill Enhancement course	Full Stack Development - II	0	1	2	2
9	Engineering Science	Tinkering Lab	0	0	2	1
10	Evaluation of Community Service Internship		-	-	-	2
Total			15	1	10	23

B. Tech – III Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Compiler Design	3	0	0	3
2	Professional Core	Cloud Computing	3	0	0	3
3	Professional Core	Cryptography & Network Security	3	0	0	3
4	Professional Elective-II	1. Software Testing Methodologies 2. Cyber Security 3. DevOps 4. Embedded Systems 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Professional Elective-III	1. Software Project Management 2. Mobile Adhoc	3	0	0	3



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

		Networks 3. Natural Language Processing 4. Distributed Operating System 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS				
6	Open Elective – II		3	0	0	3
7	Professional Core	Cloud Computing Lab	0	0	3	1.5
8	Professional Core	Cryptography & Network Security Lab	0	0	3	1.5
9	Skill Enhancement Course	Soft skills OR IELTS	0	1	2	2
10	Audit Course	Technical Paper Writing & IPR	2	0	0	-
Total			20	1	08	23
Mandatory Industry Internship of 08 weeks duration during summer vacation						

B. Tech – IV Year I Semester

S.No.	Category	Title	L	T	P	Credits
1	Professional Core	Deep Learning	2	1	0	3
2	Management Course- II	Human Resource Management	2	0	0	2
3	Professional Elective-IV	1. Software Architecture & Design Patterns 2. Blockchain Technology 3. Augmented Reality & Virtual Reality 4. Internet of Things 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
4	Professional Elective-V	1. Agile methodologies 2. Metaverse 3. Computer Vision 4. Cyber Physical Systems 5. 12 week MOOC Swayam/NPTEL course recommended by the BoS	3	0	0	3
5	Open Elective-III		3	0	0	3

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B.Tech - Computer Science & Engineering**

6	Open Elective-IV		3	0	0	3
7	Skill Enhancement Course	Prompt Engineering	0	1	2	2
8	Audit Course	Gender Sensitization	2	0	0	-
9	Internship	Evaluation of Industry Internship	-	-	-	2
Total			18	2	02	21

B. Tech– IV Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Internship & Project Work	Full semester Internship & Project Work	0	0	24	12

Open Electives**Open Electives, offered to other department students:**

- Open Elective I: Java Programming
- Open Elective II: Operating Systems
- Open Elective III: Data Base Management Systems
- Open Elective IV: Computer Networks



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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

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: Apply the 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 Generating functions, 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, Multi graphs 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.

Reference Books:

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Universal Human Values- Understanding Harmony and Ethical Human Conduct		L	T	P	C
23BTMB01T	(Common to All Branches of Engineering)		2	1	0	3
Year	II	Semester	III			

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

Lecture 4: 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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous) B.Tech - Computer Science & Engineering

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

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. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, Amarkantak, 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. Jeevan Vidya: Ek Parichaya, A Nagaraj, Jeevan Vidya Prakashan, 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 – Pandit Sunderlal
9. Rediscovering India - by Dharampal
10. Hind Swaraj 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%202023.pdf>
- <https://fdpsi.aicteindia.org/UHVII%20Class%20Notes%20&%20Handouts/UHV%20Handout%205-Harmony%20in%20the%20Nature%20and%20Existence.pdf>
- <https://fdpsi.aicteindia.org/download/FDPTeachingMaterial/3days%20FDPSI%20UHV%20Teaching%20Material/Day203%20Handouts/UHV%203D%20D3S2A%20Und%20Nature%20Existence.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 & Engineering

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

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Textbooks:

1. Computer Organization, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, 6th edition, McGraw Hill
2. Digital Design, 6th Edition, M. Morris Mano, Pearson Education.

Reference Books:

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

Online Learning Resources:

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Advanced Data Structures & Algorithm Analysis			L	T	P	C
23BTCS03T				3	0	0	3
Year	II	Semester		III			

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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, Sartaj Sahni, Sanguthevar Rajasekaran, 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=PLDN4rr148XKpZkf03iYFI-O29szjTrs_O



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Object Oriented Programming through Java		L	T	P	C
23BTCS04T			3	0	0	3
Year	II	Semester	III			

Course Objectives:

The learning objectives of this course are to:

- identify Java language components and how they work together in applications
- Learn the 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 develop GUI 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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

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, Debasis Samanta, Monalisa Sarma, Cambridge, 2023.
- 3) JAVA for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson.

References Books:

- 1) The complete Reference Java, 11th edition, 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 & Engineering

Course Code	Advanced Data Structures & Algorithm Analysis Lab			L	T	P	C
23BTCS04P				0	0	3	1.5
Year	II	Semester	III				

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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Reference Books:

1. Fundamentals of Data Structures in C++, Horowitz Ellis, Sahni Sartaj, Mehta, Dinesh, 2nd Edition, Universities Press
2. Computer Algorithms/C++ Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, 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 & Engineering

Course Code	Object Oriented Programming through Java Lab			L	T	P	C
23BTCS05P				0	0	3	1.5
Year	II	Semester	III				

Course Objectives:

The aim of this course is to

- Practice object-oriented programming in the Java programming language
- Implement Classes, Objects, Methods, Inheritance, Exception, Runtime Polymorphism, User defined Exception handling mechanism
- Illustrate inheritance, 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:**

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

- Write a JAVA program to search for an element in a given list of elements using binary search mechanism.
- Write a JAVA program to sort for an element in a given list of elements using bubble sort
- Write a JAVA program using String Buffer to delete, remove character.

Exercise - 3

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

Exercise - 4

- Write a JAVA program to implement Single Inheritance
- Write a JAVA program to implement multi-level Inheritance
- Write a JAVA program for abstract class to find areas of different shapes

Exercise - 5

- Write a JAVA program give example for “super” keyword.
- Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?
- Write a JAVA program that implements Runtime polymorphism

Exercise - 6

- Write a JAVA program that describes exception handling mechanism
- Write a JAVA program Illustrating Multiple catch clauses



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

- 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. Sachin Malhotra, Saurabh Chaudhary, "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 & Engineering

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	III			

Course Objectives:

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 mathematical problems

CO2: Develop, run and manipulate python programs using core data structures, files and object-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.

UNIT-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 Operators iv) Logical Operators v) Bit wise Operators vi) Ternary Operator vii) Membership Operators viii) 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:



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

- i. Addition ii. Insertion iii. 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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Reference Books:

1. Gowri shankar S, Veena A., Introduction to Python Programming, CRC Press.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2ndEdition, 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 & Engineering

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

Course Objectives:

- To make the students to get awareness on environment.
- To understand the importance of protecting natural resources, ecosystems for future generations and pollution causes due to the day-to-day activities of human life
- To save earth from the inventions by the engineers.
- 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.

UNIT I

Multidisciplinary Nature of Environmental Studies: – Definition, Scope, and Importance – Need for Public Awareness.

Natural Resources : Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timber extraction – Mining, dams and other effects on forest and tribal people – Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems – Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies – Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. – Energy resources:

UNIT II

Ecosystems: Concept of an ecosystem. – Structure and function of an ecosystem – Producers, consumers, and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids – Introduction, types, characteristic features, structure, and function of the following ecosystem:

- a. Forest ecosystem.
- b. Grassland ecosystem
- c. Desert ecosystem.
- d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Biodiversity and its Conservation: Introduction 0 Definition: genetic, species and ecosystem diversity – Bio-geographical classification of India – Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

local levels – India as a mega-diversity nation – Hot-spots of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts – Endangered and endemic species of India – Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT III

Environmental Pollution: Definition, Cause, effects, and control measures of:

- a. Air Pollution.
- b. Water pollution
- c. Soil pollution
- d. Marine pollution
- e. Noise pollution
- f. Thermal pollution
- g. Nuclear hazards

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, cyclone and landslides.

UNIT IV

Social Issues and the Environment: From Unsustainable to Sustainable development – Urban problems related to energy – Water conservation, rain water harvesting, watershed management – Resettlement and rehabilitation of people; its problems and concerns. Case studies – Environmental ethics: Issues and possible solutions – Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents, and holocaust. Case Studies – Wasteland reclamation. – Consumerism and waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. – Water (Prevention and control of Pollution) Act – Wildlife Protection Act – Forest Conservation Act – Issues involved in enforcement of environmental legislation – Public awareness.

UNIT V

Human Population and the Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – Environment and human health – Human Rights – Value Education – HIV/AIDS – Women and Child Welfare – Role of information Technology in Environment and human health – Case studies.

Field Work: Visit to a local area to document environmental assets River/forest grassland/hill/mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study of common plants, insects, and birds – river, hill slopes, etc.

Textbooks:

1. Textbook of Environmental Studies for Undergraduate Courses Erach Bharucha for University Grants Commission, Universities Press.
2. Palaniswamy, “Environmental Studies”, Pearson education
3. S.Azeem Unnisa, “Environmental Studies” Academic Publishing Company
4. K.Raghavan Nambiar, “Text book of Environmental Studies for Undergraduate Courses as per UGC model syllabus”, Scitech Publications (India), Pvt. Ltd.

References:

1. Deeksha Dave and E.Sai Baba Reddy, “Textbook of Environmental Science”, Cengage Publications.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

2. M.Anji Reddy, “Text book of Environmental Sciences and Technology”, BS Publication.
3. J.P.Sharma, Comprehensive Environmental studies, Laxmi publications.
4. J. Glynn Henry and Gary W. Heinke, “Environmental Sciences and Engineering”, PrenticeHall of India Private limited
5. G.R.Chatwal, “A Text Book of Environmental Studies” Himalaya Publishing House
6. Gilbert M. Masters and Wendell P. Ela, “Introduction to Environmental Engineering and Science, Prentice Hall of India Private limited.

Web Materials:

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 & Engineering

Course Code	Managerial Economics and Financial Analysis			L	T	P	C
23BTMB02T	(Common to All Branches of Engineering)			2	0	0	2
Year	II		Semester	IV			

Course Objectives:

- To inculcate the basic knowledge of microeconomics and financial accounting
- To make the students learn how demand is estimated for different products, input output relationship for optimizing production and cost
- To Know the Various types of market structure and pricing methods and strategy
- To give an overview on investment appraisal methods to promote the students to learn how to plan long-term investment decisions.
- To provide fundamental skills on accounting and to explain the process of preparing financial statements.

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

CO1: Understand the fundamental concepts of Economics viz., Demand, Production, cost, revenue and markets.

CO2: Apply the Concepts of Production and cost analysis for effective Business decision making.

CO3: Understand the different forms of business, markets and pricing methods.

CO4: Analyze the investment opportunities with capital budgeting techniques to maximize returns

CO5: Apply double entry system to prepare accounting statements and evaluate the financial performance of business entity

UNIT – I: Managerial Economics

Introduction – Nature, meaning, significance, functions, and advantages. Demand-Concept, Function, Law of Demand - Demand Elasticity- Types – Measurement. Demand Forecasting Factors governing Forecasting, Methods. Managerial Economics and Financial Accounting and Management.

UNIT – II: Production and Cost Analysis

Introduction – Nature, meaning, significance, functions and advantages. Production Function– Least- cost combination– Short run and long run Production Function- Isoquants and Is costs, Cost & Break-Even Analysis - Cost concepts and Cost behaviour- Break-Even Analysis (BEA) - Determination of Break-Even Point (Simple Problems).

UNIT – III: Business Organizations and Markets

Introduction – Forms of Business Organizations- Sole Proprietary - Partnership - Joint Stock Companies - Public Sector Enterprises. Types of Markets - Perfect and Imperfect Competition - Features of Perfect Competition Monopoly- Monopolistic Competition– Oligopoly-Price-Output Determination - Pricing Methods and Strategies

UNIT – IV: Capital Budgeting

Introduction – Nature, meaning, significance. Types of Working Capital, Components, Sources of Short-term and Long-term Capital, Estimating Working capital requirements.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

Capital Budgeting– Features, Proposals, Methods and Evaluation. Projects – Pay Back Method, Accounting Rate of Return (ARR) Net Present Value (NPV) Internal Rate Return (IRR) Method (sample problems).

UNIT – V: Financial Accounting and Analysis

Introduction – Concepts and Conventions- Double-Entry Bookkeeping, Journal, Ledger, Trial Balance- Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments). Introduction to Financial Analysis - Analysis and Interpretation of Liquidity Ratios, Activity Ratios, and Capital structure Ratios and Profitability.

Textbooks:

- Varshney & Maheswari: Managerial Economics, Sultan Chand.
- Aryasri: Business Economics and Financial Analysis, 4/e, MGH.
- Aryasri: Business Economics and Financial Analysis, 4/e, MGH.
- S.A. Siddiqui and A.S. Siddiqui: Managerial Economics and Financial Analysis, New Age International.
- Joseph G. Nellis and David Parker: Principles of Business Economics, Pearson, 2/e, New Delhi.
- Domnick Salvatore: Managerial Economics in a Global Economy, Cengage.

Online Learning Resources:

- <https://www.slideshare.net/123ps/managerial-economics-ppt>.
- <https://www.slideshare.net/rossanz/production-and-cost-45827016>.
- <https://www.slideshare.net/darkyla/business-organizations-19917607>.
- <https://www.slideshare.net/balarajbl/market-and-classification-of-market>.
- <https://www.slideshare.net/ruchi101/capital-budgeting-ppt-59565396>.
- <https://www.slideshare.net/ashu1983/financial-accounting>.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Organisational Behaviour		L	T	P	C
23BTMB03T	(Common to All Branches of Engineering)		2	0	0	2
Year	II	Semester	IV			

Course Objectives:

- To enable student's comprehension of organizational behavior
- To offer knowledge to students on self-motivation, leadership and management
- To facilitate them to become powerful leaders
- To Impart knowledge about group dynamics
- To make them understand the importance of change and development

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

CO1: Understand basic concepts of organisation and individual behaviour in the organisation

CO2: Apply theories of motivation in the real time business management

CO3: Understand the organizational culture and qualities of leader

CO4: Apply group decision making and team building practices in the real time business management

CO5: Understand Organisational change management practices

UNIT – I: Introduction to Organizational Behavior

Meaning, definition, nature, scope and functions - Organizing Process – Making organizing effective -Understanding Individual Behaviour –Attitude -Perception - Learning – Personality.

UNIT – II: Motivation and Leading

Theories of Motivation- Maslow's Hierarchy of Needs - Herzberg's Two Factor Theory - Vroom's theory of expectancy – Mc Clelland's theory of needs–Mc Gregor's theory X and theory Y– Adam's equity theory.

UNIT – III: Organizational Culture

Introduction – Meaning, scope, definition, Nature - Organizational Climate - Leadership - Traits Theory–Managerial Grid - Transactional Vs Transformational Leadership - Qualities of good Leader - Conflict Management -Evaluating Leader.

UNIT – IV: Group Dynamics

Introduction – Meaning, scope, definition, Nature- Types of groups - Determinants of group behaviour - Group process – Group Development - Group norms - Group cohesiveness - Small Groups - Group decision making - Team building - Conflict in the organization– Conflict resolution

UNIT – V: Organizational Change and Development

Introduction –Nature, Meaning, scope, definition and functions- Organizational Culture - Changing the Culture – Change Management – Work Stress Management – Organizational management – Managerial implications of organization's change and development



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Textbooks:

- Luthans, Fred, Organisational Behaviour, McGraw-Hill, 12 Th edition.
- P Subba Ran, Organisational Behaviour, Himalya Publishing House.
- McShane, Organizational Behaviour, TMH.
- Robbins, P. Stephen, Timothy A. Judge, Organisational Behaviour, Pearson.

Online Learning Resources:

- <https://www.slideshare.net/Knight1040/organizational-culture9608857s>
- www.slideshare.net/AbhayRajpoot3/motivation-165556714
- <https://www.slideshare.net/harshrastogi/group-dynamics-159412405>
- <https://www.slideshare.net/vanyasingla1/organizational-change-development-26565951>



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Business Environment			L	T	P	C
23BTMB04T	(Common to All Branches of Engineering)			2	0	0	2
Year	II		Semester		IV		

Course Objectives:

- To make the student to understand about the business environment.
- To enable them in knowing the importance of fiscal and monetary policy.
- To facilitate them in understanding the export policy of the country
- To Impart knowledge about the functioning and role of WTO
- To Encourage the student in knowing the structure of stock markets

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

CO1: Understand the preliminary concepts of business environment

CO2: Understand fiscal and monetary policy that affect the business environment

CO3: Understand the national and international trade policy of India

CO4: Explain about WTO, EXIM and balance of payment

CO5: Understand money market and capital market, role of SEBI and stock exchanges

UNIT – I: Overview of Business Environment

Introduction – meaning Nature, Scope, significance, functions and advantages. Types Internal & External, Micro and Macro. Competitive structure of industries –Environmental analysis- advantages & limitations of environmental analysis.

UNIT – II: Fiscal & Monetary Policy

Introduction – Nature, meaning, significance, functions and advantages. Public Revenues - Public Expenditure - Evaluation of recent fiscal policy of GOI. Highlights of Budget Monetary Policy - Demand and Supply of Money –RBI -Objectives of monetary and credit policy - Recent trends- Role of Finance Commission.

UNIT – III: India’s Trade Policy

Introduction – Nature, meaning, significance, functions and advantages. Magnitude and direction of Indian International Trade - Bilateral and Multilateral Trade Agreements – EXIM policy and role of EXIM bank -Balance of Payments– Structure & Major components - Causes for Disequilibrium in Balance of Payments - Correction measures.

UNIT – IV: World Trade Organization

Introduction – Nature, significance, functions and advantages. Organization and Structure - Role and functions of WTO in promoting world trade - GATT -Agreements in the Uruguay Round –TRIPS, TRIMS - Disputes Settlement Mechanism - Dumping and Anti-dumping Measures.

UNIT – V: Money Markets and Capital Markets

Introduction – Nature, meaning, significance, functions and advantages. Features and components of Indian financial systems - Objectives, features and structure of money markets and capital markets - Reforms and recent development – SEBI – Stock Exchanges - Investor protection and role of SEBI, Introduction to international finance.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Textbooks:

- Francis Cherunilam, International Business: Text and Cases, Prentice Hall of India.
- K. Aswathappa, Essentials of Business Environment: Texts and Cases & Exercises 13th Revised Edition.HPH.
- K. V. Sivayya, V. B. M Das, Indian Industrial Economy, Sultan Chand Publishers, New Delhi, India.
- Sundaram, Black, International Business Environment Text and Cases, Prentice Hall of India, New Delhi, India.

Online Learning Resources:

- <https://www.slideshare.net/ShompaDhali/business-environment-53111245>
- <https://www.slideshare.net/rbalsells/fiscal-policy-ppt>
- <https://www.slideshare.net/aguness/monetary-policy-presentationppt>
- <https://www.slideshare.net/DaudRizwan/monetary-policy-of-india-69561982>
- <https://www.slideshare.net/ShikhaGupta31/indias-trade-policyppt>



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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

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 elementary tools.

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.

UNIT IV: 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), χ^2 - test for goodness of fit, χ^2 - test for independence of attributes.

Textbooks:

1. Millerand Friends, **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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Reference Books:

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Operating Systems			L	T	P	C
23BTCS05T				3	0	0	3
Year	II	Semester	IV				

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 mechanisms and 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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Text 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

Reference Books:

1. Operating Systems -Internals and Design Principles, Stallings W, 9th Edition, 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>



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Database Management Systems		L	T	P	C
23BTCS06T			3	0	0	3
Year	II	Semester	IV			

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 dependency Lossless join and dependency preserving decomposition, (1NF, 2NF and 3 NF), concept of surrogate key,



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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, 8th edition, 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_01275806667282022456_shared/overview



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Software Engineering		L	T	P	C
23BTCS07T			2	1	0	3
Year	II	Semester	IV			

Course Objectives:

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 & verification procedures.

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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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, McGraw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, 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_0126058950638714_8827_shared/overview
- 3) https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0133826904110039_04735_shared/overview



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

Course Code	Operating Systems Lab		L	T	P	C
23BTCS06P			0	0	3	1.5
Year	II	Semester	IV			

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

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

CO1: Apply fundamental concepts of operating systems, including process and thread management, and mechanisms for inter-process communication.

CO2: Evaluate scheduling algorithms, synchronization techniques, and memory management principles to enhance the performance and efficiency of computer systems.

CO3: Design solutions for deadlock conditions, and evaluate memory management strategies and allocation policies in operating systems.

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

Sample Experiments:

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

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.

Online Learning Resources:

1. <https://www.cse.iitb.ac.in/~mythili/os/>
2. <http://peterindia.net/OperatingSystems.html>



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

Course Code	Database Management Systems Lab		L	T	P	C
23BTCS07P			0	0	3	1.5
Year	II	Semester	IV			

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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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. Ramez Elmasri, 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>



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

Course Code	Full Stack Development – 1(Skill Enhancement Course)	L	T	P	C
23BTCS02S	(Common to CSE, CS & IT and CSE(AI))	0	1	2	2
Year	II	Semester		IV	

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



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B.Tech - Computer Science & Engineering

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>`, `<div>`, `` 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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous) B.Tech - Computer Science & Engineering

8. Java Script Conditional Statements and Loops

- 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”.
- Write a program to display week days using switch case.
- Write a program to print 1 to 10 numbers using for, while and do-while loops.
- Write a program to print data in object using for-in, for-each and for-of loops
- 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., $13 + 53 + 33 = 153$]
- 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

- Design a appropriate function should be called to display
 - Factorial of that number
 - Fibonacci series up to that number
 - Prime numbers up to that number
 - Is it palindrome or not
- 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
 - Factorial of that number
 - Fibonacci series up to that number
 - Prime numbers up to that number
 - Is it palindrome or not
- Write a program to validate the following fields in a registration page
 - Name (start with alphabet and followed by alphanumeric and the length should not be less than 6 characters)
 - Mobile (only numbers and length 10 digits)
 - E-mail (should contain format like xxxxxxx@xxxxxx.xxx)

10. JavaScript Database Connectivity

- Introduction to server-side JavaScript with Node.js
- Connecting JavaScript applications to MySQL and Mongo DB databases

Text Books:

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

Reference Books:

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

Online Learning Resources:

- <https://www.w3schools.com/html>
- <https://www.w3schools.com/css>
- <https://www.w3schools.com/js/>
- <https://www.w3schools.com/nodejs>
- <https://www.w3schools.com/typescript>



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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

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, customer, 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. Idris Mootee, Design Thinking for Strategic Innovation, 2013, JohnWiley&Sons.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)
B.Tech - Computer Science & Engineering

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

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering****III B. Tech I Semester (CSE)**

S. No	Course Code	Title	L	T	P	Credits
1	23BTCS08T	Introduction to Artificial Intelligence	3	0	0	3
2	23BTCS09T	Introduction to Computer Networks	3	0	0	3
3	23BTCS10T	Automata Theory and Compiler Design	3	0	0	3
4	23BTCS11T	Introduction To Quantum Technologies And Applications	3	0	0	3
5	23BTCS12a	Professional Elective-I 1. Object Oriented Analysis and Design 2. Soft Computing 3. Microprocessors & Microcontrollers 4. Data Warehousing & Data Mining	3	0	0	3
	23BTCS12b					
	23BTCS12c					
6		Open Elective-I	3	0	0	3
7	23BTCS08P	Artificial Intelligence Lab	0	0	3	1.5
8	23BTCS09P	Computer Networks Lab	0	0	3	1.5
9	23BTCS03S	Skill Enhancement course Full Stack Development - II	0	1	2	2
10		Tinkering Lab	0	0	2	1
11	23BTCS13	Evaluation of Community Service Internship	-	-	-	2
Total			18	1	10	26

Open Elective – I

S.No.	Course Code	Course Name	Offered by the Dept.
1		Green Buildings	CIVIL
2		Construction Technology and Management	
3		Electrical Safety Practices and Standards	EEE
4		Sustainable Energy Technologies	ME
5		Electronic Circuits	ECE
6		Quantum Technologies And Applications	CSE & Allied
7		Mathematics for Machine Learning and AI	Mathematics
8		Materials Characterization Techniques	Physics
9		Chemistry of Energy Systems	Chemistry
10		English for Competitive Examinations	Humanities
11		Entrepreneurship and New Venture Creation	

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 Engineering

III B. Tech II Semester (CSE)

S.No	Course Code	Title	L	T	P	Credits
1	23BTCS14T	Introduction to Machine Learning	3	0	0	3
2	23BTCS15T	Cloud Computing	3	0	0	3
3	23BTCS16T	Cryptography & Network Security	3	0	0	3
4	23BTCS17a 23BTCS17b 23BTCS17c	Professional Elective-II	3	0	0	3
		1. Software Testing Methodologies				
		2. Cyber Security				
		3. DevOps				
5	23BTCS18a 23BTCS18b 23BTCS18c 23BTCS18d	Professional Elective-III	3	0	0	3
		1. Software Project Management				
		2. Mobile Adhoc Networks				
		3. Natural Language Processing				
6		Open Elective – II	3	0	0	3
7	23BTCS10P	Machine Learning Lab	0	0	3	1.5
8	23BTCS11P	Cryptography & Network Security Lab	0	0	3	1.5
9		Skill Enhancement course Soft skills	0	1	2	2
10		Audit Course Technical Paper Writing & IPR	2	0	0	-
11		Workshop	0	0	0	0
Total			20	1	08	23
Mandatory Industry Internship of 6 to 8 weeks duration during summer vacation						

Open Elective – II

S.No.	Course Code	Course Name	Offered by the Dept.
1		Disaster Management	CIVIL
2		Sustainability In Engineering Practices	
3		Renewable Energy Sources	EEE
4		Automation and Robotics	ME
5		Digital Electronics	ECE
6		Optimization Techniques for Engineers	Mathematics
7		Mathematical Foundation Of Quantum Technologies	
8		Physics Of Electronic Materials And Devices	Physics
9		Chemistry Of Polymers And Applications	Chemistry
10		Academic Writing and Public Speaking	Humanities



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

IV B. Tech I Semester (CSE)

S. No	Course Code	Title	L	T	P	Credits
1	23BTCS19T	Deep Learning	2	1	0	3
2		Management Course- II 1.Business Ethics and Corporate Governance 2.E-Business 3.Management Science	2	0	0	2
3	23BTCS20a 23BTCS20b 23BTCS20c 23BTCS20d	Professional Elective-IV 1. Software Architecture & Design Patterns 2. Blockchain Technology 3. Augmented Reality & Virtual Reality 4. Internet of Things	3	0	0	3
4	23BTCS21a 23BTCS21b 23BTCS21c 23BTCS21d	Professional Elective-V 1. Agile Methodologies 2. Metaverse 3. Computer Vision 4. Cyber Physical Systems	3	0	0	3
5		Open Elective-III	3	0	0	3
6		Open Elective-IV	3	0	0	3
7	23BTCS04S	Skill Enhancement Course Prompt Engineering	0	1	2	2
8		Audit Course Gender Sensitization	2	0	0	-
9	23BTCS22	Evaluation of Industry Internship	-	-	-	2
Total			18	2	02	21

Open Elective – III

S.No	Course Code	Course Name	Offered by the Dept.
1		Building Materials and Services	CIVIL
2		Environmental Impact Assessment	
3		Smart Grid Technologies	EEE
4		3D Printing Technologies	ME
5		Microprocessors and Microcontrollers	ECE
6		Wavelet transforms and its Applications	Mathematics
7		Smart Materials And Devices	Physics
8		Introduction to Quantum Mechanics	
9		Green Chemistry And Catalysis For Sustainable Environment	Chemistry
10		Employability Skills	Humanities



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

Open Elective – IV

S.No	Course Code	Course Name	Offered by the Dept.
1		Geo-Spatial Technologies	CIVIL
2		Solid Waste Management	
3		Electric Vehicles	EEE
4		Total Quality Management	ME
5		Transducers and Sensors	ECE
6		Financial Mathematics	Mathematics
7		Sensors And Actuators For Engineering Applications	Physics
8		Chemistry Of Nano materials and Applications	Chemistry
9		Literary Vibes	Humanities

IV B.Tech II Semester (CSE)

S.No.	Course code	Title	Category	L	T	P	Credits
1	23BTCS23	Internship		-	-	-	4
		Project					8
Total							12

COURSES OFFERED FOR HONOURS DEGREE IN CSE

S.No.	Course Code	Course Title	Contact Hours per week			Credits
			L	T	P	
1	23BTCS01H	Quantum Computing	3	0	0	3
2	23BTCS02H	No SQL Databases	3	0	0	3
3	23BTCS03H	Software Defined Data Centre	3	0	0	3
4	23BTCS04H	Robotics and Intelligent Systems	3	0	0	3
5	23BTCS05H	Cloud Security	3	0	0	3
6	23BTCS06H	No SQL Lab	0	0	3	1.5
7	23BTCS07H	Quantum & Cloud Computing Lab	0	0	3	1.5

LIST OF MINORS OFFERED TO CSE

S.No.	Minor Title	Department offering the Minor
1	Building Planning & Construction Technology	Civil
2	Micro Grid Technology	EEE
3	Energy Systems	
4	3D Printing	ME
5	Industrial Engineering	
6	Embedded Systems and IoT	ECE & VLSI

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	Introduction to Artificial Intelligence			L	T	P	C
23BTCS08T				3	0	0	3
Year	III	Semester		I			

Course Objectives: This course is designed to:

- Introduce the fundamentals, history, and applications of Artificial Intelligence.
- Develop an understanding of intelligent agents and search-based problem-solving.
- Explore knowledge representation, logical reasoning, and classical planning techniques.
- Provide foundational knowledge of reinforcement learning and natural language processing.
- Introduce real-world AI applications in robotics and vision, along with ethical perspectives.

Course Outcomes:

- Describe AI foundations, agent architectures, and environment types.
- Apply search strategies to solve classical and complex problems.
- Represent knowledge using logic and implement basic inference and planning.
- Understand reinforcement learning and natural language applications.
- Explain the principles of robotics and vision, and discuss AI ethics

UNIT-I Introduction**Lecture 9Hrs**

Introduction: What is AI, Foundations of AI, History of AI, The State of Art.

Intelligent Agents: Agents and Environments, Good Behaviour: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

UNIT-II Solving Problems by searching**Lecture 9 Hrs**

Problem Solving Agents, Example problems, Searching for Solutions, Uninformed Search Strategies, Informed search strategies, Heuristic Functions, Beyond Classical Search: Local Search Algorithms and Optimization Problems, Local Search in Continuous Spaces, Searching with Nondeterministic Actions, Searching with partial observations, online search agents and unknown environments.

UNIT-III: Knowledge Representation and Planning**Lecture 8 Hrs**

Knowledge-Based Agents and the Wumpus World, Propositional Logic: Syntax, Semantics, and Theorem Proving, Agents based on Propositional Logic and Model Checking.

First-Order Logic: Syntax, Semantics, and Inference Mechanisms, Unification, Forward and Backward Chaining, and Resolution, Knowledge Engineering and Ontological Representation, Introduction to Classical Planning and Planning Algorithm

UNIT-IV Reinforcement Learning & Natural Language Processing**Lecture 8Hrs**

Reinforcement Learning: Introduction, Passive Reinforcement Learning, Active Reinforcement Learning, Generalization in Reinforcement Learning, Policy Search, applications of RL

Natural Language Processing: Language Models, Text Classification, Information Retrieval, Information Extraction.

UNIT-V Robotics and Computer Vision**Lecture 10Hrs**

Robotics: Introduction, Robot Hardware, Robotic Perception, planning to move, planning uncertain movements, Moving, Robotic software architectures, application domains.

Perception: Computer Vision, Introduction, Image Formation, Simple Image Features, Classifying Images, Detecting Objects, The 3D World, Using Computer Vision.

Philosophical foundations: The Limits of AI, Can Machines Really Think? The Ethics of AI.

Textbooks:

1. Stuart J. Russell, Peter Norvig, —Artificial Intelligence A Modern Approach, 3rd Edition, Pearson Education, 2019.

Reference Books:

1. Nilsson, Nils J., and Nils Johan Nilsson. Artificial intelligence: a new synthesis. Morgan Kaufmann, 1998.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

2. Johnson, Benny G., Fred Phillips, and Linda G. Chase. "An intelligent tutoring system for the accounting cycle: Enhancing textbook homework with artificial intelligence." Journal of Accounting Education 27.1 (2009): 3039.

Online Learning Resources:

<http://peterindia.net/AIlinks.html>

<http://nptel.ac.in/courses/106106139/>

<https://nptel.ac.in/courses/106/105/106105152/>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	INTRODUCTION TO COMPUTER NETWORKS			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

- Identify the software and hardware components of a computer network
- Design software for a computer network
- Develop error, routing, and congestion control algorithms
- Assess critically the existing routing protocols
- Explain the functionality of each layer of a computer network
- Choose the appropriate transport protocol based on the application requirements

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

What Is the Internet? Network Edge, The Network Core, Delay, Loss, and Throughput in Packet Switched Networks (Textbook 2), Reference Models, Multimedia Networks, Guided Transmission 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 Link Protocols, Sliding Window Protocols (Textbook 1)

Introduction to the Link Layer, Error-Detection and -Correction Techniques, Multiple Access Links and Protocols, Switched Local Area Networks, Link Virtualization: A Network as a Link Layer, 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.



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

Reference Books:

1. Forouzan, *Data Communications and Networking*, 5th Edition, McGraw Hill Publication.
2. Youlu Zheng, Shakil Akhtar, *Networks for Computer Scientists and Engineers*, Oxford Publishers, 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 Engineering**

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 the regular languages and equivalent FSMs.
2. Able to construct pushdown automata equivalent to Context free Grammars, construct Turing Machines 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:

1. Demonstrate knowledge on Automata Theory, Regular Expression and Analyze and Design of finite automata, and prove equivalence of various finite automata.
2. Demonstrate knowledge on context free grammar, Analyze and design of PDA and TM.
3. Understand the basic concept of compiler design, and its different phases which will be helpful to construct new tools like LEX, YACC, etc.
4. Ability to implement semantic rules into a parser that performs attribution while parsing and apply error detection and correction methods.
5. Apply the code optimization techniques to improve the space and time complexity of programs while 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, Simplification of CFG's, Chomsky Normal Form, Greibach Normal Form, Push Down Automata (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 of Compiler, 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 address codes.

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.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

3. Compilers: Principles, Techniques, and Tools, Updated 2e July 2023 Alfred V. Aho , Monica S. Lam, Ravi Sethi , Jeffrey D. Ullman , Sorav Bansal

Reference Books:

1. Introduction to Languages and Theory of Computation, John C Martin, 1/e, 2009, Tata McGraw Hill Education, 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 Cengage Learning, 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, Morgan Kaufmann, 2011.

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	INTRODUCTION TO QUANTUM TECHNOLOGIES			L	T	P	C
23BTCS11T	AND APPLICATIONS (Qualitative Treatment)			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):

- Explain core quantum principles in a non-mathematical manner.
- Compare classical and quantum information systems.
- Identify theoretical issues in building quantum computers.
- Discuss quantum communication and computing concepts.
- 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 Quantum mechanics – 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, A snapshot 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 bits vs quantum bits, Quantum systems: trapped ions, superconducting circuits, photons (non engineering 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, Vision vs 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,

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

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*, MIT Press, 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. **Giuliano Benenti, Giulio Casati, Giuliano Strini**, *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 Engineering**

Course Code	OBJECT ORIENTED ANALYSIS AND DESIGN (Professional Elective –I)		L	T	P	C
23BTCS12a			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

1. The importance of modelling in UML.
2. Compare and contrast the object-oriented model with the E-R and EER models.
3. Design use case diagram. Design an application using deployment diagram.
4. Apply UML diagrams to build library application.

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.

Advanced Structural 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 B Jackson 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. Atul Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
4. Mark Priestley: Practical Object-Oriented Design with UML, TMH.
5. Applying UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	SOFT COMPUTING			L	T	P	C
23BTCS12b				3	0	0	3
Year	III	Semester		I			

Course Objectives:

- Familiarize with soft computing concepts
- Introduce and use the idea of fuzzy logic and use of heuristics based on human experience
- Familiarize the Neuro-Fuzzy modelling using Classification and Clustering techniques
- Learn the concepts of Genetic algorithm and its applications
- Acquire the knowledge of Rough Sets.

Course Outcomes:

- Identify the difference between Conventional Artificial Intelligence to Computational Intelligence.
- Understand fuzzy logic and reasoning to handle and solve engineering problems
- Apply the Classification techniques on various applications.
- Perform various operations of genetic algorithms and Rough Sets.

UNIT - I

Introduction to Soft Computing: Evolutionary Computing, "Soft" computing versus "Hard" computing, Soft Computing Methods, Recent Trends in Soft Computing, Characteristics of Soft computing, Applications of Soft Computing Techniques.

UNIT- II

Fuzzy Systems: Fuzzy Sets, Fuzzy Relations, Fuzzy Logic, Fuzzy Rule-Based Systems

UNIT- III

Fuzzy Decision Making, Particle Swarm Optimization.

UNIT- IV

Genetic Algorithms: Basic Concepts, Basic Operators for Genetic Algorithms, Crossover and Mutation Properties, Genetic Algorithm Cycle, Fitness Function, Applications of Genetic Algorithm.

UNIT- V

Rough Sets, Rough Sets, Rule Induction, and Discernibility Matrix, Integration of Soft Computing Techniques.

TEXT BOOK:

1. Soft Computing – Advances and Applications - Jan 2015 by B.K. Tripathy and J. Anuradha – Cengage Learning

REFERENCE BOOKS:

1. S. N. Sivanandam & S. N. Deepa, —Principles of Soft Computing, 2nd edition, Wiley India, 2008.
2. David E. Goldberg, —Genetic Algorithms-In Search, optimization and Machine learning, Pearson Education.
3. J. S. R. Jang, C.T. Sun and E.Mizutani, —Neuro-Fuzzy and Soft Computing, Pearson Education, 2004.
4. G.J. Klir & B. Yuan, —Fuzzy Sets & Fuzzy Logic, PHI, 1995.
5. Melanie Mitchell, —An Introduction to Genetic Algorithm, PHI, 1998.
6. Timothy J. Ross, —Fuzzy Logic with Engineering Applications, McGraw- Hill International editions, 1995.

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	MICROPROCESSORS AND MICROCONTROLLERS		L	T	P	C
			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

1. Gain knowledge on the architecture, operation, and configurations of the 8086 microprocessors.
2. Get familiar with 8086 programming concepts, instruction set, and assembly language development tools.
3. Know the interfacing of 8086 with memory, peripherals, and controllers for various applications.
4. Learn the architecture, instruction set, and programming of the 8051 microcontrollers.
5. Understand microcontroller interfacing techniques, peripheral programming, and processor comparisons.

UNIT-I

8086 Architecture: Main features, pin diagram/description, 8086 microprocessor family, internal architecture, 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, Tata McGraw Hill Education Private Limited, 3rd Edition, 1994.
2. K M Bhurchandi, A K Ray, Advanced Microprocessors and Peripherals, 3rd edition, McGraw Hill Education, 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, 6th edition, 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 Engineering

Course Code	DATA WAREHOUSING & DATA MINING			L	T	P	C
23BTCS12c				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:

- Design a Data warehouse system and perform business analysis with OLAP tools (L6).
- Apply suitable pre-processing and visualization techniques for data analysis (L3)
- Apply frequent pattern and association rule mining techniques for data analysis (L3)
- Design appropriate classification and clustering techniques for data analysis (L6)
- Infer knowledge from raw data (L4)

UNIT- I:**Lecture 9Hrs**

Basic Concepts – Data Warehousing Components – Building a Data Warehouse – Database Architectures 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, Data Visualization, 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 Selection Techniques to improve Classification Accuracy. Clustering Techniques – Cluster analysis-Partitioning Methods – Hierarchical Methods – Density Based Methods – Grid Based Methods – Evaluation of clustering – 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 Micheline Kamber, —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, Shyam Diwakar and V. Ajay, —Insight into Data Mining Theory and Practice, Eastern Economy Edition, Prentice Hall of India, 2006.
3. Ian H. 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 Engineering**

Course Code	ARTIFICIAL INTELLIGENCE LAB			L	T	P	C
23BTCS08P				0	0	3	1.5
Year	III	Semester		I			

Course Objectives:

- To teach the methods of implementing algorithms using artificial intelligence techniques
- To illustrate search algorithms

To demonstrate the building of intelligent agents

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

- Implement search algorithms
- Solve Artificial intelligence problems
- Design chatbot and virtual assistant

List of Experiments:

1. Write a program to implement DFS and BFS
2. Write a Program to find the solution for traveling salesman Problem
3. Write a program to implement Simulated Annealing Algorithm
4. Write a program to find the solution for the wumpus world problem
5. Write a program to implement 8 puzzle problem
6. Write a program to implement Towers of Hanoi problem
7. Write a program to implement A* Algorithm
8. Write a program to implement Hill Climbing Algorithm
9. Build a Chatbot using AWS Lex, Pandora bots.
10. Build a bot that provides all the information related to your college.
11. Build a virtual assistant for Wikipedia using Wolfram Alpha and Python
12. The following is a function that counts the number of times a string occurs in another string: #

Count the number of times string s1 is found in string s2 defcountsubstring(s1,s2):

count = 0

for i in range(0,len(s2)-

len(s1)+1): if s1 ==

s2[i:i+len(s1)]:

count

+= 1

return

count

For instance, count substring ('ab','cabalaba') returns 2.

Write a recursive version of the above function. To get the rest of a string (i.e. everything but the first character).

13. Higher order functions. Write a higher-order function count that counts the number of elements in a list that satisfy a given test. For instance: count (lambda x: x>2, [1, 2, 3, 4, 5]) should return 3, as there are three elements in the list larger than 2. Solve this task without using any existing higher order function.

14. Brute force solution to the Knapsack problem. Write a function that allows you to generate random problem instances for the knapsack program. This function should generate a list of items containing N items that each have a unique name, a random size in the range 1 5 and a random value in the range 1..... 10.

Next, you should perform performance measurements to see how long the given knapsack solver take to solve different problem sizes. You should perform at least 10 runs with different randomly generated problem instances for the problem sizes 10,12,14,16,18,20 and 22. Use a backpack size of 2:5 x N for each value problem size N. Please note that the method used to generate random numbers can also affect performance, since different distributions of values can make the initial conditions of the problem slightly more or less demanding. How much longer time does it take to run this program when we increase the number of items? Does the backpack size affect the answer?

Try running the above tests again with a backpack size of 1 x N and with 4:0 x N.

15. Assume that you are organising a party for N people and have been given a list L of people who, for social reasons, should not sit at the same table. Furthermore, assume that you have C tables (that are

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

infinitely large). Write a function layout (N,C,L) that can give a table placement (i.e. a number from 0 : : :C -1) for each guest such that there will be no social mishaps.

For simplicity we assume that you have a unique number 0N-1 for each guest and that the list of restrictions is of the form [(X, Y) ...] denoting guests X, Y that are not allowed to sit together. Answer with a dictionary mapping each guest into a table assignment, if there are no possible layouts of the guests you should answer False.

References:

1. David Poole, Alan Mackworth, Randy Goebel, Computational Intelligence: a logical approach, Oxford University Press, 2004.
2. G. Luger, —Artificial Intelligence: Structures and Strategies for complex problem solving, Fourth Edition, Pearson Education, 2002.
3. J. Nilsson, —Artificial Intelligence: A new Synthesis, Elsevier Publishers, 1998.
4. Artificial Neural Networks, B. Yagna Narayana, PHI
5. Artificial Intelligence, 2nd Edition, E.Rich and K.Knight, TMH.
6. Artificial Intelligence and Expert Systems, Patterson, PHI.

Online Learning Resources/Virtual Labs:

<https://www.tensorflow.org/><https://pytorch.org/>
<https://github.com/pytorch> <https://keras.io/> <https://github.com/kerasteam>
<http://deeplearning.net/software/theano/>
<https://github.com/Theano/Theano><https://caffe2.ai/>
<https://github.com/caffe2>
<https://deeplearning4j.org/Scikit-learn>:<https://scikit-learn.org/stable/>
<https://github.com/scikit-learn/scikit-learn>
<https://www.deeplearning.ai/>
<https://opencv.org/>
<https://github.com/qqwweee/keras-yolo3>
<https://www.pyimagesearch.com/2018/11/12/yolo-object-detection-with-opencv/>
<https://developer.nvidia.com/cuda-math-library>
http://vlabs.iitb.ac.in/vlabs-dev/labs/machine_learning/labs/index.php

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	COMPUTER NETWORKS 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 topology and observe its performance
- To analyze the traffic flow and the contents of protocol frames.
- Familiarize with the applications of Internet.

Course Outcomes:

- To understand the working principle of various communication protocols.
- To understand the network simulator environment and visualize a network topology and observe its performance.
- To analyze the traffic flow and the contents of protocol frames.
- Critique the existing routing protocols

List of Experiments:

1. Implement the data link layer framing methods such as character, character-stuffing and bit stuffing.
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 window protocol, and loss recovery using the Go-Back-N mechanism.
4. Implement Dijkstra'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.



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

Reference Books:

1. Computer Networks–B. K. Mathan Nagan, T. Maha lakshmi- Charulatha Publications Private Limited-2019.
2. Computer Networks-Dr.Amol V. Dhumane Nitin 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 Engineering**

Course Code	FULL STACK DEVELOPMENT – II		L	T	P	C
23BTCS03S	(Skill Enhancement Course)		0	1	2	2
Year	III	Semester	I			

Course Objectives: The main objectives of the course are to

- Make use of Modern- day JavaScript with ES6 standards for designing Dynamic web pages
- Building robust & responsive User Interfaces using popular JavaScript library **React.js**. Building robust backend APIs using **Express.js**
- Establishing the connection between frontend (React) User interfaces and backend APIs (Express) with Data Bases(My SQL)
- Familiarize students with GitHub for remote repository hosting and collaborative development.

Course Outcomes:

- CO1: Building fast and interactive UIs
- CO2: Applying Declarative approach for developing web apps
- CO3: Understanding ES6 features to embrace modern JavaScript
- CO4: Building reliable APIs with Express. Js
- CO5: Create and manage Git repositories, track changes, and push code to GitHub.

Experiments covering the Topics:

- Introduction to DOM (Document Object Model), Ecma Script (ES6) standards and features like Arrow functions, Spread operator, Rest operator, Type coercion, Type hoisting, String literals, Array and Object Destructuring.
- Basics of React. js like React Components, JSX, Conditional rendering Differences between Real DOM and Virtual DOM.
- Important React.js concepts like React hooks, Props, React forms, Fetch API, Iterative rendering using JavaScript map() function.
- JavaScript runtime environment node. js and its uses, Express. js and Routing, Micro-Services architecture and MVC architecture, database connectivity using (My SQL)
- Introduction to My SQL, setting up MySQL and configuring, Databases, My SQL queries, sub queries, creating My SQL driver for database connectivity to Express. js server.
- Introduction to Git and Git Hub and upload project& team collaboration

Sample Experiments:

1. Introduction to Modern JavaScript and DOM

- Write a JavaScript program to link JavaScript file with the HTML page
- Write a JavaScript program to select the elements in HTML page using selectors
- Write a JavaScript program to implement the event listeners
- Write a JavaScript program to handle the click events for the HTML button elements
- Write a JavaScript program to with three types of functions
 - Function declaration
 - Function definition
 - Arrow functions

2. Basics of React. js

- Write a React program to implement a counter button using react class components
- Write a React program to implement a counter button using react functional components
- Write a React program to handle the button click events in functional component
- Write a React program to conditionally render a component in the browser
- Write a React program to display text using String literals

3. Important concepts of React. js

- Write a React program to implement a counter button using React use State hook
- Write a React program to fetch the data from an API using React use Effect hook
- Write a React program with two react components sharing data using Props.

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

- d. Write a React program to implement the forms in react
- e. Write a React program to implement the iterative rendering using map() function.

4. Introduction to Git and GitHub**a. Setup**

- o Install Git on local machine.
- o Configure Git (user name, email).
- o Create GitHub account and generate a personal access token.

b. Basic Git Workflow

- o Create a local repository using git init
- o Create and add files → git add .
- o Commit files → git commit -m "Initial commit"
- o Connect to GitHub remote → git remote add origin <repo_url>
- o Push to GitHub → git push -u origin main

c. Branching and Collaboration

- o Create a branch → git checkout -b feature1
- o Merge branch to main → git merge feature1
- o Resolve merge conflicts (guided)

5. Upload React Project to GitHub

- o Create a new React app using npx create-react-app myapp
- o Initialize a git repo and push to GitHub
- o Use .gitignore to exclude node_modules
- o Create multiple branches: feature/navbar, feature/form
- o Practice merge and pull requests (can use GitHub GUI)

6. Introduction to Node. js and Express. js

- a. Write a program to implement the `'_hello world'` message in the route through the browser using Express
- b. Write a program to develop a small website with multiple routes using Express. js
- c. Write a program to print the `'_hello world'` in the browser console using Express. js
- d. Write a program to implement the CRUD operations using Express. js
- e. Write a program to establish the connection between API and Database using Express – My SQL driver

7. Introduction to My SQL

- a. Write a program to create a Database and table inside that database using My SQL Command line client
- b. Write a My SQL queries to create table, and insert the data, update the data in the table
- c. Write a My SQL queries to implement the subqueries in the My SQL command line client
- d. Write a My SQL program to create the script files in the My SQL workbench
- e. Write a My SQL program to create a database directory in Project and initialize a database. sql file to integrate the database into API

8. Team Collaboration Using GitHub

- o Form groups of 2–3 students
- o Create a shared GitHub repo
- o Assign tasks and work in branches
- o Use Issues, Pull Requests, and Code Reviews
- o Document code with README.md

Textbooks:

1. Web Design with HTML, CSS, JavaScript and JQuery Set Book by Jon Duckett Professional JavaScript for Web Developers Book by Nicholas C. Zakas



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

2. John Dean, Web Programming with HTML5, CSS and JavaScript, Jones & Bartlett Learning, 2019.
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasana Subramanian, 2nd edition, APress, O'Reilly.
4. Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites by Robin Nixon
5. AZAT MARDAN, Full Stack Java Script: Learn Back bone. js, Node.js and Mongo DB. 2015

Reference Books:

1. Full-Stack JavaScript Development by Eric Bush.
2. Programming the World Wide Web, 7th Edition, Robert W Sebesta, Pearson, 2013.
3. Tomasz Dyl, KamilPrzeorski, MaciejCzarnecki, Mastering Full Stack React Web Development 2017

Online Learning Resources:

- <https://ict.iitk.ac.in/product/full-stack-developer-html5-css3-js-bootstrap-php-4/>
- <https://www.w3schools.com/html>
- <https://www.w3schools.com/css>
- <https://www.w3schools.com/js/>
- <https://www.w3schools.com/nodejs>
- <https://www.w3schools.com/typescript>
- <https://docs.github.com/>
- <https://education.github.com/git-cheat-sheet-education.pdf>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	TINKERING LAB			L	T	P	C
				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

- 1 **Encourage Innovation and Creativity**
- 2 **Provide Hands-on Learning and Impart Skill Development**
- 3 **Foster Collaboration and Teamwork**
- 4 **Enable Interdisciplinary Learning, Prepare for Industry and Entrepreneurship**
- 5 **Impart Problem-Solving mind-set**

These labs bridge the gap between academia and industry, providing students with the practical experience. Some students may also develop entrepreneurial skills, potentially leading to start ups 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

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

- 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 list of 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 Engineering**

Course Code	INTRODUCTION TO MACHINE LEARNING			L	T	P	C
23BTCS14T				3	0	0	3
Year	III	Semester		II			

Course Objectives:

- To introduce the fundamental concepts and types of machine learning.
- To develop a deep understanding of supervised and unsupervised learning algorithms.
- To understand mathematical foundations of learning models and algorithms.
- To evaluate model performance using appropriate statistical and analytical tools.
- To apply machine learning techniques to solve real-world problems using tools such as Scikit learn.

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

- Understand and distinguish among different types of learning methods.
- Apply supervised and unsupervised learning algorithms to datasets.
- Analyze model performance using cross-validation and error metrics.
- Build, test, and improve machine learning models for classification and prediction.
- Use Python-based libraries (e.g., Scikit-learn) to implement ML algorithms.

UNIT I: Introduction to Machine Learning and Linear Models

Definition and Scope of Machine Learning, Applications and Types of Learning: Supervised, Unsupervised, Reinforcement, Linear Regression: Least Squares, Cost Function, Gradient Descent, Polynomial Regression and Over fitting, Evaluation Metrics: RMSE, MAE, R² Score, Bias-Variance Trade off.

UNIT II: Classification Algorithms

Classification Overview and Decision Boundaries, Logistic Regression: Sigmoid Function and Cost, KNearest Neighbors (KNN), Naïve Bayes Classifier, Decision Trees and Random Forests, Model Evaluation: Confusion Matrix, Precision, Recall, F1-Score.

UNIT III: Support Vector Machines and Ensemble Methods

Support Vector Machines: Concepts, Kernels, Hyper plane and Margin Concepts, Kernel Tricks: RBF and Polynomial, Ensemble Learning: Bagging, Boosting, and Voting, Gradient Boosting, AdaBoost, and XGBoost, Model Tuning and Hyper parameter Optimization.

UNIT IV: Unsupervised Learning Techniques

Clustering Overview: Applications, K-Means Clustering Algorithm, Hierarchical Clustering, DBSCAN and Density-Based Methods, Principal Component Analysis (PCA) for Dimensionality Reduction, Silhouette Score, Davies-Bouldin Index for Cluster Validation.

UNIT V: Advanced Topics and Applications

Reinforcement Learning Basics and Markov Decision Processes, Introduction to Neural Networks and Deep Learning, Cross-Validation Techniques: k-Fold, Leave-One-Out, Feature Engineering and Feature Selection, Deployment of ML Models (Flask, Streamlit, etc.), Case Studies: Medical Diagnosis, Spam Detection, Credit Scoring.

Textbooks:

1. Tom Mitchell, **Machine Learning**, McGraw-Hill Education.
2. Aurélien Géron, **Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow**, O'Reilly Media.
3. Ethem Alpaydin, **Introduction to Machine Learning**, MIT Press.

Reference Books:

1. Trevor Hastie, Robert Tibshirani, Jerome Friedman, **The Elements of Statistical Learning**, Springer.
2. Kevin P. Murphy, **Machine Learning: A Probabilistic Perspective**, MIT Press.
3. Christopher Bishop, **Pattern Recognition and Machine Learning**, Springer.



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

Online Learning Resources:

1. Coursera – Machine Learning by Andrew Ng (Stanford University)
2. Scikit-learn Documentation
3. Kaggle Learn – Machine Learning
4. Google’s Machine Learning Crash Course
5. YouTube – StatQuest with Josh Starmer



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

Course Code	CLOUD COMPUTING			L	T	P	C
23BTCS15T				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

- Ability to create cloud computing environment
- Ability to design applications for Cloud environment
- Design & develop back up strategies for cloud data based on features.
- Use and Examine different cloud computing services.
- Apply different cloud programming model as per need.

UNIT I Basics of Cloud computing**Lecture 8Hrs**

Introduction to cloud computing: Introduction, Characteristics 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 Platform forms: Compute Services, Storage Services, Database 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 Cluster set up.

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

Python Basics: Introduction, Installing Python, Python data Types & Data Structures, Control flow, Function, Modules, Packages, File handling, Date/Time Operations, Classes. Structures, Control flow, Function, Modules, Packages, File handling, Date/Time Operations, Classes.

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 Framework, Designing a REST full web API.

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

UNIT IV Big data, multimedia and Tuning**Lecture 8Hrs**

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 Benchmarking and Tuning: Introduction, Work load Characteristics, Application Performance Metrics, Design Considerations for a Benchmarking Methodology, Benchmarking Tools, Deployment Prototyping, Load Testing & Bottleneck Detection case Study, Hadoop benchmarking case Study.

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

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

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 ||By Arshdeep Bahga, Vijay Madiseti, Universities Press, 2016
2. Cloud Computing Principles and Paradigms: By RajKumar Buyya, James Broberg, Andrzej Goscinski, Wiley, 2016

Reference Books:

1. Masterin g Cloud Computing by Rajkumar Buyya, Christian Vecchiola, S Thamarai Selvi, TMH
2. Cloud computing AHands-On Approach by Arshdeep Bahga and Vijay Madiseti.
3. Cloud Computing: A Practical Approach, Anthony T.Velte, To by J.Velte, Robert Elsenpeter, Tata Mc Graw Hill, rp 2011.
4. Enterprise Cloud Computing, Gautam Shroff, 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 Engineering**

Course Code	CRYPTOGRAPHY & NETWORK SECURITY			L	T	P	C
23BTCS16T				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 key cryptography, hashes, and message digests, and public key algorithms
- Design issues and working principles of various authentication protocols, PKI standards
- Various secure communication standards including Kerberos, IPsec, TLS and email
- Concepts of crypto graphic utilities and authentication mechanisms to design secure applications

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

- Identify information security goals, classical encryption techniques and acquire fundamental knowledge on the concepts of in it fields and number theory
- Compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication
- Apply the knowledge of cryptographic check sums and evaluate the performance of different message digest algorithms for verifying the integrity of varying message sizes.
- Demonstrate the ability to apply **user authentication principles** including **Kerberos** for secure authentication
- 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: Symmetric Cipher 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 $GF(2^n)$.

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**

User Authentication: Remote User Authentication Principles, Kerberos.

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

IP Security: IP Security Overview, IP Security Policy, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange.

UNIT V**Lecture8Hrs**

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

Fire walls: Fire wall Character is tics and Access Policy, Types of Fire walls, Fire wall Location and Configurations.



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

Text books:

- 1) Cryptography and Network Security – William Stallings, Pearson Education, 8th Edition.
- 2) Cryptography, Network Security and Cyber Laws–Bernard Menezes, Cengage Learning, 2010 edition.

Reference Books:

- 1) Cryptography and Network Security-Behrouz A Forouzan, Debdeep Mukhopadhyaya, Mc- Graw Hill, 3rd Edition, 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/lecturebyDr.SouravMukhopadhyay> IIT Kharagpur[VideoLecture]
- 3) <https://www.mitel.com/articles/web-communication-cryptography-and-network> security web articles by Mitel Power Connections

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	SOFTWARE TESTING METHODOLOGIES			L	T	P	C
23BTCS17a	(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:

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

UNIT-I**Lecture 9Hrs**

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**Lecture 8Hrs**

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

UNIT-IV**Lecture 9Hrs**

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**Lecture 9Hrs**

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 Engineering**

Course Code	CYBER SECURITY			L	T	P	C
23BTCS17b	Professional Elective-II			3	0	0	3
Year	III	Semester		II			

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

- Classify the cybercrimes and understand the Indian ITA 2000
- Analyse the vulnerabilities in any computing system and find the solutions
- Predict the security threats of the future
- Investigate the protection mechanisms
- 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, Key loggers 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 Engineering**

Course Code	Dev Ops			L	T	P	C
23BTCS17c	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

- Enumerate the principles of continuous development and deployment, automation of configuration management, inter-team collaboration, and IT serviceability
- Describe Dev Ops & Dev Sec Ops methodologies and their key concepts
- Illustrate the types of version control systems, continuous integration tools, continuous monitoring tools, and cloud models
- Set up complete private infrastructure using version control systems and CI/CD tools

UNIT I**Lecture 8 Hrs**

Dev Ops: An Overview, Dev Ops: Origins, Dev Ops: Roots, Dev Ops: Practices Dev Ops: Culture.

Adopting Dev Ops: Developing the Playbook. Developing a Business Case for a Dev Ops: 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. Dev Ops Plays for Optimizing the delivery Pipeline: Dev Ops as an optimization Exercise, Core Themes, The Dev Ops Plays, Specializing Core Plays

UNIT III**Lecture 8Hrs**

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

UNIT IV**Lecture 10 Hrs**

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

UNIT V**Lecture 10 Hrs**

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

Text books:

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

Reference Books:

1. Sanjeev Sharma & Bernie Coyne, Dev Ops for Dummies, Published by John Wiley & Sons, Inc.
2. Michael Huttermann, Dev Ops 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 Engineering**

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 on board 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

Onboard 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 communications shared 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, Mc Graw 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 Engineering**

Course Code	SOFTWARE PROJECT MANAGEMENT			L	T	P	C
23BTCS18a	(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 Software Project 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

- Describe the fundamentals of Project Management
- Recognize and use Project Scheduling Techniques
- Familiarize with Project Control Mechanisms
- Understand Team Management
- Recognize the importance of Project Documentation and Evaluation

UNIT-I**Lecture 9Hrs**

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

UNIT-II**Lecture 9Hrs**

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

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

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

UNIT-III**Lecture 9Hrs**

Work Flows of the process: Software process work flows, Inter Trans work flows. Check points of the Process: 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**Lecture 9Hrs**

Process Automation: Automation Building Blocks, The Project Environment.

Project Control and Process instrumentation: The seven core Metrics, Management indicators, quality indicators

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

UNIT-V**Lecture 9Hrs**

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. Bob Hughes, Mike Cotterell and Rajib Mall—Software Project Management, 6th Edition, Mc Graw Hill Edition, 2017

Reference Books:

1. Pankaj Jalote,—Software Project Management in practice, 5th Edition, Pearson Education, 2017.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

2. Murali K.Chemuturi,Thomas M.Cagley Jr.『Mastering Software Project Management: Best Practices, 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 Engineering**

Course Code	MOBILE ADHOC NETWORKS		L	T	P	C
23BTCS18b	(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:

- Describe the unique issues in ad-hoc/sensor networks.
- Describe current technology trends for the implementation and deployment of wireless adhoc/sensor networks.
- Discuss the challenges in designing MAC, routing and transport protocols for wireless adhoc/sensor networks.
- Discuss the challenges in designing routing and transport protocols for wireless Adhoc/sensor networks.
- 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 -TCP over 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 Engineering**

Course Code	NATURAL LANGUAGE PROCESSING			L	T	P	C
23BTCS18c	(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 language processing (NLP)
- Discuss approaches to syntax and semantics in NLP.
- Examine current methods for statistical approaches to machine translation.
- Teach machine learning techniques used in NLP.

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

- Understand the various NLP Applications and Organization of Natural language, able to learn and implement realistic applications using Python.
- Apply the various Parsing techniques, Bayes Rule, Shannon game, Entropy and Cross Entropy.
- Understand the fundamentals of CFG and parsers and mechanisms in ATN's.
- Apply Semantic Interpretation and Language Modelling.
- Apply the concept of Machine Translation and multilingual Information Retrieval systems and 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: An outline 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 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 Phenomenon in 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 microroles, Speech acts & embedded sentences, Defining semantics structure model theory.

Language Modelling: Introduction, n-Gram Models, Language model Evaluation, Parameter Estimation, Language Model Adaption, Types of Language Models, Language-Specific Modelling Problems, Multilingual and Cross lingual Language Modelling.

UNIT-V Machine Translation

Survey: Introduction, Problems of Machine Translation, Is Machine Translation Possible, Brief History, Possible Approaches, Current Status.

Anusarakaor Language Accessor: Background, Cutting the Gordian Knot, The Problem, Structure of Anusaraka System, User Interface, Linguistic Area, Giving up Agreement in Anusarsaka Output, Language Bridges.

Multilingual Information Retrieval

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

Multilingual Automatic Summarization

Introduction, Approaches to Summarization, Evaluation, How to Build a Summarizer, Competitions and Datasets.



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

Textbooks:

1. James Allen, Natural Language Understanding, 2nd Edition, 2003, Pearson Education.
2. Multilingual Natural Language Processing Applications: From Theory To Practice- Daniel M.Bikel and ImedZitouni, Pearson Publications.
3. Natural Language Processing, A paninian perspective, Akshar Bharathi, 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, 2nd Edition, Prentice Hall,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 Engineering**

Course Code	DISTRIBUTED OPERATING SYSTEMS			L	T	P	C
23BTCS18d	(Professional Elective-III)			3	0	0	3
Year	III		Semester		II		

Course Objectives

- To study, learn, and understand the main concepts of advanced operating systems (parallel processing systems, distributed systems, real time systems, network operating systems, and open source operating systems)
- Hardware and software features that support these systems.

Course Outcomes

- Understand the design approaches of advanced operating systems
- Analyze the design issues of distributed operating systems.
- Evaluate design issues of multiprocessor operating systems.
- Identify the requirements Distributed File System and Distributed Shared Memory.
- 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, Lamport'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-Agrawala Algorithm, Maekawa's Algorithm, TokenBased Algorithms: Suzuki-Kasami's Broadcast Algorithm, Singhal's Heuristic 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, Basic Multiprocessor System Architectures Multi-Processor Operating Systems: Introduction, Structures of Multiprocessor 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, Mukesh Singhal, Niranjana G. Shivaratri, Tata Mc Graw Hill 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 Engineering**

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

- Understand the Mathematical and statistical prospective of machine learning algorithms through python programming
- Appreciate the importance of visualization in the data analytics solution.
- Derive insights using Machine learning algorithms

List of Experiments:

Note: a. The programs can be implemented in either JAVA or Python.

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

- Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.
- 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.
- 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.
- Build an Artificial Neural Network by implementing the Back-propagation algorithm and test the same using appropriate data sets.
- 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.
- 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.
- 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.
- 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.
- 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.
- 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

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

References:

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

Online Learning Resources/ Virtual Labs:

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

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	CRYPTOGRAPHY AND NETWORK SECURITY LAB			L	T	P	C
23BTCS11P				0	0	3	1.5
Year	III	Semester		II			

List of Experiments:

1. Write a C program that contains a string (char pointer) with a value `__Hello world'`. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value `__Hello world'`. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms
 - a. Ceaser cipher
 - b. Substitution cipher
 - c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text `—Hello world|` using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA.

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	DEEP LEARNING		L	T	P	C
23BTCS19T			2	1	0	3
Year	IV	Semester	I			

Course Objectives:

- Demonstrate the major technology trends driving Deep Learning
- Build, train, and apply fully connected deep neural networks
- Implement efficient (vector zed) neural networks
- Analyse the key parameters and hyper parameters in a neural network's architecture

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

- Demonstrate the mathematical foundation of neural network
- Describe the machine learning basics
- Differentiate architecture of deep neural network
- Build a convolution neural network
- Build and train RNN and LSTMs

UNIT-I

Lecture 8Hrs

Linear Algebra: Scalars, Vectors, Matrices and Tensors, Matrix operations, types of matrices, Norms, Eigen decomposition, Singular Value Decomposition, Principal Components Analysis. Probability and Information Theory: Random Variables, Probability Distributions, Marginal Probability, Conditional Probability, Expectation, Variance and Covariance, Bays' Rule, Information Theory. Numerical Computation: Overflow and Underflow, Gradient-Based Optimization, Constrained Optimization, Linear Least Squares.

UNIT- II

Lecture 9Hrs

Machine Learning: Basics and Under fitting, Hyper parameters and Validation Sets, Estimators, Bias and Variance, Maximum Likelihood, Bayesian Statistics, Supervised and Unsupervised Learning, Stochastic Gradient Descent, Challenges Motivating Deep Learning. Deep Feed forward Networks: Learning XOR, Gradient-Based Learning, Hidden Units, Architecture Design, Back-Propagation and other Differentiation Algorithms.

UNIT-III

Lecture 8Hrs

Regularization for Deep Learning: Parameter Norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised Learning, Multi-Task Learning, Early Stopping, Parameter Tying and Parameter Sharing, Sparse Representations, Bagging and Other Ensemble Methods, Dropout, Adversarial Training, Tangent Distance, Tangent Prop and Manifold Tangent Classifier. Optimization for Training Deep Models: Pure Optimization, Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates, Approximate Second-Order Methods, Optimization Strategies and Meta-Algorithms.

UNIT-IV

Lecture 9Hrs

Convolution Networks: The Convolution Operation, Pooling, Convolution, Basic Convolution Functions, Structured Outputs, Data Types, Efficient Convolution Algorithms, Random or Unsupervised Features, Basis for Convolution Networks.

UNIT-V

Lecture 8Hrs

Sequence Modelling: Recurrent and Recursive Nets: Unfolding Computational Graphs, Recurrent Neural Networks, Bidirectional RNNs, Encoder-Decoder Sequence-to-Sequence Architectures, Deep Recurrent Networks, Recursive Neural Networks, Echo State Networks, LSTM, Gated RNNs, Optimization for Long-Term Dependencies, Auto encoders, Deep Generative Models.

Textbooks:

1. Ian Good fellow, Joshua Bagnio, Aaron Carville, —Deep Learning, MIT Press,2016.



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

2. Josh Patterson and Adam Gibson, —Deep learning: A practitioner's approach, O'Reilly Media, First Edition, 2017.

Reference Books:

1. Fundamentals of Deep Learning, Designing next-generation machine intelligence algorithms, Nikhil Baume, O'Reilly, Sheriff Publishers, 2019.
2. Deep learning Cook Book, Practical recipes to get started Quickly, Douse Using, O'Reilly, Sheriff Publishers, 2019

Online Learning Resources:

1. <https://keras.io/datasets/>
2. <http://deeplearning.net/tutorial/deeplearning.pdf>
3. <https://arxiv.org/pdf/1404.7828v4.pdf>
4. <https://www.cse.iitm.ac.in/~miteshk/CS7015.html>
5. <https://www.deeplearningbook.org>
6. <https://nptel.ac.in/courses/106105215>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	SOFTWARE ARCHITECTURE AND DESIGN			L	T	P	C
23BTCS20a	PATTERNS (Professional Elective –IV)			3	0	0	3
Year	IV	Semester		I			

Course Objectives: After completing this course, the student should be able to:

- To understand the concept of patterns and the Catalog.
- To discuss the Presentation tier design patterns and their affect on: sessions, client access, validation and consistency.
- To understand the variety of implemented bad practices related to the Business and Integration tiers.

Course Out comes:

- To highlight the evolution of patterns.
- To learn how to add functionality to designs while minimizing complexity
- To learn what design patterns really are, and are not
- To know about specific design patterns.
- To learn how to use design patterns to keep code quality high without over design.

UNIT-I

Envisioning Architecture: The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

Creating Architecture: Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT- II

Analysing Architectures: Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

Moving from one system to many: Software Product Lines, Building systems from off the shelf components, Software architecture in future.

UNIT- III

Patterns: Pattern Description, Organizing catalogs, role in solving design problems, Selection and usage.

Creational and Structural patterns: Abstract factory, builder, factory method, prototype, singleton, adapter, bridge, composite, façade, flyweight.

UNIT- IV

Behavioural patterns: Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy. template method, visitor.

UNIT- V

Case Studies: A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development.

TEXT BOOKS:

1. Software Architecture in Practice, second edition, Len Bass, Paul Clements & Rick Kazman, Pearson Education, 2003.
2. Design Patterns, Erich Gamma, Pearson Education.

REFERENCES:

1. Beyond Software architecture, Luke Hohmann, Addison wesley, 2003.
2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
3. Software Design, David Budgen, second edition, Pearson education, 2003
4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.
5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education, 2006
6. J2EE Patterns, Deepak Alur, John Crupi & Dan Malks, Pearson education, 2003.
7. Design Patterns in C#, Steven John metsker, Pearson education, 2004.
8. Pattern Oriented Software Architecture, F.Buschmann & others, John Wiley & Sons

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	BLOCK CHAIN TECHNOLOGY		L	T	P	C
23BTCS20b	(Professional Elective –IV)		3	0	0	3
Year	IV	Semester	I			

Course Objectives:

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

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

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

UNIT - I Introduction

Lecture 8Hrs

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

UNIT - II Block chain Concepts

Lecture 9Hrs

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

UNIT - III Architecting Block chain solutions

Lecture 9Hrs

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

UNIT - IV Ethereum Block chain Implementation

Lecture 8Hrs

Ethereum Block chain Implementation: Introduction, Tuna Fish Tracking Use Case, Ethereum Ecosystem, Ethereum Development, Ethereum Tool Stack, Ethereum Virtual Machine, Smart Contract Programming, Integrated Development Environment, Truffle Framework, Ganache, Unit Testing, Ethereum Accounts, My Ether Wallet, Ethereum Networks/Environments, Infura, Ether scan, Ethereum Clients, Decentralized Application, Metamask, Tuna Fish Use Case Implementation, Open Zeppelin in Contracts .

UNIT - V Hyper ledger Block chain Implementation

Lecture 8Hrs

Hyper ledger Implementation: Introduction, Use Case – Car Ownership Tracking, Hyper ledger Fabric, Hyper ledger Fabric Transaction Flow, FabCar Use Case Implementation, Invoking Chain code Functions Using Client Application.

Advanced Concepts in Block chain: Introduction, Inter Planetary File System (IPFS), Zero Knowledge Proofs, Oracles, Self-Sovereign Identity, Block chain with IoT and AI/ML Quantum Computing and Block chain, Initial Coin Offering, Block chain Cloud Offerings, Block chain and its Future Potential.

Textbooks:

1. Ambadas, Arshad Sarfarz Ariff, Sham —Block chain for Enterprise Application Developersl, Wiley, 2020
2. Andreas M. Antonopoulos, —Mastering Bitcoin: Programming the Open Blockchainl, O'Reilly, 2017



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

Reference Books:

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

Online Learning Resources:

<https://github.com/blockchainedindia/resources>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	AUGMENTED REALITY AND VIRTUAL REALITY		L	T	P	C
23BTCS20c	(Professional Elective –IV)		3	0	0	3
Year	IV	Semester	I			

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:

1. Understand the core concepts of Virtual Reality and Augmented Reality, and their differences.
2. Learn about the hardware and software components required for VR and AR systems, as well as the impact of human physiology and perception on the virtual experience.
3. Gain knowledge of input devices (trackers, navigation, and gesture interfaces) and output devices (graphics, sound displays, and haptic feedback).
4. Develop skills in modelling techniques, including geometric, kinematics, physical, and behavior modelling for VR and AR environments.
5. 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, Tele presence, 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.



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

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. Grigore C. Burdea, Philippe Coiffet, *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 Engineering**

Course Code	INTERNET OF THINGS			L	T	P	C
23BTCS20d	(Professional Elective –IV)			3	0	0	3
Year	IV	Semester	I				

Course Objectives:

- Understand the basics of Internet of Things and protocols.
- Discuss the requirement of IoT technology
- Introduce some of the application areas where IoT can be applied.
- Understand the vision of IoT from a global perspective, understand its applications, determine its market perspective using gateways, devices and data management

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

- Understand general concepts of Internet of Things.
- Apply design concept to IoT solutions
- Analyze various M2M and IoT architectures
- Evaluate design issues in IoT applications
- Create IoT solutions using sensors, actuators and Devices

UNIT- I Introduction to IoT

Definition and Characteristics of IoT, physical design of IoT, IoT protocols, IoT communication models, IoT Communication APIs, Communication protocols, Embedded Systems, IoT Levels and Templates

UNIT- II Prototyping IoT Objects using Microprocessor/Microcontroller

Working principles of sensors and actuators, setting up the board – Programming for IoT, Reading from Sensors, Communication: communication through Bluetooth, Wi-Fi.

UNIT-III IoT Architecture and Protocols

Architecture Reference Model- Introduction, Reference Model and architecture, IoT reference Model, Protocols- 6LowPAN, RPL, CoAP, MQTT, IoT frameworks- Thing Speak.

UNIT- IV Device Discovery and Cloud Services for IoT

Device discovery capabilities- Registering a device, Deregister a device, Introduction to Cloud Storage models and communication APIs Web-Server, Web server for IoT.

UNIT- V UAV IoT

Introduction to Unmanned Aerial Vehicles/Drones, Drone Types, Applications: Defense, Civil, Environmental Monitoring; UAV elements and sensors- Arms, motors, Electronic Speed Controller(ESC), GPS, IMU, Ultra sonic sensors; UAV Software –Arudpilot, Mission Planner, Internet of Drones(IoD)- Case study FlytBase.

Text books:

1. Vijay Madiseti and Arshdeep Bahga, — Internet of Things (A Hands-on-Approach)l, 1st Edition, VPT, 2014.
2. Handbook of unmanned aerial vehicles, K Valavanis;George J Vachtsevanos, New York, Springer, Boston, Massachusetts : Credo Reference, 2014. 2016.

Reference Books:

1. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, — From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligencel, 1st Edition, Academic Press, 2014.
2. Arshdeep Bahga, Vijay Madiseti - Internet of Things: A Hands-On Approach, Universities Press, 2014.
3. The Internet of Things, Enabling technologies and use cases – Pethuru Raj, Anupama C. Raman, CRC Press.
4. Francis daCosta, —Rethinking the Internet of Things: A Scalable Approach to Connecting Everythingl, 1st Edition, Apress Publications, 2013



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

5. Cuno P fister, Getting Started with the Internet of Things, O'Reilly Media, 2011, ISBN: 9781- 4493-9357-1
6. DGCA RPAS Guidance Manual, Revision 3 – 2020
7. Building Your Own Drones: A Beginners' Guide to Drones, UAVs, and ROVs, John Baichtal

Online Learning Resources:

1. <https://www.arduino.cc/>
2. <https://www.raspberrypi.org/>
3. <https://nptel.ac.in/courses/106105166/5>
4. <https://nptel.ac.in/courses/108108098/4>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	AGILE METHODOLOGIES		L	T	P	C
23BTCS21a	(Professional Elective –V)		3	0	0	3
Year	IV	Semester	I			

Course Objectives:

- To provide students with a theoretical as well as practical understanding of agile software development practices and how small teams can apply them to create high-quality software.
- To provide good understanding of software design and a set of software technologies and APIs.
- To carry out detailed examination and demonstration of agile development and testing techniques.
- To discuss Agile software development

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

- Realize the importance of interacting with business stakeholders in determining the requirements for a software system
- Perform iterative software development processes: how to plan them, how to execute them.
- Point out the impact of social aspects on software development success.
- Develop techniques and tools for improving team collaboration and software quality.
- Perform Software process improvement as an on-going task for development teams.
- Show how agile approaches can be scaled up to the enterprise level.

UNIT I AGILE METHODOLOGY**Lecture 9 Hrs**

Theories for Agile Management – Agile Software Development – Traditional Model vs. Agile Model - Classification of Agile Methods – Agile Manifesto and Principles – Agile Project Management – Agile Team Interactions – Ethics in Agile Teams - Agility in Design, Testing – Agile Documentations – Agile Drivers, Capabilities and Values

UNIT II AGILE PROCESSES**Lecture 8Hrs**

Lean Production - SCRUM, Crystal, Feature Driven Development- Adaptive Software Development - Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT III AGILITY AND KNOWLEDGE MANAGEMENT**Lecture 8 Hrs**

Agile Information Systems – Agile Decision Making - Earl_S Schools of KM – Institutional Knowledge Evolution Cycle – Development, Acquisition, Refinement, Distribution, Deployment, Leveraging – KM in Software Engineering – Managing Software Knowledge – Challenges of Migrating to Agile Methodologies – Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

UNIT IV AGILITY AND REQUIREMENTS ENGINEERING**Lecture 9 Hrs**

Impact of Agile Processes in RE–Current Agile Practices – Variance – Overview of RE Using Agile Managing Unstable Requirements – Requirements Elicitation – Agile Requirements Abstraction Model – Requirements Management in Agile Environment, Agile Requirements Prioritization – Agile Requirements Modeling and Generation – Concurrency in Agile Requirements Generation.

UNIT V AGILITY AND QUALITY ASSURANCE**Lecture 9 Hrs**

Agile Product Development – Agile Metrics – Feature Driven Development (FDD) – Financial and Production Metrics in FDD – Agile Approach to Quality Assurance - Test Driven Development – Agile Approach in Global Software Development.

Text books:

1. David J. Anderson and Eli Schragenheim, —Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Prentice Hall, 2003.
2. Hazza and Dubinsky, —Agile Software Engineering, Series: Undergraduate Topics in Computer Science, Springer, 2009.

Reference Books:

1. Craig Larman, —Agile and Iterative Development: A Manager's Guide, Addison-Wesley, 2004.
2. Kevin C. Desouza, —Agile Information Systems: Conceptualization, Construction, and Management, Butterworth-Heinemann, 2007.

Online Learning Resources:

<https://www.nptelvideos.com/video.php?id=904>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	METAVERSE		L	T	P	C
23BTCS21b	(Professional Elective –V)		3	0	0	3
Year	IV	Semester	I			

Course Objectives: The main objectives of the course are to:

1. Present and discuss Metaverse characteristics, concepts and layers.
2. Explain and analyse Metaverse technologies, tools, platforms, and applications.
3. Discuss design theories and practices relevant to the Metaverse.
4. Explore cyber security and cybercrime in the Metaverse.
5. Examine open challenges in the Metaverse.

Course Out comes: After completion of the course students are expected to be able to:

1. Understand the characteristics, and interdisciplinary nature of the Metaverse, the opportunities and risks it presents.
2. Analyze Metaverse layers, the technologies used in creating them, as well as design theories and practices for Metaverse.
3. Examine and discuss Metaverse platforms, applications and the latest technological developments in this area.
4. Identify cyber security issues, understand cybercrime, and discuss the open challenges.
5. Building Metaverse Applications

Unit-1

Metaverse fundamentals: Metaverse evolution, Metaverse importance and characteristics, the interdisciplinary nature of the Metaverse, Metaverse opportunities and risks, Computer-mediated communication (social presence theory, social information processing theory, media richness theory, cyborg theory), Avatar-mediated communication.

Unit-2

The seven layers of Metaverse: Experience Discovery, Creator economy, Spatial computing, Decentralization, Human interface, Infrastructure
Metaverse Technologies part I: AR/VR/MR/XR, 3D reconstruction, Game engines, Smart glasses, wearables, haptic devices, headsets and headwear.

Unit-3

Metaverse technologies part II: Blockchain, smart contracts, tokens, NFTs, Cryptography, Artificial Intelligence (AI), Internet of Things (IoT), Edge computing and 5G, 6G. Design theories and practices: Social presence and co-presence, Motion sickness and cyber sickness, Uncanny valley, Sense of selflocation, sense of agency and sense of body ownership, Universal simulation principle, Prototyping, Evaluation techniques (qualitative and quantitative).

Unit-4

Tools and technologies for Metaverse UX and UI: Tools and services for avatar systems, Spatial user interface design, Cross-platform user experience design, Multimodal user interface, Technologies and devices for human computer interaction in Metaverse, Metaverse platforms: Decentra land, SANDBOX, Roblox, Axie Infinity, uHive, Hyper Nation, Nakamoto (NAKA), Metahero (HERO), Star Atlas (ATLAS), Bloktopia (BLOK), Stageverse, Spatial, PalkaCity, Viverse, Sorare, Illuvium, Upland, Second Life, Sansar, Sensorium Galaxy

Unit-5

Metaverse applications - part I: Gaming and entertainment, Travel and tourism, Education and learning, Remote working, Commerce and business, Metaverse applications - part II: Real estate, Banking and Finance, Healthcare, Social media, Fashion, Metaverse and cyber security: Cyber security concerns in Metaverse: Social engineering attacks, Data theft, Decentralization vs vulnerabilities, Cyber security risks in Metaverse: process, people, technology, Metaverse and cybercrime: Scam and theft, Rug pull, Money manipulation and wash trading, Money laundering, Metaverse challenges and open issues: Persistency,

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Interoperability and scalability, Maturity, Regulation, Usefulness and ease-of-use, Privacy and data security, Content creation, NFTs and creator economy, Social, legal and ethical issues in the Metaverse

Text books

1. The Metaverse, Terry Winters, Independently published, 2021, ISBN: 979-8450959283

Reference Books:

1. Ball, M., 2022, —The Metaverse and How It Will Revolutionize Everything—, Liveright, ISBN: 978-1324092032
2. Damar, M. (2021). Metaverse shape of your life for future: A bibliometric snapshot. *Journal of Metaverse*, 1(1), 1–8.
3. Day, J. (2022) Metaverse will see cyber warfare attacks unlike anything before: ‘Massively elevated’, February 28.
<https://www.express.co.uk/news/science/1570844/metaverse-news-cyberwarfare-attacks-virtual-worlds-russia-china-spt>.
4. Polyviou, A., Sharma K., Pappas, I.O.(2023). Training in the metaverse: Employing physiological data to improve how we build metaverses for businesses. *The next generation internet: The role of metaverses, AR, VR, MR, and digital twins*, Temple University Institute for Business and Information Technology
Link: <https://ibit.temple.edu/nextgenerationinternet>
5. QuHarrison T. , Keeney, S., 2022, —The Metaverse Handbook: Innovating for the Internet's NextTectonic Shiftl, Wiley, ISBN: 978-1119892526
6. The mistocleous, M., Christodoulou, K., & Katelaris, L. (2023). An Educational Metaverse Experiment: The first on-chain and in- Metaverse academic course. *Information Systems. EMCIS2022. Lecture Notes in Business Information Processing*, Springer, Cham.
7. Stephenson, N., 1992, —Snow Crashl, ISBN: 978-055338

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	COMPUTER VISION (Professional Elective –V)		L	T	P	C
23BTCS21c			3	0	0	3
Year	IV	Semester	I			

Objectives:

The objective of this course is to understand the basic issues in computer vision and major approaches to address the methods to learn the Linear Filters, segmentation by clustering, Edge detection, Texture.

Course Outcomes: After completing the course, you will be able to:

- Identify basic concepts, terminology, theories, models and methods in the field of computer vision.
- Describe known principles of human visual system,
- Describe basic methods of computer vision related to multi-scale representation, edge detection and detection of other primitives, stereo, motion and object recognition,
- Suggest a design of a computer vision system for a specific problem.

UNIT-I LINEAR FILTERS**Lecture 8Hrs**

Introduction to Computer Vision, Linear Filters and Convolution, Shift Invariant Linear Systems, Spatial Frequency and Fourier Transforms, Sampling and Aliasing Filters as Templates, Technique: Normalized Correlation and Finding Patterns, Technique: Scale and Image Pyramids.

UNIT-II EDGE DETECTION**Lecture 9Hrs**

Noise- Additive Stationary Gaussian Noise, Why Finite Differences Respond to Noise, Estimating Derivatives - Derivative of Gaussian Filters, Why Smoothing Helps, Choosing a Smoothing Filter, Why Smooth with a Gaussian? Detecting Edges-Using the Laplacian to Detect Edges, Gradient-Based Edge Detectors, Technique: Orientation Representations and Corners.

UNIT-III TEXTURE**Lecture 9Hrs**

Representing Texture –Extracting Image Structure with Filter Banks, Representing Texture using the Statistics of Filter Outputs, Analysis (and Synthesis) Using Oriented Pyramids –The Laplacian Pyramid, Filters in the Spatial Frequency Domain, Oriented Pyramids, Application: Synthesizing Textures for Rendering, Homogeneity, Synthesis by Sampling Local Models, Shape from Texture, Shape from Texture for Planes.

UNIT-IV SEGMENTATION BY CLUSTERING**Lecture 8Hrs**

What is Segmentation, Human Vision: Grouping and Gestalt, Applications: Shot Boundary Detection and Background Subtraction. Image Segmentation by Clustering Pixels, Segmentation by Graph Theoretic Clustering. The Hough Transform, Fitting Lines, Fitting Curves

UNIT-V RECOGNITION BY RELATIONS BETWEEN TEMPLATES**Lecture 8Hrs**

Finding Objects by Voting on Relations between Templates, Relational Reasoning Using Probabilistic Models and Search, Using Classifiers to Prune Search, Hidden Markov Models, Application: HMM and Sign Language Understanding, Finding People with HMM.

Text books:

1. David A. Forsyth, Jean Ponce, Computer Vision – A modern Approach, PHI, 2003.

Reference Books:

1. Geometric Computing with Clifford Algebras: Theoretical Foundations and Applications in Computer Vision and Robotics, Springer;1 edition,2001by Sommer.
2. Digital Image Processing and Computer Vision,1/e, by Sonka.
3. Computer Vision and Applications: Concise Edition (With CD) by Jack Academy Press, 2000.

Online Learning Resources:

<https://nptel.ac.in/courses/106105216><https://nptel.ac.in/courses/108103174>

**VEMU INSTITUTE OF TECHNOLOGY (Autonomous)****B. Tech - Computer Science and Engineering**

Course Code	CYBER PHYSICAL SYSTEMS			L	T	P	C
23BTCS21d				3	0	0	3
Year	IV	Semester	I				

Course Objective:

The objective of this course is to provide students with a comprehensive understanding of the various techniques and methodologies used to design, secure, synchronize, and schedule operations within **Cyber- Physical Systems (CPS)**. The course will cover symbolic synthesis for CPS, security aspects, distributed synchronization, real-time scheduling, and model integration, with an emphasis on both basic principles and advanced techniques.

Course Out comes: Upon the Successful Completion of the Course, the Students would be able to:

1. Understand the core principles behind CPS
2. Identify Security mechanisms of Cyber physical systems
3. Understand Synchronization in Distributed Cyber-Physical Systems
4. To Understand the Scheduling for Cyber-Physical Systems
5. To understand the various Cyber-Physical System models

UNIT - I Symbolic Synthesis for Cyber-Physical Systems

Introduction and Motivation, Basic Techniques - Preliminaries, Problem Definition, Solving the Synthesis Problem, Construction of Symbolic Models, Advanced Techniques: Construction of Symbolic Models, Continuous-Time Controllers, Software Tools

UNIT - II Security of Cyber-Physical Systems

Introduction and Motivation, Basic Techniques - Cyber Security Requirements, Attack Model, Countermeasures, Advanced Techniques: System Theoretic Approaches

UNIT - III Synchronization in Distributed Cyber-Physical Systems

Challenges in Cyber-Physical Systems, A Complexity-Reducing Technique for Synchronization, Formal Software Engineering, Distributed Consensus Algorithms, Synchronous Lockstep Executions, Time-Triggered Architecture, Related Technology, Advanced Techniques

UNIT - IV Real-Time Scheduling for Cyber-Physical Systems

Introduction and Motivation, Basic Techniques - Scheduling with Fixed Timing Parameters, Memory Effects, Multiprocessor/Multicore Scheduling, Accommodating Variability and Uncertainty

UNIT - V Model Integration in Cyber-Physical Systems

Introduction and Motivation, Causality, Semantic Domains for Time, Interaction Models for Computational Processes, Semantics of CPS DSMLs, Advanced Techniques, For Spec, The Syntax of CyPhyML, Formalization of Semantics, Formalization of Language Integration.

TEXT BOOKS:

1. Raj Raj kumar, Dion is io De Niz, and Mark Klein, Cyber-Physical Systems, AddisonWesley Professional.
2. Rajeev Alur, Principles of Cyber-Physical Systems, MIT Press, 2015



Course Code	PROMPT ENGINEERING		L	T	P	C
23BTCS04S	Skill Enhancement Course		0	1	2	2
Year	IV	Semester	I			

Course Objective:

This course delves into prompt engineering principles, strategies, and best practices, a crucial aspect in shaping AI models' behaviour and performance. Understanding Prompt Engineering is a comprehensive course designed to equip learners with the knowledge and skills to effectively generate and utilize prompts in natural language processing (NLP) and machine learning (ML) applications. This course delves into prompt engineering principles, strategies, and best practices, a crucial aspect in shaping AI models' behaviour and performance.

Course Out comes:

- Understanding the fundamentals and evolution of prompt engineering.
- Gaining the ability to craft effective closed-ended, open-ended, and role-based prompts.
- Learning to probe and stress-test AI models for bias and robustness.
- Applying prompt optimization techniques and performance evaluation methods.
- Mitigating bias and promoting ethical prompting practices in NLP/ML systems.

Module 1: Introduction to Prompt Engineering

- *Lesson 1: Foundations of Prompt Engineering*
 - Overview of prompt engineering and its significance in NLP and ML.
 - Historical context and evolution of prompt-based approaches.

Module 2: Types of Prompts and Their Applications

- *Lesson 2: Closed-Ended Prompts*
 - Understanding and creating prompts for specific answers.
 - Applications in question- answering systems.
- *Lesson 3: Open-Ended Prompts*
 - Crafting prompts for creative responses.
 - Applications in language generation models.

Module 3: Strategies for Effective Prompting

- *Lesson 4: Probing Prompts*
 - Designing prompts to reveal model biases.
 - Ethical considerations in using probing prompts.
- *Lesson 5: Adversarial Prompts*
 - Creating prompts to stress-test models.
 - Enhancing robustness through adversarial prompting.

Module 4: Fine-Tuning and Optimizing with Prompts

- *Lesson 6: Fine-Tuning Models with Prompts*
 - Techniques for incorporating prompts during model training.
 - Balancing prompt influence and generalization.
- *Lesson 7: Optimizing Prompt Selection*
 - Methods for selecting optimal prompts for specific tasks.
 - Customizing prompts based on model behavior.

Module 5: Evaluation and Bias Mitigation

- *Lesson 8: Evaluating Prompt Performance*
 - Metrics and methodologies for assessing model performance with prompts.
 - Interpreting and analyzing results.
- *Lesson 9: Bias Mitigation in Prompt Engineering*
 - Strategies to identify and address biases introduced by prompts.
 - Ensuring fairness and inclusivity in prompt-based models.



VEMU INSTITUTE OF TECHNOLOGY (Autonomous)

B. Tech - Computer Science and Engineering

Module 6: Real-World Applications and Case Studies

- *Lesson 10: Case Studies in Prompt Engineering*
- *Exploration of successful implementations and challenges in real-world scenarios.*
- *Guest lectures from industry experts sharing their experiences.*

Text books:

1. "Prompt Engineering in Action" – *Danny D. Sullivan*
2. "The Art of Prompt Engineering with Chat GPT: A Hands-On Guide" – *Nathan Hunter.*

Reference Books:

1. "Prompt Engineering in Practice" – *Michael F. Lewis*
2. "Mastering AI Prompt Engineering: The Ultimate Guide for Chat GPT Users" – *Adriano Damiao*
3. "Writing AI Prompts For Dummies" – *Stephanie Diamond and Jeffrey Allan*
4. "Prompt Engineering Guide" (Online Resource) – *promptingguide.ai*

Online Resource link :

<https://www.udemy.com/course/understanding-prompt-engineering/?couponCode=NVDINCTA35TRT>