VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY HYDERABAD
B.TECH. II YEAR
(COMPUTER SCIENCE AND BUSINESS SYSTEMS)

III SEMESTER **R19**

Course Code	Title of the Course	L	T	P/D	Contact Hours/ Week	Credits
19PC1CB01	Formal Language and Automata Theory	2	1	0	3	3
19PC1CB02	Computer Organization and Architecture	3	1	0	4	4
19PC1CB03	Object Oriented Programming	2	1	0	3	3
19PC1CB04	Computational Statistics	2	1	0	3	3
19PC1CB05	Software Engineering	2	1	0	3	3
19PC2CB01	Object Oriented Programming Laboratory	0	0	2	2	1
19PC2CB02	Computational Statistics Laboratory	0	0	2	2	1
19PC2CB03	Software Engineering Laboratory	0	0	2	2	1
Total		**11**	**5**	**6**	**22**	**19**
19MN6HS04	Indian Constitution	**2**	**0**	**0**	**2**	**0**

IV SEMESTER **R19**

Course Code	Title of the Course	L	T	P/D	Contact Hours/ Week	Credits
19PC1CB06	Operating Systems	2	1	0	3	3
19PC1CB07	Database Management System Concepts	2	1	0	3	3
19PC1CB08	Software Design With UML	2	1	0	3	3
19HS1IE01	Introduction to Innovation, IP Management and Entrepreneurship	2	1	0	3	3
19ES1ME10	Operations Research	2	1	0	3	3
19PC2CB04	Operating Systems (Unix)Laboratory	0	0	2	2	1
19PC2CB05	Database Management System Concepts Laboratory	0	0	2	2	1
19PC2CB06	Software Design With UML Laboratory	0	0	2	2	1
19ES2ME10	Operations Research Laboratory	0	0	2	2	1
19HS2EN04	Business Communication and Value Science – III	1	0	2	3	2
Total		**11**	**5**	**10**	**26**	**21**
19MN6HS05	Essence of Indian Traditional Knowledge	2	0	0	2	0

L – Lecture T – Tutorial P – Practical D – Drawing

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB01) FORMAL LANGUAGE AND AUTOMATA THEORY

COURSE OBJECTIVES:

- To discuss the relationships between languages and machines such as FA, PDA, LBA and TM
- To identify a language's location in the Chomsky hierarchy (regular sets, context-free, context-sensitive, and recursively enumerable languages)
- To convert among equivalently powerful notations for a language, including among DFAs, NFAs, and regular expressions, and between PDAs and CFGs
- To build the foundation for students to pursue research in the areas of automata theory, formal languages, compiler design and computational power of machines

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: List computational devices according to their computational power, and tools which will allow us to tell if a device is powerful enough to solve a given computational problem

CO-2: Relate the concept of the grammar with the concept of programming language

CO-3: Design Solutions for problems related to Finite Automata, RE, CFG, PDA and Turing Machine

CO-4: Analyze various problems and categorize them into P, NP, NP- Complete and NP-Hard problems

UNIT – I:

Introduction: Alphabet, languages and Chomsky hierarchy of languages, deterministic finite automata (DFA), nondeterministic finite automata (NFA) and equivalence with DFA, Myhill-Nerode theorem and its uses, minimization of finite automata.

UNIT – II:

Regular languages and finite automata: Regular expressions, deterministic finite automata (DFA) and equivalence with regular expressions, grammars, productions and derivation, and languages, regular grammars and equivalence with finite automata, properties of regular languages, Kleene's theorem, pumping lemma for regular languages.

UNIT – III:

Context-free languages and pushdown automata: Context-free grammars (CFG) and languages (CFL), Chomsky and Greibachnormal forms, deterministic push down automata, nondeterministic pushdown automata (PDA) and equivalence with CFG, pumping lemma for context-free languages, closure properties of CFLs. Context-sensitive languages:Context-sensitive grammars (CSG) and languages, linear bounded automata andequivalence with CSG.

UNIT – IV:

Turing machines: The basic model for Turing machines (TM), Turingrecognizable (recursively enumerable) and Turing-decidable (recursive) languages and theirclosure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMsas enumerators.

UNIT – V:

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

UNIT – VI:

Basic Introduction to Complexity: Introductory ideas on Time complexity of deterministic and nondeterministic Turing machines, P and NP, NP- completeness, Cook's Theorem, other NP -Complete problems.

TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman

REFERENCES:

1. Elements of the Theory of Computation, Harry R. Lewis and Christos H. Papadimitriou
2. Automata and Computability, Dexter C. Kozen
3. Introduction to the Theory of Computation, Michael Sipser
4. Introduction to Languages and the Theory of Computation, John Martin
5. Computers and Intractability: A Guide to the Theory of NP Completeness, M. R. Garey and D. S. Johnson

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	3	**1**	**4**

(19PC1CB02) COMPUTER ORGANIZATION AND ARCHITECHTURE

COURSE OBJECTIVES:

- To describe the functional blocks of a computer to interpret the instructions and various addressing modes for the execution of instruction cycle
- To perform Arithmetic micro operations on integers and Floating point numbers
- To analyze the cost performance and design trade-offs in designing and constructing a computer processor including memory
- To discuss the different ways of communicating with I/O devices & interfaces and the design techniques to enhance the performance using pipelining, parallelism

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Interpret the functional architecture of computing systems
CO-2: Explore memory, control and I/O functions
CO-3: Impart the knowledge on micro programming
CO-4: Analyze instruction level parallelism, Concepts of advanced pipeline techniques

UNIT – I:
Revision of basics in Boolean logic and Combinational/Sequential Circuits
Functional blocks of a computer: CPU, memory, input-output subsystems, control unit.
Instruction set architecture of a CPU: Registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Outlining instruction sets of some common CPUs.

UNIT – II:
Data representation: Signed number representation, fixed and floating point representations, character representation.
Computer arithmetic: Integer addition and subtraction, ripple carry adder, carry look-ahead adder, etc. multiplication – shift-and-add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic, IEEE 754 format.

UNIT – III:
Introduction to x86 architecture.
CPU control unit design: Hardwired and micro-programmed design approaches, design of a simple hypothetical CPU.

UNIT – IV:
Memory system design: Semiconductor memory technologies, memory organization. Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers – program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes – role of interrupts in process state transitions, I/O device interfaces – SCII, USB

UNIT – V:

Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards. Parallel Processors: Introduction to parallel processors, Concurrent access to memory and cache coherency.

UNIT – VI:

Memory organization: Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.

TEXT BOOKS:

1. Computer System Architecture M. M. Mano, 3rd Edition, Prentice Hall of India, New Delhi, 1993
2. Computer Organization and Design: The Hardware/Software Interface, David A. Patterson and John L. Hennessy
3. Computer Organization and Embedded Systems, Carl Hamacher

REFERENCES:

1. Computer Architecture and Organization, John P. Hayes
2. Computer Organization and Architecture: Designing for Performance, William Stallings
3. Computer System Design and Architecture, Vincent P. Heuring and Harry F. Jordan

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB03) OBJECT ORIENTED PROGRAMMING

COURSE OBJECTIVES:

- To understand the basics concepts of OOPs and features of C++
- To understand the creation and memory allocation of objects and concept of friend
- To understand the advanced concepts like inheritance, polymorphism, and exception handling
- To understand the Object-Oriented Analysis and Design tools for developing models

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Apply the object-oriented concepts for solving real time problems
CO-2: Create generic solutions for the given problem
CO-3: Understand how to apply the major object-oriented concepts like encapsulation, inheritance and polymorphism to implement programs in C++
CO-4: Design models by using oops concepts

UNIT – I:
Procedural programming, An Overview of C: Types Operator and Expressions, Scope and Lifetime, Constants, Pointers, Arrays, and References, Control Flow, Functions and Program Structure, Namespaces, error handling, Input and Output (C-way), Library Functions (string, math, stdlib), Command line arguments, Pre-processor directive
Some difference between C and C++: Single line comments, Local variable declaration within function scope, function declaration, function overloading, stronger type checking, Reference variable.

UNIT – II:
Parameter passing – value vs reference, passing pointer by value or reference, #define constant vs const, Operator new and delete, the typecasting operator, Inline Functions in contrast to macro, default arguments
The Fundamentals of Object Oriented Programming: Necessity for OOP, Data Hiding, Data Abstraction, Encapsulation, Procedural Abstraction, Class and Object.

UNIT – III:
More extensions to C in C++ to provide OOP Facilities: Scope of Class and Scope Resolution Operator, Member Function of a Class, private, protected and public Access Specifier, this Keyword, Constructors and Destructors, friend class, error handling (exception)

UNIT – IV:
Essentials of Object Oriented Programming: Operator overloading, Inheritance – Single and Multiple, Class Hierarchy, Pointers to Objects, Assignment of an Object to another Object, Polymorphism through dynamic binding, Virtual Functions, Overloading, overriding and hiding, Error Handling

UNIT – V:

Generic Programming: Template concept, class template, function template, template specialization

Input and Output: Streams, Files, Library functions, formatted output

UNIT – VI:

Object Oriented Design and Modelling: UML concept, Use case for requirement capturing, Class diagram, Activity diagram and Sequence Diagram for design, Corresponding C++ code from design

TEXT BOOKS:

1. The C++ Programming Language, Bjarne Stroustrup, Addison Wesley
2. C++ and Object-Oriented Programming Paradigm, Debasish Jana, PHI Learning Pvt. Ltd

REFERENCES:

1. Programming – Principles and Practice Using C++, Bjarne Stroustrup, Addison Wesley
2. The Design and Evolution of C++, BjarneStroustrup, Addison Wesley

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB04) COMPUTATIONAL STATISTICS

COURSE OBJECTIVES: To learn
- Multidimensional generalization of a univariate normal random variable.
- Concept of multivariable linear regression model
- linear discriminant function analysis
- Data Summarization and data reduction using Factor analysis
- Grouping the data using Cluster analysis

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Model the linear relationship between the explanatory (independent) variables and response (dependent) variables
CO-2: Make better decisions using linear regression techniques
CO-3: Develop discriminant functions that will discriminate between the categories of the dependent variable in a perfect manner
CO-4: Reduce a large number of variables into fewer numbers of factors using factor analysis

UNIT – I:
Multivariate Normal Distribution: Multivariate Normal Distribution Functions, Conditional Distribution and its relation to regression model, Estimation of parameters.

UNIT – II:
Multiple Linear Regression Model: Standard multiple regression models with emphasis on detection of collinearity, outliers, non-normality and autocorrelation, Validation of model assumptions.

UNIT – III:
Multivariate Regression: Assumptions of Multivariate Regression Models, Parameter estimation, Multivariate Analysis of variance and covariance

UNIT – IV:
Discriminant Analysis: Statistical background, linear discriminant function analysis, Estimating linear discriminant functions and their properties.
Principal Component Analysis: Principal components, Algorithm for conducting principal component analysis, deciding on how many principal components to retain, H-plot.

UNIT – V:
Factor Analysis: Factor analysis model, Extracting common factors, determining number of factors, Transformation of factor analysis solutions, Factor scores.

UNIT – VI:
Cluster Analysis: Introduction, Types of clustering, Correlations and distances, clustering by partitioning methods, hierarchical clustering, overlapping clustering, K-Means Clustering-Profiling and Interpreting Clusters.

TEXT BOOKS:

1. An Introduction to Multivariate Statistical Analysis, T.W. Anderson
2. Applied Multivariate Data Analysis, Vol I & II, J.D. Jobson
3. Statistical Tests for Multivariate Analysis, H. Kris

REFERENCES:

1. Regression Diagnostics, Identifying Influential Data and Sources of Collinearety, D.A. Belsey, E. Kuh and R.E. Welsch
2. Applied Linear Regression Models, J. Neter, W. Wasserman and M.H. Kutner
3. Python for Data Analysis, Wes Mc Kinney
4. Applied Linear Regression Models, J. Neter, W. Wasserman and M.H. Kutner
5. Introduction to Linear Regression Analysis, D.C. Montgomery and E.A. Peck

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB05) SOFTWARE ENGINEERING

COURSE OBJECTIVES:

- To identify the software development activities and process models
- To understand the importance of Project planning and management
- To explore various metrics and quality assurance strategies
- To analyze different strategies for testing and risk management

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Analyse software engineering framework activities and process models that can be tailored with appropriate methods for developing the projects

CO-2: Understand the role of formal specifications in project design and be able to develop such specifications

CO-3: To design an interface and develop a prototype for a software system

CO-4: Understand the role of testing in the software development cycle and be capable of developing a test plan to deliver quality software

UNIT – I:

Introduction: Programming in the small vs. programming in the large; software project failures and importance of software quality and timely availability; engineering approach to software development; role of software engineering towards successful execution of large software projects; emergence of software engineering as a discipline.

UNIT – II:

Software Project Management: Basic concepts of life cycle models – different models and milestones; software project planning –identification of activities and resources; concepts of feasibility study; techniques for estimation of schedule and effort; software cost estimation models and concepts of software engineering economics; techniques of software project control and reporting; introduction to measurement of software size; introduction to the concepts of risk and its mitigation; configuration management.

UNIT – III:

Software Quality and Reliability: Internal and external qualities; process and product quality; principles to achieve software quality; introduction to different software quality models like McCall, Boehm, FURPS / FURPS+, Dromey, ISO – 9126; introduction to Capability Maturity Models (CMM and CMMI); introduction to software reliability, reliability models and estimation.

UNIT – IV:

Software Requirements Analysis, Design and Construction: Introduction to Software Requirements Specifications (SRS) and requirement elicitation techniques; techniques for requirement modelling – decision tables, event tables, state transition tables, Petri nets; requirements documentation through use cases; introduction to UML, introduction to software metrics and metrics based control methods; measures of code and design quality.

UNIT – V:

Object Oriented Analysis, Design and Construction: Concepts -- the principles of abstraction, modularity, specification, encapsulation and information hiding; concepts of abstract data type; Class Responsibility Collaborator (CRC) model; quality of design; design measurements; concepts of design patterns; Refactoring; object oriented construction principles; object oriented metrics.

UNIT – VI:

Software Testing: Introduction to faults and failures; basic testing concepts; concepts of verification and validation; black box and white box tests; white box test coverage – code coverage, condition coverage, branch coverage; basic concepts of black-box tests – equivalence classes, boundary value tests, usage of state tables; testing use cases; transaction based testing; testing for non-functional requirements – volume, performance and efficiency; concepts of inspection.

TEXT BOOKS:

1. Software Engineering, Ian Sommerville
2. Fundamentals of Software Engineering, Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino
3. Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices, Michael Jackson

REFERENCES:

1. The Unified Development Process, Ivar Jacobson, Grady Booch, James Rumbaugh
2. Design Patterns: Elements of Object-Oriented Reusable Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
3. Software Metrics: A Rigorous and Practical Approach, Norman E Fenton, Shari Lawrence Pfleeger
4. Software Engineering: Theory and Practice, Shari Lawrence Pfleeger and Joanne M. Atlee
5. Object Oriented Software Engineering: A Use Case Driven Approach --Ivar Jacobson

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19PC2CB01) OBJECT ORIENTED PROGRAMMING LABARATORY

COURSE OBJECTIVES:

- To identify and practice the basic concepts of object-oriented programming
- To familiarize students with advanced concepts of object-oriented programming in C++
- To facilitate students with the skills required to solve problems using object oriented programming
- To design models using object oriented programming

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Understand the concepts of Object Oriented Programming
CO-2: Implement the advanced OOP concepts like inheritance and polymorphism
CO-3: Design the applications by using OOP concepts
CO-4: Represent user requirements using the artifacts of UML (use case, interaction diagrams) and design the activity diagram and state diagram for a given case study

WEEK 1
Moving towards C to C++ -- Sample programs using C++, Sample programs using classes

WEEK 2
Programs on Parameter passing methods, Inline functions, Static members, Access specifiers

WEEK 3
Programs on default arguments, constructors, Constructor overloading, destructors, "this" pointers

WEEK 4
Programs on Dynamic memory allocation, friend functions and classes

WEEK 5
Programs on function overloading, Operator Overloading

WEEK 6
Programs on function and class templates

WEEK 7
Programs on Inheritance- Different forms of inheritance 17

WEEK 8
Programs using abstract classes, polymorphism

WEEK 9
Sample Programs on Exception handling – Basic constructs Programs on multiple catch statement, Exceptions in Constructors and destructors

WEEK 10
Programs on different operations on files

WEEK 11
Programs on random access to files

WEEK 12
Design a Use case Diagram for ATM system, College Admission

WEEK 13
Design a class and activity diagrams for ATM and College admission

WEEK 14
Design a class and sequence diagrams for ATM and College admission

TEXT BOOKS:

1. The C++ Programming Language, Bjarne Stroustrup, Addison Wesley
2. C++ and Object-Oriented Programming Paradigm, Debasish Jana, PHI Learning Pvt. Ltd

REFERENCES:

1. Programming – Principles and Practice Using C++, Bjarne Stroustrup, Addison Wesley
2. The Design and Evolution of C++, Bjarne Stroustrup, Addison Wesley

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19PC2CB02) COMPUTATIONAL STATISTICS LABORATORY

COURSE OBJECTIVES:
- To learn object oriented concepts in python
- To reading and writing data from different sources
- To learn the available libraries in python
- To relate statistical methods to real time applications

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Apply predefined libraries for various applications
CO-2: Analyze the results by applying various statistical methods
CO-3: Plot graphs for the given data sets
CO-4: Summarize the data by applying statistical analysis

WEEK 1,2,3,4:
Python Concepts, Data Structures, Classes: Interpreter, Program Execution, Statements, Expressions, Flow Controls, Functions, Numeric Types, Sequences and Class Definition, Constructors, Text & Binary Files - Reading and Writing

WEEK 5,6,7,8,9:
Visualization in Python: Matplotlib package, Plotting Graphs, Controlling Graph, Adding Text, More Graph Types, Getting and setting values, Patches

WEEK 10,11,12:
Multivariate data analysis: Multiple regression, multi variate regression, cluster analysis with various algorithms, factor analysis, PCA and linear discriminant analysis. Various datasets should be used for each topic

TEXT BOOKS:

1. An Introduction to Multivariate Statistical Analysis, T.W. Anderson
2. Applied Multivariate Data Analysis, Vol I & II, J.D. Jobson
3. Statistical Tests for Multivariate Analysis, H. Kris

REFERENCES:

1. Regression Diagnostics , Identifying Influential Data and Sources of Collinearety, D.A. Belsey, E. Kuh and R.E. Welsch
2. Introduction to Linear Regression Analysis, D.C. Montgomery and E.A. Peck
3. Programming Python, Mark Lutz
4. Python 3 for Absolute Beginners, Tim Hall and J-P Stacey
5. Python for Data Analysis, Wes Mc Kinney

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19PC2CB03) SOFTWARE ENGINEERING LABORATORY

COURSE OBJECTIVES:

- To understand the role of formal specifications in project design and be able to develop such specifications
- To be able to design an interface and develop a prototype for a complex software system;
- To understand the role of testing in the software development cycle and be capable of developing a test plan
- To be aware of and able to use Computer Aided Software Engineering (CASE) tool

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Develop requirement specifications for a software problem in hand
CO-2: Perform functional oriented and object oriented design
CO-3: Implement the concepts of object oriented to develop a real world application
CO-4: Prepare test cases to rigorously test the application for ensuring quality

WEEK 1& 2:
Phases in software development project, overview and need. Understand problems in existing systems and perform system analysis: Requirement analysis, SRS

WEEK 3:
To perform the function oriented design: Data flow diagrams and Structured chart

Week 4:
To perform the user's view analysis: Use case diagram

Week 5:
To draw the structural view diagram: Class diagram, object diagram.

Week 6:
To draw the behavioral view diagram: Sequence diagram, Collaboration diagram

Week 7:
To draw the behavioral view diagram: State-chart diagram, Activity diagram

Week 8:
To draw the implementation view diagram: Component diagram.

Week 9:
To draw the implementation view diagram: deployment diagram

Week 10:
To perform various techniques for testing using the testing tool: unit testing, integration.

Week 11&12:
Use a Software Configuration Management tool for tracking and controlling changes in the software.

TEXT BOOKS:
1. Software Engineering, Ian Sommerville
2. Fundamentals of Software Engineering, Carlo Ghezzi, Jazayeri Mehdi, Mandrioli Dino
3. Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices, Michael Jackson

REFERENCES:

1. The Unified Development Process, Ivar Jacobson, Grady Booch, James Rumbaugh
2. Design Patterns: Elements of Object-Oriented Reusable Software, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides
3. Software Metrics: A Rigorous and Practical Approach, Norman E Fenton, Shari Lawrence Pfleeger
4. Software Engineering: Theory and Practice, Shari Lawrence Pfleeger and Joanne M. Atlee
5. Object Oriented Software Engineering: A Use Case Driven Approach --Ivar Jacobson

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. III Semester	**L**	**T/P/D**	**C**
	2	**0**	**0**

(19MN6HS04) INDIAN CONSTITUTION

COURSE PRE-REQUISITES: Basic knowledge of Panchayat Raj and human rights studied at schooling level.

COURSE OBJECTIVE:

- To develop constitutional awareness
- To understand democracy at grass root level
- To familiarization of human rights and duties among students
- To inculcate responsibilities towards nation building through technology

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Understand constitutional creation its process and the process of Panchayat Raj and its working

CO-2: Get familiar with the science of Nation building through constitutional process of India

CO-3: Understand human rights, responsibilities and recognize the responsibilities for societal well-being

CO-4: Recognize the roles of constitution for corporate culture building

UNIT – I:

(Union Government and its Administration-Part 1)

Constitution' meaning of the term, Indian Constitution: Sources and constitutional history, Features: Citizenship, Preamble, Fundamental Rights and Duties, Directive Principles of State Policy,

UNIT – II:

(Union Government and its Administration-Part 2)

Structure of the Indian Union: Federalism, Centre- State relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok Sabha, Rajya Sabha.

UNIT – III:

(State Government and its Administration-Part 1)

Governor: Role and Position, CM and Council of ministers, State Secretariat: Organization, Structure and Functions.

UNIT – IV:

(State Government and its Administration-Part 2)

Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation, Pachayati raj: Introduction, PRI: Zila Pachayat, Elected officials and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational

Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy.

UNIT – V:
(Duties and Fundamental Rights-Part 1)
Features of fundamental rights, laws inconsistent with fundamental rights, right to equality, right to freedom, right against exploitation, right to freedom of religion, cultural and educational rights, right to constitutional remedies, criticism of fundamental rights, significance of fundamental rights

UNIT – VI:
(Duties and Fundamental Rights-Part 1)
Swaran Singh's Committees recommendation, list of fundamental duties, features of fundamental duties, Protection of Human Rights Act, 1993.

REFERENCE BOOKS:

1. Indian Polity' by Laxmikanth Pub Macgrow Hill
2. Indian Constitution by M.V. Pylee
3. Human Rights in Constitutional Law by Durgadas Basu
4. Indian Constitution Upkar Publication

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB06) OPERATING SYSTEMS

COURSE OBJECTIVES:

- To analyze the tradeoffs inherent in operating system design
- To summarize various approaches to solve the problem of process concurrency in an operating system
- To evaluate the memory usage trade-offs in terms of size (main memory, auxiliary memory) and processor speed
- To understand disk storage strategies and file strategies with protection and security issues

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Identify System calls and evaluate process scheduling criteria of OS
CO-2: Develop procedures for process synchronization and scheduling services of an OS
CO-3: Distinguish disk access, file systems supported by an OS
CO-4: Extend operating systems virtual memory, protection and security aspects

UNIT – I:
Introduction: Concept of Operating Systems (OS), Generations of OS, Types of OS, OS Services, Interrupt handling and System Calls, Basic architectural concepts of an OS, Concept of Virtual Machine, Resource Manager view, process view and hierarchical view of an OS.
Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching.
Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads.

UNIT – II:
Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.
Scheduling algorithms: Pre-emptive and non-pre-emptive, FCFS, SJF, RR; Multiprocessor scheduling: Real Time scheduling: RM and EDF.

UNIT – III:
Inter-process Communication: Concurrent processes, precedence graphs, Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Semaphores, Strict Alternation, Peterson's Solution, The Producer / Consumer Problem, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem, Barber's shop problem.

UNIT – IV:
Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

Concurrent Programming: Critical region, conditional critical region, monitors, concurrent languages, communicating sequential process (CSP); Deadlocks - prevention, avoidance, detection and recovery.

UNIT – V:

Memory Management: Basic concept, Logical and Physical address maps, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page allocation, Partitioning, Paging, Page fault, Working Set, Segmentation, Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

UNIT – VI:

I/O Hardware: I/O devices, Device controllers, Direct Memory Access, Principles of I/O.

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed),Free-space management (bit vector, linked list, grouping), directory implementation(linear list, hash table), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.

Case study: UNIX OS file system, shell, filters, shell programming, programming with the standard I/O, UNIX system calls.

TEXT BOOKS:

1. Operating System Concepts Essentials. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne

REFERENCES:

1. Operating Systems: Internals and Design Principles. William Stallings
2. Operating System: A Design-oriented Approach. Charles Patrick Crowley
3. Operating Systems: A Modern Perspective. Gary J. Nutt
4. Design of the Unix Operating Systems. Maurice J. Bach
5. Understanding the Linux Kernel, Daniel Pierre Bovet, Marco Cesati

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB07) DATATBASE MANGEMENT SYSTEM CONCEPTS

COURSE OBJECTIVES:

- To introduction of Data Base Management systems concepts and to give the description of structure of Data Base systems
- To know the features of various models of data and query representations
- To prepare the database through normalization, Query optimization and understand the concepts of data storage
- To introduce the concepts and protocols related to transaction management, database recovery and security

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Appreciate the underlying concepts of database system architecture and technologies
CO-2: Query the database using the relevant programming language
CO-3: Develop database schema for a given scenario and implement optimization techniques
CO-4: Design schedules using multiple transactions, implement data base recovery and security

UNIT – I:
Introduction: Introduction to Database. Hierarchical, Network and Relational Models. Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML).
Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

UNIT – II:
Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server.

UNIT – III:
Relational database design: Domain and data dependency, Armstrong's axioms, Functional Dependencies, Normal forms, Dependency preservation, Lossless design.

UNIT – IV:
Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.
Storage strategies: Indices, B-trees, Hashing.

UNIT – V:
Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.

UNIT – VI:

Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.

Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

TEXT BOOKS:

1. Database System Concepts. Abraham Silberschatz, Henry F. Korth and S. Sudarshan

REFERENCES:

1. Principles of Database and Knowledge – Base Systems, Vol 1 by J. D. Ullman
2. Fundamentals of Database Systems. R. Elmasri and S. Navathe
3. Foundations of Databases. Serge Abiteboul, Richard Hull, Victor Vianu

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19PC1CB08) SOFTWARE DESIGN WITH UML

COURSE OBJECTIVES:

- To understand how and why models are used during software development and maintenance
- To identify the different perspectives from which software can be modeled
- To distinguish between static and dynamic models and Identify the view or perspective provided by each type of model described
- To be able to create and interpret examples of each type of model described here

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Specify, analyse and organise requirements for a software product
CO-2: Model and analyse software requirements (use case modeling, Interaction Modeling) using UML
CO-3: Apply appropriate UML design patterns for creating static and dynamic models
CO-4: Apply appropriate UML design patterns and notations to the design of components of a product

UNIT – I:
Introduction to Object Oriented Technologies and the UML Method: Software development process: The Waterfall Model vs. The Spiral Model, The Software Crisis, description of the real world using the Objects Model, Classes, inheritance and multiple configurations, Quality software characteristics, Description of the Object Oriented Analysis process vs. the Structure Analysis Model.
Introduction to UML Language: Standards, Elements of the language, General description of various models, The process of Object Oriented software development, Description of Design Patterns, Technological Description of Distributed Systems.

UNIT – II:
Requirements Analysis Using Case Modeling: Analysis of system requirements, Actor definitions, Writing a case goal, Use Case Diagrams, Use Case Relationships.

UNIT – III:
Transfer from Analysis to Design in the Characterization Stage: Interaction Diagrams: Description of goal , Defining UML Method, Operation, Object Interface, Class, Sequence Diagram, Finding objects from Flow of Events, Describing the process of finding objects using a Sequence Diagram, Describing the process of finding objects using a Collaboration Diagram.

UNIT – IV:
The Logical View Design Stage: The Static Structure Diagrams: The Class Diagram Model, Attributes descriptions, Operations descriptions, Connections descriptions in the Static Model, Association, Generalization, Aggregation, Dependency, Interfacing, Multiplicity.

UNIT – V:

Package Diagram Model: Description of the model, White box, black box. Connections between packages., interfaces, Create Package Diagram, Drill Down. Dynamic Model: State Diagram / Activity Diagram: Description of the State Diagram, Events Handling, Description of the Activity Diagram, Exercise in State Machines.

UNIT – VI:

Component Diagram Model: Physical Aspect, Logical Aspect, Connections and Dependencies, User face, Initial DB design in a UML environment.

Deployment Model: Processors, Connections, Components, Tasks, Threads, Signals and Events.

TEXT BOOKS:

1. Object-Oriented Software Engineering: using UML, Patterns, and Java, Bernd Bruegge and Allen H. Dutoit

REFERENCES:

1. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19HS1IE01) INTRODUCTION TO INNOVATION, IP MANAGEMENT AND ENTREPRENEURSHIP

COURSE PRE-REQUISITES: Good knowledge of Fundamentals of Management (Covered in Year 2, Semester 1)

COURSE OBJECTIVES:
The major emphasis of the course will be on creating a learning system through which management students

- To can enhance their innovation and creative thinking skills
- To acquaint themselves with the special challenges of starting new ventures
- To turn market opportunities into a business plan
- To use IPR as an effective tool to protect their innovations and intangible assets from exploitation

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Learn to be familiar with creative and innovative thinking styles
CO-2: Learn to investigate, understand and internalize the process of founding a startup
CO-3: Learn to manage various types of IPR to protect competitive advantage
CO-4: Independently formulate a business plan based on a business idea in technology, plan and implement a development project in a team,

UNIT – I:
Innovation: What and Why?
Innovation as a core business process, Sources of innovation, Knowledge push vs. need pull innovations.
Class Discussion- Is innovation manageable or just a random gambling activity?

UNIT – II:
Building an Innovative Organization: Creating new products and services, Exploiting open innovation and collaboration, Use of innovation for starting a new venture
Class Discussion- Innovation: Co-operating across networks vs. 'go-it-alone' approach

UNIT – III:
Entrepreneurship:

- Opportunity recognition and entry strategies
- Entrepreneurship as a Style of Management
- Maintaining Competitive Advantage- Use of IPR to protect Innovation

UNIT – IV:
Entrepreneurship- Financial Planning:

- Financial Projections and Valuation
- Stages of financing
- Debt, Venture Capital and other forms of Financing

UNIT – V:
Intellectual Property Rights (IPR)

- Introduction and the economics behind development of IPR: Business Perspective
- IPR in India – Genesis and Development
- International Context
- Concept of IP Management, Use in marketing

UNIT – VI:
Types of Intellectual Property

- Patent- Procedure, Licensing and Assignment, Infringement and Penalty
- Trademark- Use in marketing, example of trademarks- Domain name
- Geographical Indications- What is GI, Why protect them?
- Copyright- What is copyright
- Industrial Designs- What is design? How to protect?

Class Discussion- Major Court battles regarding violation of patents between corporate companies

HOME ASSIGNMENT:
Case study materials book will be given to students. Students are required to meet in groups before coming to class and prepare on the case for the day. Instructor may ask the student groups to present their analysis and findings to the class.
Further, the topic for class discussion will be mentioned beforehand and students should be ready to discuss these topics (in groups) in class. Students are required to meet in groups before coming to class and prepare on the topic. Few topics are mentioned below as examples. Instructor can add or change any topic as per requirement.

Topic 1- Is innovation manageable or just a random gambling activity?
Topic 2- Innovation: Co-operating across networks vs. 'go-it-alone' approach
Topic 3- Major Court battles regarding violation of patents between corporate companies

TEXT BOOKS:

1. Managing Innovation: Integrating Technological, Market and Organizational Change, Joe Tidd, John Bessant,
2. Case Study Materials: To be distributed for class discussion

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	2	**1**	**3**

(19ES1ME10) OPERATIONS RESEARCH

COURSE OBJECTIVES:

- To introduces students to use quantitive methods and techniques for effective decisions–making
- To develop operational research models from the verbal description of the real system
- To model formulation and applications that are used in solving business decision problems
- To understand the mathematical tools that are needed to solve optimization problems

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: The characteristics of different types of decision-making environments and the appropriate decision making approaches and tools to be used in each type
CO-2: Build and solve Transportation Models and Assignment Models
CO-3: Design new simple models, like: CPM, MSPT to improve decision – making and develop critical thinking and objective analysis of decision problems
CO-4: Practical and subject specific skills (Transferable Skills). - Be able to implement practical cases, by using TORA, WinQSB

UNIT – I:

Introduction to OR: Origin of OR and its definition. Concept of optimizing performance measure, Types of OR problems, Deterministic vs. Stochastic optimization, Phases of OR problem approach – problem formulation, building mathematical model, deriving solutions, validating model, controlling and implementing solution.

UNIT – II:

Linear Programming: Linear programming–Examples from industrial cases, formulation & definitions, Matrix form.Implicit assumptions of LPP.
Some basic concepts and results of linear algebra – Vectors, Matrices, Linear Independence/Dependence of vectors, Rank, Basis, System of linear eqns., Hyperplane, Convex set, Convex polyhedron, Extreme points, Basic feasible solutions.
Geometric method: 2-variable case, Special cases – infeasibility, unboundedness, redundancy &degeneracy, Sensitivity analysis.
Simplex Algorithm – slack, surplus & artificial variables, computational details, big-M method, identification and resolution of special cases through simplex iterations.
Duality – formulation, results, fundamental theorem of duality, dual-simplex and primal-dual algorithms.

UNIT – III:

Transportation and Assignment problems: TP - Examples, Definitions – decision variables, supply & demand constraints, formulation, Balanced & unbalanced situations, Solution methods – NWCR, minimum cost and VAM, test for optimality(MODI method), degeneracy and its resolution.
AP - Examples, Definitions – decision variables, constraints, formulation, Balanced &unbalanced situations, Solution method – Hungarian, test for optimality (MODI method), degeneracy & its resolution.

UNIT – IV:

PERT – CPM: Project definition, Project scheduling techniques – Gantt chart, PERT & CPM, Determination of critical paths, Estimation of Project time and its variance in PERT using statistical principles, Concept of project crashing/time-cost trade-off.

UNIT – V:

Inventory Control: Functions of inventory and its disadvantages, ABC analysis, Concept of inventory costs, Basics of inventory policy (order, lead time, types), Fixed order-quantity models – EOQ, POQ & Quantity discount models. EOQ models for discrete units, sensitivity analysis and Robustness, Special cases of EOQ models for safety stock with known/unknown stock out situations, models under prescribed policy, Probabilistic situations.

UNIT – VI:

Queuing Theory: Definitions – queue (waiting line), waiting costs, characteristics (arrival, queue, service discipline) of queuing system, queue types (channel vs. phase).

Kendall's notation, Little's law, steady state behaviour, Poisson's Process & queue, Models with examples - M/M/1 and its performance measures; M/M/m and its performance measures; brief description about some special models.

Simulation Methodology:

Definition and steps of simulation, random number, random number generator, Discrete Event System Simulation – clock, event list, Application in Scheduling, Queuing systems and Inventory systems.

TEXT BOOKS:

1. Operations Research: An Introduction, H.A. Taha
2. Principles of OR with Application to Managerial Decisions, H.M. Wagner
3. Introduction to Operations Research, F.S. Hiller and G.J. Lieberman

REFERENCES:

1. Linear Programming, K.G. Murthy
2. Elements of Queuing Theory, Thomas L. Saaty
3. Operations Research and Management Science, Hand Book: Edited By A. Ravi Ravindran
4. Management Guide to PERT/CPM. Wiest & Levy
5. Modern Inventory Management, J.W. Prichard and R.H. Eagle

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19PC2CB04) OPERATING SYSTEMS (UNIX) LABORATORY

COURSE OBJECTIVES:
- To implement various CPU Scheduling Algorithms
- To implement Process Creation and Inter Process Communication.
- To implement Deadlock Avoidance and Deadlock Detection Algorithms
- To implement Page Replacement Algorithms
- To implement File Organization and File Allocation Strategies

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Compare the performance of various CPU Scheduling Algorithms
CO-2: Implement Deadlock avoidance and Detection Algorithms
CO-3: Analyze the performance of the various Page Replacement Algorithms
CO-4: Implement File Organization and File Allocation Strategies

WEEK 1:
Write programs using the following system calls of UNIX operating system fork, exec, getpid, exit, wait, close, stat, opendir, readdir

WEEK 2:
Write C programs to simulate UNIX commands like cp, ls, grep, etc.

WEEK 3:
Shell Programming

WEEK 4:
Write C programs to implement the various CPU Scheduling Algorithms

WEEK 5:
Implementation of Semaphores

WEEK 6:
Implementation of Shared memory and IPC

WEEK 7:
Bankers Algorithm for Deadlock Avoidance

WEEK 8:
Implementation of Deadlock Detection Algorithm

WEEK 9:
Write C program to implement Threading & Synchronization Applications

WEEK 10:
Implementation of the following Memory Allocation Methods for fixed partition
a) First Fit b) Worst Fit c) Best Fit

WEEK 11:
Implementation of Paging Technique of Memory Management

WEEK 12:
Implementation of the following Page Replacement Algorithms
a) FIFO b) LRU c) LFU

WEEK 13:
Implementation of the various File Organization Techniques

WEEK 14:
Implementation of the following File Allocation Strategies
a) Sequential b) Indexed c) Linked

TEXT BOOKS:
1. Operating System Concepts Essentials, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne

REFERENCES:
1. Operating Systems: Internals and Design Principles, William Stallings
2. Operating System: A Design-oriented Approach, Charles Patrick Crowley
3. Operating Systems: A Modern Perspective, Gary J. Nutt
4. Design of the Unix Operating Systems, Maurice J. Bach
5. Understanding the Linux Kernel, Daniel Pierre Bovet, Marco Cesati

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19PC2CB05) DATATBASE MANGEMENT SYSTEM CONCEPT LABORATORY

COURSE OBJECTIVES:
- To provide the fundamental concepts of database creation
- To implement the concepts of Data manipulation
- To develop procedures for querying multiple tables
- To understand the concepts of PL / SQL

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Implement SQL functions using the DUAL table
CO-2: Apply Integrity constraints for creating consistent RDBMS environment
CO-3: Create, maintain and manipulate the Data through SQL commands
CO-4: Develop Triggers, query through PL /SQL structures

WEEK 1:
Implement the following using DUAL table:
a. Character functions.
b. Numeric functions.
c. Date functions.
d. Conversion functions.

WEEK 2:
Practice DDL and DML commands on a basic table without integrity constraints.

WEEK 3:
Practice DDL and DML commands on a Relational Database, specifying the Integrity constraints. (Primary Key, Foreign Key, CHECK, NOT NULL)

WEEK 4:
Apply the concepts of Joins, SET operations and SQL functions on any two relational schemas.

WEEK 5-7
Apply the concepts of Joins, SET operations and SQL functions on the following schema:

a) Employee:

Name	**Datatype**	**width**	**Constraint**	**Description**
Empno	Integer	4	Primary Key	Employee Number
Ename	Varchar	20		Employee Name
Job	Char	12		Designation
Mgr	Integer	4		Manager Number
Hiredate	Date			

Sal	Number	(8,2)		Salary
Comm	Number	(6,2)		Commission
Deptno	Integer	2	Foreign Key	Department Number

b) Dept:

Name	**Datatype**	**width**	**Constraint**	**Description**
Deptno	Integer	2	Primary Key	Department Number
Dname	Varchar	12		Department Name
Loc	Char	10		Location

c) Salgrade:

Name	**Datatype**	**width**	**Constraint**	**Description**
Grade	Integer	1		Grade
Hisal	Integer	4		Upper scale of salary
Losal	Integer	5		Lower scale of salary

WEEK 8:
Sessional Examination-I

WEEK 9 – 12:
End to end implementation of a schema for a specific system along with the illustrations of querying.
A system is described by specifying the functional and non-functional requirements. Based on this description, the major entities are identified and modelled. Further the relationships are modelled to form the initial schema. The schema is further refined by removing redundancies through normalization. Also based on the query requirements, the schema is remodelled to facilitate querying. Finally an illustration of various queries to extract required information from the system is shown using SQL / MYSQL.

The five major workflows to be implemented are:

1. System Specification
2. Design of Initial Schema
3. Schema refinement using functional dependencies and normalization
4. Schema refinement using query requirements
5. Illustration of querying the system using SQL / MYSQL.

WEEK 13:
Implementation of PL / SQL concepts

WEEK 14:
Creating and executing Cursors.

WEEK 15:
Creation and application of TRIGGERS on a Relational schema.

WEEK 16:
Sessional Examination-II

TEXT BOOKS:
1. Database System Concepts, Abraham Silberschatz, Henry F. Korth and S. Sudarshan

REFERENCES:
1. Principles of Database and Knowledge – Base Systems, Vol 1 by J. D. Ullman
2. Fundamentals of Database Systems, R. Elmasri and S. Navathe
3. Foundations of Databases, Serge Abiteboul, Richard Hull, Victor Vianu

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19PC2CB06) SOFTWARE DESIGN WITH UML LABORATORY

COURSE OBJECTIVES:

- To learn good design, good modeling practices, document them and be able to discuss the pros and cons of your designs and models
- To draw the UML diagrams for the given specification
- To map the design properly to code
- To improve the design by applying appropriate design patterns

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Capture the requirements specification for an intended software system develop structural models with the tool

CO-2: Develop static and behavioural models using UML as per the requirements gathered

CO-3: Select the set of appropriate diagrams to develop a system with UML

CO-4: Design UML diagrams for any complex problem

WEEK 1,2,3

1. All diagrams for Online course reservation system:

WEEK 4,5,6

4. All diagrams for Exam Registration

WEEK 7,8,9

5. All diagrams for ATM simulation

WEEK 10,11,12

6. Simulate railway registration system

WEEK 13

Lab Internals

WEEK 13,14,15

6. All diagrams for Online shopping

TEXT BOOKS:

1. Object-Oriented Software Engineering: using UML, Patterns, and Java, Bernd Bruegge and Allen H. Dutoit

REFERENCES:

1. Design Patterns: Elements of Reusable Object-Oriented Software, Erich Gamma, Richard Helm, Ralph Johnson, and John M. Vlissides

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	0	**2**	**1**

(19ES2ME10) OPERATIONS RESEARCH LABORATORY

COURSE OBJECTIVES:

- To understand the concept of graphical analysis of LPP
- To identify a transportation problem and implement various transportation models
- To derive the solution of an assignment problem and integer programming problems using various methods
- To understand Classification of Queuing Models (Finite and Infinite Models) and the simulation models

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Give solution of any LPP by using simplex algorithm
CO-2: Implement feasible, basic, non-degenerate solutions of a transportation problem
CO-3: Implement assignment problem that will be balanced and also solve integer programming problem
CO-4: Implement queuing theory techniques and simulation models

WEEK 1, 2, 3
Experiment 1. To solve Linear Programming Problem using Graphical Method with

I. Unbounded solution
II. Infeasible solution
III. Alternative or multiple solutions.

WEEK 4 ,5, 6
Experiment 2. Solution of LPP with simplex methods.

WEEK 7, 8 ,9
Experiment 3. Solving transportation problems.

WEEK 10,11,12
Experiment 4. solving assignment problems.

WEEK 13,14,15,16
Experiment 5. Solving queuing and simulation models

TEXT BOOKS:

1. Operations Research: An Introduction, H.A. Taha
2. Principles of OR with Application to Managerial Decisions, H.M. Wagner
3. Introduction to Operations Research, F.S. Hiller and G.J. Lieberman

REFERENCES:

1. Linear Programming, K.G. Murthy
2. Elements of Queuing Theory, Thomas L. Saaty
3. Operations Research and Management Science, Hand Book: Edited By A. Ravi Ravindran
4. Management Guide to PERT/CPM, Wiest & Levy
5. Modern Inventory Management, J.W. Prichard and R.H. Eagle

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester - CSBS	**L**	**T/P/D**	**C**
	1	**2**	**2**

(19HS2EN04) BUSINESS COMMUNICATION AND VALUE SCIENCE -III

COURSE PRE-REQUISITE(S):
1. Basic Knowledge of English (verbal and written)
2. Completion of all units from Semesters 1, 2 and 3

COURSE OBJECTIVES:
- To enable the students to create clear, accurate, and succinct content to write business letters, resume, SOP, Proposals and Technical Reports for academics as well as for workplace
- To introduce students to Self-analysis techniques like SWOT & TOWS
- To introduce students to key concepts of Pluralism, cultural spaces and Cross-cultural communication
- To understand the importance of science for Nation building

COURSE OUTCOMES: After completion of the course, the student should be able to
CO-1: Identify the best practices of technical writing and Understand, apply &analyze the tools of technical writing
CO-2: Apply &analyze the basic principles of SWOT & life positions
CO-3: Identify & respect pluralism in cultural spaces- Analyze cross cultural communication and apply the concepts of Global, glocal and translocational; and Recognize the roles and relations of different genders
CO-4: Apply the science of Nation building

UNIT – I:
Basic principles of SWOT and Life Positions.
i. SWOT and Life Positions
ii. Apply SWOT in real life scenarios, Create SWOT
iii. SWOT Vs. TOWS-The Balancing Act
iv. Importance of Motivation in real life
v. Leverage motivation in real-life scenarios.

UNIT – II:
Pluralism in cultural spaces
i. Awareness and respect for pluralism in cultural spaces
ii. Rhythms of India (Cultures in India)
iii. Defibe and Differentiate -Global, glocal, translocational
iv. Cross-cultural communication- Culture shock
v. Gender awareness

UNIT – III:
Role of science in nation building
i. Role of scientists and mathematicians from ancient India.
ii. Role of science post- independence
iii. Inventions –Inventors-Institutes-Information technology
iv. Introduction to technical writing

v. Basic rules of technical writing

UNIT – IV:
Artificial intelligence –Voice of the future

i. Artificial intelligence in Everyday Life
ii. Communicating with machines
iii. Applying technical writing in profession
iv. Scenario-based technical writing
v. Best practices of Technical writing

UNIT – V:
Technical Writing

i) Summarizing & Synthesising
ii) Abstract Writing
iii) Report Writing
iv) Product Description
v) Description of a mechanism

UNIT – VI:
Project Work

- Visit rural area/ underprivileged parts of city to address some of the local issues; if relevant, suggest a practical technology solution to the issues.

TEXTBOOKS:
There are no prescribed texts for Semester 4 – there will be handouts and reference links.

REFERENCES:

1. Effective Technical Communication (2005), Ashraf, Rizvi M, New Delhi: Tata Mc Graw Hill Publishing Company Limited, 2nd Edition
2. Technical Communication: A Reader-Centered Approach (2003), Anderson, Paul V. Reports In Paul V. Anderson's 9th Edition, Boston: Heinle
3. Technical Communication: A Practical Approach, (2012) William S. Pfeiffer, 8th Edition, Pearson
4. Technical Communication (2001), Burnett, Rebecca, 6th Edition, Cengage Learning

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

B.Tech. IV Semester	**L**	**T/P/D**	**C**
	0	**2**	**0**

(19MN6HS05) ESSENCE OF INDIAN TRADITIONAL KNOWLEDGE

COURSE OBJECTIVE:

- To give exposure to the repositories of our indigenous knowledge and wisdom
- To familiarize on how this knowledge and wisdom has evolved over centuries
- To map on how knowledge and wisdom still continue to serve social functions
- To understand how knowledge and wisdom help in cultural functions

COURSE OUTCOMES: After completion of the course, the student should be able to

CO-1: Understand basic principles, thought process, reasoning and inference of Indian Traditional Knowledge Systems

CO-2: Recognize wisdom of Sanskrit literature and its importance in modern society with rapid technological advancements

CO-3: Be familiar with scientific worldview and basic principles of Yoga and holistic health care system

CO-4: Understand that sustainability is at the core of Indian Traditional Knowledge Systems connecting society and nature

UNIT – I:

(Basic Structure of Indian Knowledge System-Part 1): The historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK), Traditional Knowledge (TK) Vs western knowledge traditional knowledge vis-à-vis formal knowledge. Significance of TK Protection, value of TK in global economy, Role of Government to harness TK.

UNIT – II:

(Basic Structure of Indian Knowledge System-Part 2): The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights) Act, 2006, Plant Varieties Protection and Farmer's Rights Act, 2001 (PPVFR Act).

UNIT – III:

(Modern Science and Indian Knowledge System-Part 1): Mathematics in India, Early Historical Period, The Classical Period, The Classical Period, post-Āryabhaṭa, Features of Indian Mathematics.

UNIT – IV:

(Modern Science and Indian Knowledge System-Part 2): Early Chemical Techniques, Atomism in Vaiśeṣika, Chemistry in Early Literature, Indian Philosophy Sāṁkhya, Yoga, Vaiśeṣika, Nyāya, Mīmāṁsā, Vedānta, Sāṁkhya.

UNIT – V:

(Yoga and Holistic Health care- Part 1): Ayurveda for Life, Health and Well-being Definition of Ayurveda,

UNIT – VI:

(Yoga and Holistic Health care- Part 2): The Principles of Ayurvedic Healing, Treating diseases to restore health, Astanga Ayurveda.

REFERENCES:

1. Cultural Heritage of India-course material, by Sivaramakrishnan (Ed.), Bharatiya Vidya Bhavan, Mumbai, 5th Edition, 2014
2. Holistic Science and Vedant, by Swami Jitatmanand, Bharatiya Vidya Bhavan
3. "Knowledge Traditions and Practices of India" Kapil Kapoor, Michel Danino
4. Science of Consciousness Psychotherapy and Yoga Practices, by RN Jha, Vidyanidhi Prakasham, Delhi, 2016

E-Resources:

1. https://www.youtube.com/watch?v=LZP1StpYEPM
2. http://nptel.ac.in/courses/121106003/