R18

M.Tech. (SOFTWARE ENGINEERING)

M.Tech. R18 CBCS Curriculum

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

An Autonomous, ISO 9001:2015 & QS I-Gauge Diamond Rated Institute, Accredited by NAAC with 'A++' Grade
NBA Accreditation for B.Tech. CE, EEE, ME, ECE, CSE, EIE, IT Programmes
Approved by AICTE, New Delhi, Affiliated to JNTUH, NIRF 127 Rank in Engineering Category
Recognized as "College with Potential for Excellence" by UGC
Vignana Jyothi Nagar, Pragathi Nagar, Nizampet (S.O), Hyderabad – 500 090, TS, India.
Telephone No: 040-2304 2758/59/60, Fax: 040-23042761
E-mail: postbox@vnrvjiet.ac.in, Website: www.vnrvjiet.ac.in

VISION OF THE INSTITUTE

To be a World Class University providing value-based education, conducting interdisciplinary research in cutting edge technologies leading to sustainable development of the nation

MISSION OF THE INSTITUTE

- To produce technically competent and socially responsible engineers, managers and entrepreneurs, who will be future ready.
- To involve students and faculty in innovative research projects linked with industry, academic and research institutions in India and abroad.
- To use modern pedagogy for improving the teaching-learning process.

DEPARTMENT OF

COMPUTER SCIENCE AND ENGINEERING

VISION OF THE DEPARTMENT

To achieve academic and research excellence in essential technologies of Computer Science and Engineering by promoting a creative environment for learning and innovation.

MISSION OF THE DEPARTMENT

- To provide dynamic, innovative and flexible curriculum which equip the students with the necessary,
- Problem driven skills to strengthen their career prospects and potential to pursue higher studies.
- To foster inquisitive-driven research among students and staff so as to reinforce the domain knowledge and address contemporary societal issues.

- To inculcate ethical values, leadership qualities and professional behaviour skills for improving the living standards of people

M.TECH.
(SOFTWARE ENGINEERING)

M.TECH. (SE)

PROGRAM EDUCATIONAL OBJECTIVES

PEO-I. Develop technologically competent computer professionals in todays IT centric scenario by training them in the contemporary software engineering principles and paradigms.

PEO-II. Provide students a deep insight into various cutting-edge technologies & tools and thereby creating diverse career opportunities.

PEO-III. Improve analytical, logical and presentation skills of the students by applying evolving technologies of software engineering in developing practical solutions to complex problems in consonance with the legal and ethical responsibilities.

PEO-IV. Provide the students with project engineering and management skills catering to the changing industry needs and constraints across the advancing domains of computing.

PEO-V. Prepare the students to take up research oriented projects, industry internships and entrepreneurship endeavours by training them to work with multi-disciplinary teams and engaging them for life-long learning in pursuit of their professional accomplishment.

M.TECH. (SE)

PROGRAM OUTCOMES

PO-1: An ability to independently carry out research / investigation and development work to solve practical problems.

PO-2: An ability to write and present a substantial technical report / document.

PO-3: An ability to demonstrate a degree of mastery over the area as per the specialization of the program.

PO-4: An ability to integrate the knowledge of software engineering principles and paradigms in the design of system components and processes to meet the specific needs of the industry.

PO-5: An ability to apply the cutting-edge technologies, skills and CASE tools necessary to identify, analyze and formulate solutions to complex engineering problems with societal commitment.

PO-6: An ability to recognize the need to engage in lifelong learning that helps to explore all dimensions of software engineering practices and contemporary technologies with ethical values.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY, HYDERABAD
M.TECH. I YEAR COURSE STRUCTURE AND SYLLABUS

(SOFTWARE ENGINEERING)

I SEMESTER **R18**

Course Type	Course Code	Name of the Course	L	T	P	Credits
Professional Core-I	18PC1CP01	Mathematical Foundations of Computer Science	3	0	0	3
Professional Core-II	18PC1CP02	Advanced Data Structures	3	0	0	3
Professional Core-III	18PC1SE01	Data Analytics	3	0	0	3
Professional Elective-I	18PE1SE01	Software Requirements and Estimation	3	0	0	3
	18PE1SE02	Object Oriented Modeling				
	18PE1SE03	Android Application Development				
Professional Elective -II	18PE1ES02	Internet of Things	3	0	0	3
	18PE1CP03	Ethical Hacking and Computer Forensics				
	18PE1SE04	Formal Methods in Software Engineering				
Professional Core Lab-I	18PC2CP01	Advanced Data Structures Laboratory	0	0	3	1.5
Professional Core Lab-II	18PC2SE01	Data Analytics Laboratory	0	0	3	1.5
Project	18PW4SE01	Technical Seminar	0	0	4	2
Audit	18AU5CS01	Research Methodology and IPR	2	0	0	0
Total			**17**	**0**	**10**	**20**

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY, HYDERABAD
M.TECH. I YEAR COURSE STRUCTURE AND SYLLABUS

(SOFTWARE ENGINEERING)

II SEMESTER **R18**

Course Type	Course Code	Name of the Course	L	T	P	Credits
Professional Core-IV	18PC1CP04	Advanced Algorithms	3	0	0	3
Professional Core-V	18PC1CP05	Soft Computing	3	0	0	3
Professional Core-VI	18PC1SE02	Software Quality Assurance and Testing	3	0	0	3
Professional Elective-III	18PE1SE05	Advanced Software Engineering Principles and Practices	3	0	0	3
	18PE1SE06	Web Services and Service Oriented Architecture				
	18PE1SE07	Scripting Languages				
Professional Elective -IV	18PE1SE08	Software Reengineering	3	0	0	3
	18PE1CP04	Cloud Computing				
	18PE1SE09	Software Architecture and Design Patterns				
Professional Core Lab-III	18PC2CP03	Advanced Algorithms Laboratory	0	0	3	1.5
Professional Core Lab-IV	18PC2SE02	Software Testing and Case Tools Laboratory	0	0	3	1.5
Project	18PW4SE02	Mini-Project	0	0	4	2
Audit	18AU5EN01	English for Academic and Research Writing	2	0	0	0
Total			**17**	**0**	**10**	**20**

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY, HYDERABAD
M.TECH. II YEAR COURSE STRUCTURE AND SYLLABUS

(SOFTWARE ENGINEERING)

III SEMESTER **R18**

Course Type	Course Code	Name of the Course	L	T	P	Credits
Professional Elective-V	18PE1CP11	Blockchain Technology	3	0	0	3
	18PE1SE10	Software Defined Networks				
	18PE1SE11	Software Metrics				
Open Elective	18OE1CN01	Business Analytics	3	0	0	3
	18OE1AM01	Industrial Safety				
	18OE1AM02	Operations Research				
	18OE1AM03	Composite Materials				
	18OE1PS01	Waste to Energy				
Project	18PW4SE03	Project Part - I	0	0	16	8
Total			**6**	**0**	**16**	**14**

IV SEMESTER **R18**

Course Type	Course Code	Name of the Course	L	T	P	Credits
Project	18PW4SE04	Project Part - II	0	0	28	14
Total			**0**	**0**	**28**	**14**

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PC1CP01) MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

COURSE PRE-REQUISITES: Discrete Mathematics

COURSE OBJECTIVES:

- To understand the mathematical fundamentals that is prerequisites for a variety of courses like Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Computer architecture, operating systems, distributed systems, Bioinformatics, Machine learning
- To develop the understanding of the mathematical and logical basis to many modern techniques in information technology like machine learning, programming language design, and concurrency
- To study various sampling and classification problems
- To construct various kinds of graphs

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Understand the basic notions of discrete and continuous probability
CO-2: Understand the methods of statistical inference, and the role that sampling distributions play in those methods
CO-3: Able to perform correct and meaningful statistical analyses of simple to moderate complexity
CO-4: Analyze the techniques and applications of graph theory

UNIT-I:
Probability mass, density, and cumulative distribution functions, Parametric families of distributions, Expected value, variance, conditional expectation, Applications of the univariate and Central Limit Theorem, Probabilistic inequalities, Markov chains.

UNIT-II:
Random samples, sampling distributions of estimators, Methods of Moments and Maximum Likelihood.

UNIT-III:
Statistical inference, Introduction to multivariate statistical models: regression and classification problems, principal components analysis, The problem of overfitting model assessment.

UNIT-IV:
Graph Theory: Isomorphism, Planar graphs, graph colouring, Hamilton circuits and Euler cycles. Permutations and Combinations with and without repetition. Specialized techniques to solve combinatorial enumeration problems.

UNIT-V:
Computer science and engineering applications Data mining, Network protocols, analysis of Web traffic, Computer security, Software engineering, Computer architecture, operating systems, distributed systems, Bioinformatics, Machine learning.

UNIT-VI:
Recent Trends in various distribution functions in mathematical field of computer science for varying fields like bioinformatics, soft computing, and computer vision.

TEXT BOOKS:

1. Foundation Mathematics for Computer Science, John Vince, Springer
2. Probability and Statistics with Reliability, Queuing, and Computer Science Applications, K. Trivedi, Wiley

REFERENCES:

1. Probability and Computing: Randomized Algorithms and Probabilistic Analysis, M. Mitzenmacher and E. Upfal
2. Applied Combinatorics, Aln Tucker, Wiley

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)

L	T/P	C
3	0	3

(18PC1CP02) ADVANCED DATA STRUCTURES

COURSE PRE-REQUISITES: Data Structures

COURSE OBJECTIVES:

- To choose appropriate data structures, understand the ADT/libraries, and use it to design algorithms for a specific problem
- To understand the necessary mathematical abstraction to solve problems
- To familiarize students with advanced paradigms and data structure used to solve algorithmic problems
- To come up with analysis of efficiency and proofs of correctness

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

CO-1: Understand the implementation of symbol table using hashing techniques
CO-2: Develop and analyse algorithms for red-black trees, B-trees and Splay trees
CO-3: Develop algorithms for text processing applications
CO-4: Identify suitable data structures and develop algorithms for computational geometry problems

UNIT-I:

Algorithms, Performance analysis- time complexity and space complexity, Complexity Analysis Examples. Data structures-Linear and nonlinear data structures, ADT concept, Linear List ADT, Array representation, Linked representation, singly linked lists -insertion, deletion, search operations, doubly linked lists-insertion, deletion operations,

UNIT-II:

Dictionaries: Definition, Dictionary Abstract Data Type, Implementation of Dictionaries, **Hashing:** Review of Hashing, Hash Function, Collision Resolution Techniques in Hashing, Separate Chaining, Open Addressing, Linear Probing, Quadratic Probing, Double Hashing, Rehashing, Extendible Hashing.

Searching (linear and binary with and without recursion), counting and radix sorting techniques.

UNIT-III:

Trees: Binary Search Trees, AVL Trees, Red Black Trees, 2-3 Trees, B-Trees, Splay Trees.

UNIT-IV:

Text Processing: Sting Operations, Brute-Force Pattern Matching, The Boyer-Moore Algorithm, The Knuth-Morris -Pratt Algorithm, The Longest Common Subsequence Problem (LCS), Applying Dynamic Programming to the LCS Problem.

UNIT-V:

Computational Geometry: One Dimensional Range Searching, Two Dimensional Range Searching, Introduction to Convex Hull, finding the convex hull and Graham scan approach.

UNIT-VI:

Recent Trends in Hashing (Introduction of tries), Tries, and various computational geometry methods for efficiently solving the new evolving problem (k-D trees)

TEXT BOOKS:

1. Data Structures and Algorithm Analysis in C++, Mark Allen Weiss, 2nd Edition, Pearson, 2004
2. Algorithm Design, M T Goodrich, Roberto Tamassia, John Wiley, 2002

REFERENCES:

1. The Design and Analysis of Computer Algorithms, Aho, Hopcroft, Ullman
2. Algorithm Design, Kleinberg and Tardos
3. Introduction to Algorithms, Cormen, Leiserson, Rivest, Stein

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)

L	T/P	C
3	0	3

(18PC1SE01) DATA ANALYTICS

COURSE OBJECTIVES:
- To explore the fundamental concepts of big data analytics
- To learn to analyze the data analysis techniques
- To explore the techniques related to mining streams
- To understand and explore Hadoop environment and it's associated tools & techniques

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Work with big data platform and analyze the big data analytic techniques for useful business applications
CO-2: Design efficient algorithms for mining the data from large volumes
CO-3: Learn to use various techniques for mining data streams
CO-4: Analyze the Hadoop and Map Reduce technique and related tools & techniques associated with big data analytics.

UNIT-I:
Big Data and Data Analysis: Introduction to Big Data Platform, Challenges of Conventional Systems, Web Data, Evolution of Analytic Scalability, Analytic Processes and Tools, Analysis vs Reporting, Modern Data Analytic Tools. Statistical Concepts: Sampling Distributions, Re-Sampling, Statistical Inference - Prediction Error, Regression Modeling, Multivariate Analysis.

UNIT-II:
Classification and Clustering:
Classification: Rule Based Classifier, Nearest neighbor classifiers, Artificial Neural Network, Support Vector Machine.
Cluster Analysis: Overview, K-Means, Agglomerative, Hierarchical Clustering, Prototype based clustering.

UNIT-III:
Mining Data Streams: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, Counting Oneness in a Window, Decaying Window.

UNIT-IV:
Mining Frequent Item Sets: Mining Frequent Item-sets, Market Based Model, A-Priori Algorithm, Handling Large Data Sets in Main Memory, Limited Pass Algorithms, Counting Frequent Item-sets in a Stream.

UNIT-V:
Hadoop: Meet Hadoop, Comparison with other systems, A brief history of Hadoop and the Hadoop ecosystem, Hadoop Distributed File System, Components of Hadoop, Analyzing the Data with Hadoop, Scaling Out- Hadoop Streaming- Design of HDFS- Java interfaces to HDFS Basics- Developing a Map Reduce Application-How Map Reduce Works-Anatomy of a Map Reduce Job run-Failures-Job Scheduling-Shuffle and Sort – Task execution - Map Reduce Types and Formats- Map Reduce Features

UNIT-VI:
Frameworks and Visualization: Applications on Big Data Using Pig and Hive – Data processing operators in Pig – Hive services –HiveQL – Querying Data in Hive - fundamentals of HBase and ZooKeeper.

Visusalizations- Visual Data Analysis Techniques, Interaction Techniques.

.

TEXT BOOKS:

1. Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, Bill Franks, John Wiley & Sons, 2012
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Pearson
3. Mining of Massive Datasets, Anand Rajaraman and Jeffrey David Ullman, Cambridge University Press, 2012
4. Hadoop: The Definitive Guide, Tom White, Third Edition, O'Reilly, 2012

REFERENCES:

1. Making Sense of Data, Glenn J. Myatt, John Wiley & Sons, 2007
2. Big Data Glossary, Pete Warden, O'Reilly, 2011
3. Data Mining Concepts and Techniques, Jiawei Han, Micheline Kamber, Second Edition, Elsevier, Reprinted 2008

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE01) SOFTWARE REQUIREMENTS AND ESTIMATION

COURSE OBJECTIVES:
- To learn the concepts of Software requirements and management process
- To understand the stages of requirements engineering
- To know the techniques used for requirements development and modeling
- To study and analyze the estimation methods

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify and classify the requirements
CO-2: Select appropriate elicitation technique for a system
CO-3: Develop various requirements models
CO-4: Apply the change management process

UNIT-I:
Software Requirements: What and Why: Essentials of Software requirement Good practices for requirements engineering, Improving requirements processes, Software requirements and risk management.

UNIT-II:
Software Requirements Engineering: Requirements elicitation, requirements analysis documentation, review, elicitation techniques, analysis models, Software quality attributes, risk reduction through prototyping, setting requirements priorities, verifying requirements quality.

UNIT-III:
Software Requirements Modeling: Use Case Modeling, Analysis Models, Dataflow diagram, state transition diagram, class diagrams, Object analysis.

UNIT-IV:
Software Requirements Management: Requirements management Principles and practices, Requirements attributes, Change Management Process, Requirements Traceability Matrix.
Requirements Management Tools: Benefits of using a requirements management tool, commercial requirements management tool, Rational Requisite pro, Caliber – RM

UNIT-V:
Software Estimation: Components of Software Estimations, Estimation methods, Problems associated with estimation, Key project factors that influence estimation.
Size Estimation: Two views of sizing, Function Point Analysis, Mark II FPA, Full Function Points, LOC Estimation.

UNIT-VI:
Effort, Schedule and Cost Estimation: Productivity, Estimation Factors, Approaches to Effort and Schedule Estimation, COCOMO II
Software Estimation Tools: Desirable features in Software estimation tools, IFPUG, SLIM (Software Life Cycle Management) Tools

TEXT BOOKS:
1. Software Requirements, Karl E. Weigers, Microsoft Press
2. Software Requirements and Estimation, Rajesh Naik and Swapna Kishore, Tata McGraw Hill

REFERENCES:

1. Managing Software Requirements, Dean Leffingwell & Don Widrig, Pearson Education, 2003
2. Mastering the Requirements Process, Suzanne Robertson & James Robertson, Second Edition, Pearson Education, 2006
3. Estimating Software Costs, Capers Jones, Second Edition, Tata McGraw-Hill, 2007
4. Practical Software Estimation, M.A. Parthasarathy, Pearson Education, 2007
5. Measuring the Software Process, William A. Florac & Anita D. Carleton, Pearson Education, 1999

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)

L	T/P	C
3	0	3

(18PE1SE02) OBJECT ORIENTED MODELING

COURSE OBJECTIVES:

- To list and discuss the object-oriented concepts, principles and the artifacts that are related to classes and relationships for modeling (CRC approach)
- To illustrate the communication across the objects for a specified behavior (use case) through interaction diagrams
- To outline the various states a process/thread undergoes based on occurrence of events and the mode for deploying the components being built
- To explain the industry software modeling practice through various processes

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Relate object-oriented concepts representation through artifacts of UML
CO-2: Address the given case study through classes, their relationships and collaborations (CRC) approach or Use case Driven Approach
CO-3: Generate the list and order of activities carried out for each behavior exhibited by the system to design the model
CO-4: Apply unified process approach for case studies

UNIT-I:
Introduction to UML: The meaning of Object Orientation, object identity, Encapsulation, information hiding, polymorphism, generosity, importance of modeling, principles of modeling, object-oriented modeling, conceptual model of the UML, Architecture.

UNIT-II:
Basic Structural Modeling: Classes, Relationships, common Mechanisms, and diagrams. Class & Object Diagrams: Terms, concepts, modeling techniques for Class & Object Diagrams.
Advanced Structural Modeling: Advanced classes, Advanced relationships, Interfaces types and roles, packages, instances and object diagrams.

UNIT-III:
Collaboration Diagrams: Terms, Concepts, depicting a message, polymorphism in collaboration diagrams, iterated messages, use of self in messages.
Sequence Diagrams: Terms, concepts, depicting asynchronous messages with/without priority, callback mechanism, broadcast messages.

UNIT-IV:
Basic Behavioral Modeling: Use cases, Use case Diagrams, Activity Diagrams.
Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.
Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.

UNIT-V:
The Unified Process: use case driven, architecture centric, iterative, and incremental
The Four Ps: people, project, product, and process.
Use Case Driven Process: why use case, capturing use cases, analysis, design, and implementation to realize the use cases, testing the use cases.

UNIT-VI:
Architecture-Centric Process: Architecture in brief, why we need architecture, use cases and architecture, the steps to architecture, an architecture description.

Iterative Incremental Process: iterative incremental in brief, why iterative incremental development? The iterative approach is risk driven, the generic iteration.
Case Studies: Automation of a Library, Software Simulator application.

TEXT BOOKS:
1. The Unified Modelling Language User Guide, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education
2. The Unified Software Development Process, Ivar Jacobson, Grady Booch, James Rumbaugh, Pearson Education

REFERENCES:
1. Fundamentals of Object-Oriented Design in UML, Meilir Page-Jones, Pearson Education
2. Modeling Software Systems Using UML2, Pascal Roques, WILEY- Dreamtech
3. Object Oriented Analysis & Design, Atul Kahate, The McGraw-Hill Companies
4. Cengage Learning, Object-Oriented Analysis and Design with the Unified Process, John W. Satzinger, Robert B Jackson and Stephen D Burd
5. UML 2 Toolkit, Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado, Wiley Dreamtec

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE03) ANDROID APPLICATION DEVELOPMENT

COURSE OBJECTIVES:

- To understand the differences between android development environments and other public development environments
- To design and develop useful android applications with compiling user interfaces by using extending and creating own layouts and views by menus
- To demonstrate the advantage of android's API's for data storage, retrieval, user preferences, files, databases and content providers
- To design location-based services and create rich map based applications

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Appreciate the Mobility landscape
CO-2: Familiarize with Mobile apps development aspects
CO-3: Design and develop mobile apps, using Android as development platform, with key focus on user experience design, native data handling and background tasks and notifications
CO-4: Perform testing, signing, packaging and distribution of mobile apps

UNIT-I:
Introduction to Android: History of Mobile Software Development, The Open Handset Alliance, The Android Platform, Android SDK, Android Studio Installation, Building a Sample Android application.

UNIT-II:
Android Application Design Essentials: Anatomy of an Android applications, Android terminologies, Application Context, Activities, Intents, Receiving and Broadcasting Intents, User Interface Screen elements, Designing User Interfaces with Layouts, Fragments, Menus.

UNIT-III:
Working with Android Resources: Android Manifest File and its common settings, Using Intent Filter, Permissions, Working with different types of resources.

UNIT-IV:
Android Services: Android Service, Android Service API, Android Started Service, Android Bound Service, Android Service Life Cycle, Android Service Example.

UNIT-V:
Using Common Android APIs: Using Android Data and Storage APIs, Shared Preferences, Sharing Data between Applications with Content Providers, Managing data using SQLite.

UNIT-VI:
Device Connectivity: Bluetooth Tutorial, List Paired Devices, Working with WiFi , Working with Camera

Android Google Map: Android Map V2 API, Adding Map, Customizing Map, Google Map class, Android Google Map Application.

TEXT BOOKS:
1. J2ME: The Complete Reference, James Keogh, Tata McGraw Hill
2. Android Wireless Application Development, Lauren Darcey and Shane Conder, Pearson Education, 2nd Ed., 2011

REFERENCES:

1. Professional Android 2 Application Development, Reto Meier, Wiley India Pvt. Ltd.
2. Beginning Android, Mark L. Murphy, Wiley India Pvt. Ltd.
3. Pro Android, Sayed Y. Hashimi and Satya Komatineni, Wiley India Pvt. Ltd.
4. devloper.android.com (web)
5. Android Application Development All in one for Dummies, Barry Burd, 1st Edition

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)

L	T/P	C
3	0	3

(18PE1ES02) INTERNET OF THINGS

COURSE PRE-REQUISITES: Concepts of Programming in Java, C/C++, Embedded C, Concepts of Wireless Communication and Networking

COURSE OBJECTIVES:

- To understand the new paradigm of objects interacting with people, information systems and with other objects
- To introduce various IoT protocols
- To understand the issues in developing specific real time systems on various IoT platforms.

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify and describe different kinds of internet-connected products developed on various IoT platforms
CO-2: Appreciate the challenges involved in establishing user-interaction with connected-objects
CO-3: Develop prototype IoT application using python

UNIT-I:
Introduction to Internet of Things: Definition and Characteristics of IoT, Physical Design of IoT – IoT Architecture, Smart Objects, Bits and Atoms, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems.

UNIT-II:
IoT Standards and Protocols: Infrastructure (ex: 6LowPAN, IPv4/IPv6, RPL); Identification (ex: EPC, uCode, IPv6, URIs); Comms / Transport (ex: Wifi, Bluetooth, LoRa); Discovery (ex: Physical Web, mDNS, DNS-SD); Data Protocols (ex: MQTT, CoAP, AMQP, Websocket, Node); Device Management (ex: TR-069, OMA-DM); Semantic (ex: JSON-LD, Web Thing Model); Multi-layer Frameworks (ex: Alljoyn, IoTivity, Weave, Homekit).

UNIT-III:
Introduction to Python: Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, data/time operations, classes, Exception handling Python packages - JSON, XML, HTTPLib, URLLib, SMTPLib

UNIT-IV:
IoT Physical Devices and Endpoints: Introduction to Raspberry PI-Interfaces (serial, SPI, I2C) Programming – Python program with Raspberry PI with focus of interfacing external gadgets, controlling output, and reading input from pins.

UNIT-V:
IoT Platforms: Introduction to IoT Platforms (AWS IoT, IBM Watson, ARM Mbed), Cloud Storage models and communication APIs, Python web application framework Designing a RESTful web API.

UNIT-VI:
Iot Applications and Issues: Combination scenarios, Breaking assumptions: - Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle with Case Studies

TEXT BOOKS:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547
2. Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, Dr. Ovidiu Vermesan, Dr. Peter Friess, River Publishers, 2007
3. Building the Internet of Things. Sara Cordoba, WimerHazenberg, Menno Huisman. BIS Publishers. 2011

REFERENCES:

1. Designing the Internet of Things, Adrian Mcewen, Hakin Cassimally, 2015
2. The Internet of Things: Key Applications and Protocols, Olivier Hersent, David Boswarthick, Omar Elloumi, 2012
3. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759
4. Securing the Internet of Things: A Standardization Perspective, Keoh, Sye Loong, Sahoo Subhendu Kumar, and Hannes Tschofenig, Internet of Things Journal, IEEE 1.3 (2014): 265-275

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1CP03) ETHICAL HACKING AND COMPUTER FORENSICS

COURSE PRE-REQUISITES: None

COURSE OBJECTIVES:
- To assess and measure threats to information assets
- To evaluate where information networks are most vulnerable
- To describe the origin of computer forensics and the relationship between law enforcement and industry
- To compose electronic evidence and the computing investigation process

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify and analyze the stages an ethical hacker requires to take in order to compromise a target system
CO-2: Implement security techniques and tools used to protect system and user data
CO-3: Identify, analyze, and mitigate threats to internal computer systems
CO-4: Produce various forensic tools to collect digital evidence

UNIT-I:
Introduction to Hacking: Basics of Hacking Techniques, Ethics of Hacking, Hacking techniques, Information War, Introduction to Ethical Hacking.
Password Cracking: Introduction, Password Stealing, Password Crackers

UNIT-II:
Sniffers: Introduction to Sniffers, Working of a Sniffer, Sniffer Programs, Detecting a Sniffer, Protecting Against a Sniffer.
Buffer Overflows: Introduction, Types of Buffer Overflow, Methods to Cause a Buffer Overflow,
Buffer Overflows: Detection and Prevention

UNIT-III:
Denial-of-Service Attacks: Denial-of-Service Attacks, Flood Attacks, Software Attacks, Distributed Denial-of-Service, Prevention of DoS Attacks.
Scanning Tools: Introduction, Scanners.

UNIT-IV:
Introduction to Cyber Crime: Introduction and Overview of Cyber Crime, Nature and Scope of Cyber Crime, Types of Cyber Crime: Social Engineering, Categories of Cyber Crime, Property Cyber Crime

UNIT-V:
Introduction to Computer Forensics & Investigations: Computer Forensics & Investigations as a profession, understanding computer investigations, data acquisition, processing crime and incident scenes, Network forensics, cell phone and mobile device forensics.

UNIT-VI:
Searching and Seizing Computer Related Evidence; Processing Evidence and Report Preparation
Current Computer Forensics Tools: Evaluating Computer Forensics Tool Needs, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software.

TEXT BOOKS:

1. Hacking Tools & Technique for Incident Handling, NIIT, PHI Learning
2. Computer Forensics and Investigations, Nelson, Phillips Enfinger, Steuart, Cengage Learning

REFERENCES:

1. Computer Forensics and Cyber Crime: An Introduction, Marjie T. Britz, Pearson
2. Ethical Hacking and Network Defense, Michael T. Simpson, Cengage Learning

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE04) FORMAL METHODS IN SOFTWARE ENGINEERING

COURSE OBJECTIVES:
- To develop an appreciation of the strengths of formal methods for engineering secure Software systems
- To build up a solid background for the application of formal methods to various tasks of the Software development process using Z
- To gain a basic level of competence in using formal methods to model Software systems and Verify Software system properties
- To construct formal methods of Software engineering through Z

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Explore the role of formal methods in Software engineering and glimpse of Z
CO-2: Produce a design and specification for an application using Z
CO-3: Critically evaluate features of Z and build methods in construction of Software systems
CO-4: Apply and design formal methods to system applications through Z

UNIT-I:
Foundations of Z: Understanding formal methods – motivation for formal methods – informal requirements to Formal specifications – validating formal specifications – Overview of Z specification – basic Elements of Z – sets and types – declarations – variables – expressions – operators – Predicates and equations.

UNIT-II:
Structures in Z: Tuples and records – relations, tables, databases – pairs and binary relations – functions –Sequences – propositional logic in Z – predicate logic in Z – Z and Boolean types – set Comprehension – lambda calculus in Z – simple formal specifications – modeling systems and Change.

UNIT-III:
Z Schemas and Schema Calculus: Z schemas – schema calculus – schema conjunction and disjunction – other schema calculus Operators – schema types and bindings – generic definitions – free types – formal reasoning Checking specifications – precondition calculation – machine-checked proofs.

UNIT-IV:
Z Case Studies: Case Study: Text processing system – Case Study: Eight Queens – Case Study: Graphical User Interface – Case Study: Safety critical protection system – Case Study: Concurrency and Real time systems.

UNIT-V:
Z Refinement: Refinement of Z specification – generalizing refinements – refinement strategies – program Derivation and verification – refinement calculus – data structures – state schemas – functions and relations – operation schemas – schema expressions – refinement case study.

UNIT-VI:
Formal Semantics and Tools: Operational Semantics – Denotation Semantics – Axiomatic Semantics Proof Editors – Proof Analyser – Symbolic Simulators –Translators – Test Generation Tools.

TEXT BOOKS:

1. The Way of Z: Practical Programming with Formal Methods, Jonathan Jacky, Cambridge University Press, 1996
2. Z: An introduction to Formal Methods, Antony Diller, Second Edition, Wiley, 1994

REFERENCES:

1. Using Z – Specification, Refinement, and Proof, Jim Woodcock and Jim Davies, Prentice Hall, 1996
2. The Z Notation: A Reference Manual, J. M. Spivey, Second Edition, Prentice Hall, 1992
3. Mathematical Logic for Computer Science, M. Ben-Ari, Second Edition, Springer, 2003
4. Logic in Computer Science – Modelling and Reasoning about Systems, M. Huth and M. Ryan, Second Edition, Cambridge University Press, 2004

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	0	**3**	**1.5**

(18PC2CP01) ADVANCED DATA STRUCTURES LABORATORY

COURSE PRE-REQUISITES: None

COURSE OBJECTIVES:
- To explain the data structures and their categories
- To understand abstract data types and differentiate linear and non-linear data structures
- To advance understanding of stack, queue and their applications
- To understand searching and sorting techniques in real-world scenarios

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Understanding and applying the Techniques in Software Development Life cycle
CO-2: Apply fundamental knowledge of Data Structures in Real time applications
CO-3: Develop a project based on Algorithms and Data Structures
CO-4: Analyze and implement graphs and trees for real time applications

LIST OF PROGRAMS:

1. Write a program to implement Hash Tables with Double Hashing
2. Write a program to implement all the functions of a dictionary (ADT) using Hashing
3. Write a program to implement Hash Tables with Linear Probing
4. Write a program to implement skip list search
5. Write a program to insert a record into the skip list
6. Write a program to implement BST(binary search tree) operations
7. Write a program to implement Red black tree operations
8. Write a program to implement AVL tree operations
9. Write a program to implement B-trees operations
10. Write a program for the Boyer-Moore Algorithm
11. Write a program for the Knuth-Morris-Pratt Algorithm
12. Write a program for the Huffman Coding Algorithm
13. Write a program to implement basic Insert, Search, Delete operations on K-D Tree
14. Write a program to implement Insert, Search operations on QuadTree
15. Write a program to implement One Dimensional Range searching
16. Write a program to implement Insert, Search, Delete operations on priority Search Tree

TEXT BOOKS:
1. Data Structures and Algorithms in Java, A. Drozdek, 3rd Edition, Cengage Learning
2. Data Structures with Java, J. R. Hubbard, 2nd Edition, Schaum"s Outlines, TMH
3. Data Structures and algorithms in Java, R. Lafore, 2nd Edition, Pearson Education

REFERENCES:
1. Data Structures using Java, D. S. Malik and P. S. Nair, Cengage Learning
2. Data structures, Algorithms and Applications in Java, S. Sahani, 2nd Edition, Universities Press
3. Design and Analysis of Algorithms, P. H. Dave and H. B. Dave, Pearson Education
4. Data Structures and Java Collections Frame Work, W. J. Collins, McGraw Hill
5. Java: The Complete Reference, Herbert Schildt, 7th Editon, TMH

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	0	**3**	**1.5**

18PC2SE01) DATA ANALYTICS LABORATORY

COURSE OBJECTIVES:
- To explore various stages of data analytics life cycle and Tools used in data analytics
- To understand the programming in R
- To use various data analysis models like Hypothesis test, regression modelling
- To analyze the big data analytics using Hadoop Map-Reduce and Pig

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Understand the importance of data analytics in real life through life cycle and explore the features of R and R Studio environment
CO-2: Explore the data types and programming constructs of R with examples
CO-3: Analyze the data using data models construction and using data mining techniques
CO-4: Use Big Data Analytics using Hadoop framework

LIST OF EXPERIMENTS:

1. Data Analytics Life Cycle
2. Basic Data Analytic methods using R and R Studio environment. Explore the features
3. Explore the data types of R and demonstrate the basic operations of data types
4. Explore the control structures of R and demonstrate with one example under each case
5. Importing & exporting the data from i) CSV file ii) Excel File
6. Data Visualization through i) Histogram ii) Pie Chart iii) Box Plot iv) Density Plots
7. Conduct Hypothesis Test using „mtcars" dataset
8. Demonstrate regression analysis
9. Demonstrate "Association Rule Mining" using "groceries" dataset
10. Demonstrate clustering technique using „iris" dataset
11. Word Count using Hadoop Map-Reduce Framework
12. Usage of various Pig commands
13. Word Count using Pig

TEXT BOOKS:
1. R-The Statistical Programming Language, Mark Gardner, Wiley India Pvt. Ltd.
2. Hadoop :The Definitive Gude, Tom White, Third Edition, O'Reilly, 2012

REFERENCES:
1. R Programming, A. K. Verma, Cengage
2. Big Data Glossary, Pete Warden, O'Reilly, 2011

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)	**L**	**T/P**	**C**
	0	**4**	**2**

(18PW4SE01) TECHNICAL SEMINAR

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify a research topic related to advanced/state-of-the-art technologies
CO-2: Collect the literature and comprehend/analyze critically the technological advancements
CO-3: Engage in effective oral communication through presentation of seminar
CO-4: Engage in effective written communication through report

COURSE OUTLINE:

- A student shall present a seminar on a technical topic during I semester of the M.Tech. programme.
- A student, under the supervision of a faculty member, shall collect literature on a technical topic of his / her choice, critically review the literature and submit it to the Seminar Review Committee (SRC) in a prescribed report form.
- The SRC shall consist of Head of the Department, faculty supervisor and a senior faculty member of the specialization / department.
- Student shall make an oral presentation before the SRC after clearing the plagiarism check.
- Prior to the submission of seminar report to the SRC, its soft copy shall be submitted to the PG Coordinator for PLAGIARISM check.
- The report shall be accepted for submission to the SRC only upon meeting the prescribed similarity index.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. I Semester (SE)

L	T/P	C
2	0	0

(18AU5CS01) RESEARCH METHODOLOGY AND IPR

COURSE PRE-REQUISITES: None

COURSE OBJECTIVES:
- To introduce the characteristics of a good research problem
- To choose appropriate approaches of investigation of solutions for research problem
- To familiarize with basic Intellectual Property Rights
- To understand different Patent Rights

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Understand research problem formulation, analyze research related information and follow research ethics
CO-2: Realize the importance of ideas, concept, and creativity in the present-day context
CO-3: Recognize that when IPR would take such important place in growth of individuals and nation, it is needless to emphasize the need of information about IPR to be promoted among students in general and engineering in particular
CO-4: Appreciate IPR protection which leads to creation of new and better products, and in turn brings about, economic growth and social benefits

UNIT-I:
Introduction: Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem.
Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, necessary instrumentations.

UNIT-II:
Literature Survey: Effective literature studies approaches, analysis. Plagiarism, Research ethics.

UNIT-III:
Effective Technical Writing: How to write report, Paper. Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee

UNIT-IV:
Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of
Patenting and Development: technological research, innovation, patenting, development.
International Scenario: International cooperation on Intellectual Property. Procedure for grants of patents, Patenting under PCT.

UNIT-V:
Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications.

UNIT-VI:
New Developments in IPR: Administration of Patent System. New developments in IPR; IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR.

TEXT BOOKS:
1. Research Methodology: An Introduction for Science & Engineering Students, Stuart Melville and Wayne Goddard

2. Research Methodology: An Introduction, Wayne Goddard and Stuart Melville
3. Resisting Intellectual Property, Halbert, Taylor & Francis Ltd ,2007

REFERENCES:

1. Research Methodology: A Step-by-Step Guide for Beginners, Ranjit Kumar, 2nd Edition
2. Research Methodology: Methods and Techniques, C. R. Kothari and Gaurav Garg, New Age International Publishers
3. Intellectual Property in New Technological Age, Robert P. Merges, Peter S. Menell, Mark A. Lemley, 2016
4. Intellectual Property Rights Under WTO, T. Ramappa, S. Chand, 2008

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
3	0	3

(18PC1CP04) ADVANCED ALGORITHMS

COURSE PRE-REQUISITES: UG level course in Algorithm Design and Analysis

COURSE OBJECTIVE:

- To introduce students to the advanced methods of designing and analyzing algorithms
- To the student should be able to choose appropriate algorithms and use it for a specific problem
- To familiarize students with basic paradigms and data structures used to solve advanced algorithmic problems
- To students should be able to understand different classes of problems concerning their computation difficulties

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Analyze the complexity/performance of different algorithms
CO-2: Determine the appropriate data structure for solving a particular set of problems
CO-3: Categorize the different problems in various classes according to their complexity
CO-4: Have an insight of recent activities in the field of the advanced data structure

UNIT-I:
Sorting: Review of various sorting algorithms, topological sorting
Graph: Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkasra's), depth-first search and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis.

UNIT-II:
Matroids: Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST.
Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.

UNIT-III:
Flow-Networks: Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.
Matrix Computations: Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition.

UNIT-IV:
Shortest Path in Graphs: Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming.
Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base-representation and modulo-representation. Polynomials addition and Multiplication.

UNIT-V:
Linear Programming: Geometry of the feasibility region and Simplex algorithm
NP-Completeness: P and NP Examples, Non-Deterministic algorithm for search, satisfiability, proof of NP-hardness and NP-completeness.

UNIT-VI:
Recent Trends in problem solving paradigms using recent searching and sorting techniques by applying recently proposed data structures.

TEXT BOOKS:
1. Introduction to Algorithms, Cormen, Leiserson, Rivest, Stein
2. Fundamentals of Computer Algorithms, E. Horrowitz, S. salmi, S. Rajasekaran, Second Edition, University Press, 2007
3. Algorithm Design – Foundations, Analysis, and Internet Algorithms, M. T. Goodrich, R. Tomassia, John Wiley, 2002

REFERENCES:
1. The Design and Analysis of Computer Algorithms, Aho, Hopcroft, Ullman
2. Algorithm Design, Kleinberg and Tardos
3. Design Analysis and Algorithms, Hari Mohan Pandy, University Science Press, 2009

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
3	0	3

(18PC1CP05) SOFT COMPUTING

COURSE PRE-REQUISITES: Basic knowledge of Mathematics

COURSE OBJECTIVES:

- To introduce soft computing concepts and techniques and foster their abilities in designing appropriate technique for a given scenario
- To implement soft computing based solutions for real-world problems
- To give students knowledge of non-traditional technologies and fundamentals of artificial neural networks, fuzzy sets, fuzzy logic, genetic algorithms
- To provide students hand-on experience on MATLAB to implement various strategies

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify and describe soft computing techniques and their roles in building intelligent machines
CO-2: Analyze fuzzy logic and reasoning to handle uncertainty and solve various engineering problems
CO-3: Apply neural networks and genetic algorithms to combinatorial optimization problems
CO-4: Evaluate and compare solutions by various soft computing approaches for a given problem

UNIT-I:
Introduction to Soft Computing: Evolution of Computing: Soft Computing Constituents, From Conventional AI to Computational Intelligence: Machine Learning Basics.

UNIT-II:
Fuzzy Logic: Fuzzy Sets, Operations on Fuzzy Sets, Fuzzy Relations, Membership Functions: Fuzzy Rules and Fuzzy Reasoning, Fuzzy Inference Systems, Fuzzy Expert Systems, Fuzzy Decision Making.

UNIT-III:
Neural Networks: Machine Learning Using Neural Network, Adaptive Networks, Feed forward Networks, Supervised Learning Neural Networks,
Radial Basis Function Networks: Reinforcement Learning, Unsupervised Learning Neural Networks, Adaptive Resonance architectures, Advances in Neural networks.

UNIT-IV:
Genetic Algorithms: Introduction to Genetic Algorithms (GA), Applications of GA in Machine Learning: Machine Learning Approach to Knowledge Acquisition.

UNIT-V:
Matlab/Python Lib: Introduction to Matlab /Python, Arrays and array operations, Functions and Files, Study of neural network toolbox and fuzzy logic toolbox, Simple implementation of Artificial Neural Network and Fuzzy Logic.

UNIT-VI:
Recent Trends in deep learning, various classifiers, neural networks and genetic algorithm Implementation of recently proposed soft computing techniques.

TEXT BOOKS:

1. Neuro-Fuzzy and Soft Computing, J. S. R. Jang, C. T. Sun and E. Mizutani, PHI / Pearson Education, 2004
2. Principles of Soft Computing, S. N. Sivanandam and S. N. Deepa, Wiley India Pvt. Ltd., 2011

REFERENCES:

1. Fuzzy Sets and Fuzzy Logic: Theory and Applications, George J. Klir and Bo Yuan, Prentice Hall, 1995
2. An Introduction to Genetic Algorithm, Melanic Mitchell, MIT Press, 1996
3. Fuzzy Logic with Engineering Applications, Timothy J. Ross, Wiley, 2010
4. Neural Networks, Fuzzy Logic and Genetic Algorithms, S. Rajasekaran and G. A. V. Pai, 1st Ed., PHI, 2003

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PC1SE02) SOFTWARE QUALITY ASSURANCE AND TESTING

COURSE OBJECTIVES:
- To define quality and understand various quality standards
- To recognize metrics, measurements and methodology for assessing software quality
- To understand steps involved in testing process and identify the appropriate testing technique
- To state steps involved in testing process and testing specialized systems

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify and analyze the importance of Software Quality Assurance process and Standards
CO-2: Demonstrate Quality Metrics and it's methodology
CO-3: Illustrate test documentation policies and compare different testing techniques
CO-4: Outline testing process of specialized systems

UNIT-I:
Software Quality Assurance Framework and Standards SQA Framework: What is Quality? Software Quality Assurance, Components of Software Quality Assurance - Software Quality Assurance Plan: Steps to develop and implement a Software Quality Assurance Plan - Quality Standards: ISO 9000 and Companion ISO Standards, CMM, CMMI, PCMM, 6 Sigma.

UNIT-II:
Software Quality Assurance Metrics and Measurement Software Quality Metrics: Product Quality metrics, In-Process Quality Metrics, Metrics for Software Maintenance, Software Quality metrics methodology: Establish quality requirements, Identify Software quality metrics, Implement the Software quality metrics, analyze Software metrics results, validate the Software quality metrics - Software quality indicators

UNIT-III:
Software Testing Strategy and Environment: Establishing testing policy, structured approach to testing, test factors, Economics of System Development Life Cycle (SDLC) Testing Software Testing Methodology
Defects hard to find, verification and validation, functional and structural testing, workbench concept, eight considerations in developing testing methodologies, testing tactics checklist.

UNIT-IV:
Software Testing Techniques: Black-Box, Boundary value, Bottom-up, Branch coverage, Cause-Effect graphing, CRUD, Database, Exception, Gray-Box, Histograms, Inspections, JADs, Pareto Analysis, Prototyping, Random Testing, Risk-based Testing, Regression Testing, Structured Walkthroughs, Thread Testing, Performance Testing, White-Box Testing

UNIT-V:
Software Testing Tools: Taxonomy of Testing tools. Methodology to evaluate automated testing tools, Load Runner, QTP and Rational Testing Tools, Silk test, Java Testing Tools, Jmetre.

UNIT-VI:
Testing Process:
Eleven Step Testing Process: Overview
Testing Specialized Systems and Applications
Testing Client/Server Web applications. Testing off the Shelf Components, Testing Security, Testing a Data Warehouse

TEXT BOOKS:

1. Software Testing and Continuous Quality Improvement, William E. Lewis, Gunasekaran Vcerapillai, Second Edition, Auerbach Publications
2. Effective Methods for Software Testing, William E. Perry, Second Edition, Wiley India, 2006
3. Metrics and Models for Software Quality Engineering, Stephen Kan, Pearson Education

REFERENCES:

1. Software Testing Tools, K. V. K. K. Prasad, Dreamtech, 2008
2. Testing and Quality Assurance for Component-based Software, Gao Tsao and Wu, Artech House Publishers
3. Software Testing Techniques, Bories Beizer, Second Edition, Dreamtech
4. Managing the Testing Process, Rex Black, Wiley
5. Handbook of Software Quality Assurance, G. Gordon Schulmeyer, James McManus, Second Edition, International Thomson Computer Press

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
3	0	3

(18PE1SE05) ADVANCED SOFTWARE ENGINEERING PRINCIPLES AND PRACTICES

COURSE OBJECTIVES:

- To understand the importance of software engineering principles in software development
- To develop the necessary skills to handle software projects in a principled way
- To understand the basics of testing, debugging and errors
- To understand the best practices in software engineering

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Apply Software engineering principles to major projects
CO-2: Implement design principles to design a real time project
CO-3: Develop a reusable Software component in a real time project
CO-4: Evaluate a real time project using Software engineering principles

UNIT-I:
System Engineering: System Engineering: Computer based systems, system engineering hierarchy, Information engineering, Information strategy planning, business area analysis, product engineering, modelling the system architecture, system modelling and simulation, system specification. Computer Based System Engineering: Emergent system properties, systems and their environment, system modelling, system engineering process, system procurement.

UNIT-II:
Modern Design Concept and Principles: Design Concepts: Mapping of analysis model to design model, design process, design principles, design concepts, effective modular design, design model, design heuristics, design specification.

UNIT-III:
Architectural Design Process: Transform mapping and transaction mapping, design post processing, interface design, Human computer interface design, and interface design guidelines, procedural design.

UNIT-IV:
Real Time Software Design: Real–time systems, definition, System consideration, Real time system analysis, stimulation / Response systems, Real–time System model, system elements, Real – time programming, system design, Real-time system modelling, RTOS, process priority, process management, scheduling strategy, RT-Systems, design process, monitoring and control system, Generic architecture, data acquisitions systems

UNIT-V:
User Interface Design: The golden rules, user interface analysis and design, interface analysis, interface design steps, WebApp interface design, Design evaluation.

UNIT-VI:
Component Based Development: CBSE: Component based Software engineering, components and component models;
Component based Software engineering process, Component Composition.
Software Reuse: Management issues, Reuse process, domain engineering, Building Reusable Components classification and retrieving components.

TEXT BOOKS:

1. Software Engineering – A Practitioner's Approach, Roger S. Pressman, 4th Edition, McGraw Hill Publications
2. Software Engineering, Ian Sommerville, 6th / 7th Edition, Pearson Education
3. Software Engineering Theory and Practices, Shari Lawarence, 2nd Edition, Pfleeger

REFERENCES:

1. System Analysis and Design in Changing World, John W. Satzinger, Robert B. Jackson, Stephen D. Burd, Thomson Course Technology
2. Intelligent Software Agents, Richard Murch, Tony Johnson, Prentice Hall

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE06) WEB SERVICES AND SERVICE ORIENTED ARCHITECTURE

COURSE OBJECTIVES:

- To define and design applications as combinations of services, and be able to discuss the emergent properties of those compositions
- To understand concepts, technology and design of service orientation and web services analyzing and designing business based on SOA principles
- To understand web service specifications and assess support by platforms like J2EE and .NET

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Define distributed computing evolution and web service basics
CO-2: Understand SOA fundamentals and build Services with WS-* Specifications
CO-3: Assess the service layer abstraction & analyze and design the building an SOA
CO-4: Examine SOA support provided by J2EE and .NET platform

UNIT-I:
Evolution of Distributed Computing: Core distributed computing technologies-client/server, CORBA, JAVA RMI, Microsoft DCOM, MOM, Challenges in Distributed Computing Introduction to Web Services: The definition of web services, basic operational model of webservices, Core Web Service Standards, benefits and challenges of using web services

UNIT-II:
SOA and Web Services Fundamentals:
Introducing SOA: Fundamental SOA, Common Characteristics of Contemporary SOA, Common tangible benefits of SOA, Common pitfalls of adopting SOA.
The Evolution of SOA: An SOA timeline, The continuing evolution of SOA, The roots of SOA.
Web Services and primitive SOA: The Web Services frame work, Services, Service Descriptions, and Messaging.

UNIT-III:
SOA and WS-* Extensions:
Web Services and Contemporary SOA (Part-I): Message Exchange Patterns, Service Activity Coordination, Atomic transactions, Business Activities, Orchestration, and Choreography.
Web Services and Contemporary SOA (Part-II): Addressing, Reliable Messaging, Correlation, Policies, Metadata Exchange, Security, Notification and Eventing.

UNIT-IV:
SOA and Services – Orientation: Principles of Service-Orientation: Service - Orientation and the Enterprise, Anatomy of SOA, Common Principles of Service – Orientation, Interrelation between Principles of Service-Orientation, Service Orientation and Object Orientation, Native Web Services support for Principles of Service-Orientation. Service Layers: Service-Orientation and contemporary SOA, Service Layer abstraction, Application Service Layer , Business Service Layer, Orchestration Service Layer, Agnostic Services, Service Layer Configuration Scenarios.

UNIT-V:
Building SOA (Planning and Analysis)
SOA Delivery Strategies: SOA delivery lifecycle phases, the top-down strategy, the bottom-upstrategy, the agile strategy.
Service-Oriented Analysis (Part I-Introduction): Introduction to Service Oriented Analysis, Benefits of a Business-Centric SOA, Deriving Business Services.

Service Oriented Analysis (Part-II-Service Modelling): Service Modelling, Service Modelling guidelines, Classifying Service model logic, Contrasting Service modelling Approaches.

UNIT-VI:
Building SOA (Technology and Design)
Service Oriented Design (Part I-Introduction): Introduction to Service-Oriented Design, WSDL related XML Schema language basics. WSDL language basics, SOAP language basics, Service interface design tools.
Service Oriented Design (Part II-SOA Composition Guidelines): SOA Composing steps, Considerations for choosing service layers, Considerations for positioning core SOA standards, Considerations for choosing SOA extensions.
Service Oriented Design (Part III - Service Design): Service Design overview, Entity-centric business Service Design, Application Service Design, Task-centric business Service Design, Service Design guidelines.
Service Oriented Design (Part IV-Business Process Design): WS-BPEL language basics, WS-Coordination overview, Service Oriented Business process Design.
Fundamental WS-* Extensions: WS-Addressing language basics, WS-Reliable Messaging language basics, WS-Policy language basics, WS-Metadata Exchange language basics, WS-Security language basics.
SOA Platforms: SOA platform basics, SOA support in J2EE and .NET, Integration considerations.

TEXT BOOKS:
1. Service-Oriented Architecture: Concepts, Technology & Design, Thomas Erl, Pearson Edu
2. Developing Java Web Services, R. Nagappan, R. Skoczylas, R. P. Sriganesh, Wiley India

REFERENCES:
1. Web Services & SOA Principles and Technology, Michael P. Papazoglou, Second Edition
2. The Definitive Guide to SOA, Jeff Davies & Others, Apress, Dreamtech
3. Understanding SOA with Web Services, Eric Newcomer, Greg Lomowand Pearson Education
4. Java SOA Cook Book, E. Hewitt, SPD
5. SOA in Practice, N. M. Josuttis, SPD

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
3	0	3

(18PE1SE07) SCRIPTING LANGUAGES

COURSE OBJECTIVES

- To appreciate the nature of scripting and the role of scripting languages
- To design and implement Perl and Python software solutions that accommodate specified requirements and constraints
- To evaluate modern, representative programming languages critically
- To design and implement PHP and PYTHON software solutions that accommodate specified requirements and constraints, based on analysis or modeling or requirements specification

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Explain the differences between typical scripting languages and typical system and application programming languages
CO-2: Apply the syntax and semantics of languages such as PHP and PYTHON for effective scripting
CO-3: Create software systems using scripting languages, including Perl and Python
CO-4: Design and implement the appropriate software solutions using Scripting Languages

UNIT-I:
Scripts & Programs: Origin of Scripting, Characteristics of scripting languages. PHP Basics-Features, Embedding PHP Code in Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries.

UNIT-II:
PHP Arrays: Creating an Array, Adding and Removing Array Elements, Locating Array Elements, Traversing Arrays, Determining Array Size and Uniqueness, Sorting Arrays. Strings and Regular Expressions.

UNIT-III:
PHP and Web Forms: Files, PHP authentication methodologies- Hard-coded, File-Based, Database based and IP based. Uploading Files with PHP, Sending Email using PHP, PHP's Encryption Functions, the MCrypt package.

UNIT-IV:
Introducing MySQL: What Makes MySQL so popular, The Evaluation of MySQL, Installing and Configuring MySQL, MySQL Storage Engines and Data types: Storage Engines, Data types and Attributes, Working with Databases and Tables, Securing MySQL.

UNIT-V:
PHP with MySQL: Handling Installation Prerequisites, Using the mysqli Extension, Interacting with the Database, Executing Database Transactions. MySQL Triggers, MySQL Views, Indexes and Searching: Database Indexing, Form – Based Searches.

UNIT-VI:
Introduction to Python Language: Python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling, Web Application Framework, Applications of Python programming.

TEXT BOOKS:

1. Beginning PHP and MySQL, Jason Gilmore, 3rd Edition, Dreamtech
2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications

REFERENCES:

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J. Lee and B. Ware, Pearson Education (Addison Wesley)
2. Programming Python, M. Lutz, SPD
3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publication
4. Core Python Programming, Chun, Pearson Education

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
3	0	3

(18PE1SE08) SOFTWARE REENGINEERING

COURSE OBJECTIVES:
- To understand and analyze the importance and basic concepts of Software reengineering
- To identify the standards in Software reengineering
- To describe the concepts of organizing and reorganizing process
- To analyze the need of Software Reuse Tools

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Illustrate the importance of Software reengineering
CO-2: Outline the standards in Software reengineering
CO-3: Use the concepts of organizing and reorganizing process in present industry
CO-4: Compare and Contrast various Software Reuse Tools

UNIT-I:
Introduction: The Need, What is reuse, Types of reuse, Domain Analysis, Hypertext
Software Lifecycle: Requirements, Design, Implementation, Testing and Documentation, Maintenance, Standards

UNIT-II:
Management: Software Team Organization, Process Modelling, Software Project Modelling, Scheduling
Reengineering Concepts: Reengineering Objectives, General Models of Software Reengineering,

UNIT-III:
Software Reliability Methods, Dealing with Software reliability, Reengineering Phases and Tasks, Reengineering Implementation.
Standards: Expectations, Existing Related Standards, Recommendations, Conclusion.

UNIT-IV:
Enterprise & Standards:
Reuse Framework: Process Idioms and Sources, Reuse management, Asset creation, Asset management, Asset Utilization, Cost Vs Benefits, legal Issues.

UNIT-V:
Organize to Reorganize:
Organizing: Indexing, Document Outlines, Domain Models, Code Organizing, Framework.
Retrieving: Retrieval specification, Document Retrieval, Program Retrieval, Reorganizing: Retrieved Component suitability, Program reorganizing, Code generators, Testing after Reuse.

UNIT-VI:
Practical Examples:
Software Reuse tools: CASE, Practitioner, MUCH, Softclass, User Interface Generator.
Case Studies: IBM Reuse and Boblingen Experience, HP Reuse, Motorola Reuse.

TEXT BOOKS:
1. Reengineering Software–How to Reuse Programming to Build New, State-of-the-Art Software, Roy Rada, 2nd Edition, Eric Dobby Publishing

2. Application Software Reengineering, Afshar Alam, Tendai Padenga, Pearson Education India, 2010

REFERENCES:

1. Secrets of Reverse Engineering, Eldad Eilal, Wiley Publishing
2. Advancement in Software Maintenance Management: Technologies and Solutions, Macario Polo, Ed. Idea Group Publishing, 2003
3. Software Reengineering, Robert S. Arnold, IEEE Comp. Society

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
3	0	3

(18PE1CP04) CLOUD COMPUTING

COURSE OBJECTIVES:

- To understand cloud computing paradigm, recognize its various forms
- To get a clear understanding of Cloud Computing fundamentals and its importance to various organizations
- To master the concepts of IaaS, PaaS, SaaS, Public and Private clouds
- To understand AWS and learn to develop applications in AWS

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

CO-1: Articulate the main concepts, key technologies, strengths, and limitations of cloud computing

CO-2: Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, public cloud, private cloud, hybrid cloud, etc.

CO-3: Explain the core issues of cloud computing such as security, privacy, and interoperability

CO-4: Identify problems, and explain, analyze, and evaluate various cloud computing solutions

UNIT-I:

Systems Modelling, Clustering and Virtualization: Distributed System Models and Enabling Technologies, Computer Clusters for Scalable Parallel Computing, Virtual Machines and Virtualization of Clusters and Data centres.

UNIT-II:

Foundations: Introduction to Cloud Computing, Migrating into a Cloud, Enriching the 'Integration as a Service' Paradigm for the Cloud Era, the Enterprise Cloud Computing Paradigm.

UNIT-III:

Infrastructure as a Service (IAAS) & Platform and Software as a Service (PAAS / SAAS)Virtual machines provisioning and Migration services, On the Management of Virtual machines for Cloud Infrastructures, Enhancing Cloud Computing Environments using a cluster as a Service.

UNIT-IV:

Secure Distributed Data Storage in Cloud Computing. Aneka, Comet Cloud, T-Systems', Workflow Engine for Clouds, Understanding Scientific Applications for Cloud Environments. An Architecture for Federated Cloud Computing.

UNIT-V:

SLA Management in Cloud Computing, Performance Prediction for HPC on Clouds, Best Practices in Architecting Cloud Applications in the AWS cloud, Building Content Delivery networks using Clouds, Resource Cloud Mashups.

UNIT-VI:

Governance and Case Studies: Organizational Readiness and Change management in theCloud age, Data Security in the Cloud, Legal Issues in Cloud computing, Achieving Production Readiness for Cloud Services.

TEXT BOOKS:

1. Cloud Computing: Principles and Paradigms, Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012
3. Cloud Computing: A Practical Approach, Anthony T. Velte, Toby J. Velte, Robert Elsenpeter, Tata McGraw Hill, 2011

REFERENCES:

1. Cloud Computing: A Practical Approach, Anthony T. Velte, Toby J. Velte, Robert Elsen peter, Tata McGraw Hill, 2011
2. Enterprise Cloud Computing, Gautam Shroff, Cambridge University Press, 2010
3. Cloud Computing: Implementation, Management and Security, John W. Rittinghouse, James F. Ransome, CRC Press, 2012
4. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud, George Reese, O'Reilly, SPD, 2011
5. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, 2011

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE09) SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

COURSE OBJECTIVES:

- To outline the fundamentals of software architecture and quality requirements of a software system
- To define the fundamental principles and guidelines for software architecture design, architectural styles, patterns, and frameworks
- To identify the methods, techniques, and tools for software architecture and document different software product lines
- To relate various design patterns for software architecture system

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Construct and design suitable Software architecture for small software systems
CO-2: Analyze major Software architectural styles, design patterns, and frameworks
CO-3: Elaborate Software architecture using various documentation approaches and architectural description languages
CO-4: Demonstrate a number of the fundamental patterns and principles of software architectural styles and design any case study

UNIT-I:
Envisioning Architecture: The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, Reference architectures, architectural structures and views.
Creating an Architecture, Quality Attributes, Achieving qualities. Architectural styles and patterns, designing the Architecture, Documenting Software architectures, Reconstructing Software Architecture.

UNIT-II:
Analyzing Architectures: Architecture Evaluation, Architecture design decision making, ATAM, CBAM.

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

UNIT-IV:
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, facade, flyweight Proxy.

UNIT-V:
Behavioral Patterns: Chain of responsibility, command. Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

UNIT-VI:
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, Len Bass, Paul Clements & RickKa, Second Edition, Pearson Publication, 2003
2. Design Patterns, Erich Gamma, Pearson Education, 1995

REFERENCES:

1. Beyond Software Architecture, Luke I-lohmann, Addison Wesley, 2003
2. Software Architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall PTR, 2001
3. Pattern Oriented Software Architecture, F. Buschmann & Others, John Wiley & Sons
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

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
0	3	1.5

(18PC2CP03) ADVANCED ALGORITHMS LABORATORY

COURSE OBJECTIVES:

- To familiarize the advanced methods of designing and analyzing algorithms
- To illustrate appropriate algorithms and use it for a specific problem
- To train basic paradigms and data structures used to solve advanced algorithmic problems
- To understand different classes of problems concerning their computation difficulties

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Implement the complexity/performance of different algorithms
CO-2: Determine the appropriate data structure for solving a particular set of problems
CO-3: Categorize the different problems in various classes according to their complexity
CO-4: Obtain an insight of recent activities in the field of the advanced data structures and algorithms

LIST OF PROGRAMS:

1. Write a program for Topological sorting.
2. Write a program for Shortest path by BFS.
3. Write a program for Dijkasras's Algorithm.
4. Write a program for maximum matching using edmonds blossom algorithm.
5. Write a program to implement Maxflow-minicut theorem.
6. Write a program to implements Ford-Fulkerson algorithm to compute maximum flow.
7. Write a program to implement strassen's algorithm.
8. Write a program to implement LUP-decomposition of a matrix.
9. Write a program to implement Floyd-Warshall Algorithm.
10. Write a program to implement to perform dfs using fast Fourier transform.
11. Write a program to implement the Schonhage-Strassen algorithm for Multiplication of Two Numbers.
12. Write a program to implement Binomial Probability using randomized algorithms.
13. Write a program to compute Modular power using number theoretic algorithms.
14. Write a program to implement greedy approximation algorithm for Graph Coloring.

TEXT BOOKS:

1. Introduction to Algorithms, Cormen, Leiserson, Rivest, Stein
2. Fundamentals of Computer Algorithms, E. Horrowitz, S. salmi, S. Rajasekaran, Second Edition, University Press, 2007
3. Algorithm Design – Foundations, Analysis, and Internet Algorithms, M. T. Goodrich, R. Tomassia, John Wiley, 2002

REFERENCES:

1. The Design and Analysis of Computer Algorithms, Aho, Hopcroft, Ullman
2. Algorithm Design, Kleinberg and Tardos
3. Design of Analysis and Algorithms, Hari Mohan Pandy, University Science Press, 2009
4. A Practical Guide to Data Structures and Algorithms using Java, S. Goldman & K. Goldman, Chapman & Hall/CRC, Taylor & Francis Group. a. (Note: Use packages like java.io, java.util, etc.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)	**L**	**T/P**	**C**
	0	**3**	**1.5**

(18PC2SE02) SOFTWARE TESTING AND CASE TOOLS LABORATORY

COURSE OBJECTIVES:
- To define the process of test documentation
- To identify the usage of automated testing tools
- To understand different software testing tools and their features
- To apply test cases for a software project using different testing and tracking tools

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Generate various test documents
CO-2: Understand and Analyse different testing tools and their mechanisms
CO-3: Identify and apply appropriate automated testing tool
CO-4: Analyze different testing tools like Selenium, Unified Functional Testing and Bugzilla for web testing and bug tracking

LIST OF EXPERIMENTS:

1. Take any system (e.g. ATM system) and study its system specifications and report the various bugs
3. Write the test cases for any known application (e.g. Banking application)
4. Create a test plan document for any application (e.g. Library Management System)
5. Study of any testing tool (e.g. QTP/UFT)
6. Study of any web testing tool (e.g. Selenium)
7. Study of any bug tracking tool (e.g. Bugzilla)
8. Study of any test management tool (e.g. Quality Centre/ALM)
9. Take a mini project (e.g. University admission, Placement Portal) and execute it. During the Life cycle of the mini project create the various testing documents* and final test report document

TEXT BOOKS:
1. The Art of Software Testing, Glenford Myers et. el., 2nd Edition, 2004
2. Test Automation Using HP Unified Functional Testing - Explore Latest Version of Qtp, Garg, MR Navneesh

REFERENCES:
1. How to Break Software: A Practical Guide to Testing, James Whittaker, 2002
2. Effective Methods for Software Testing, William E. Perry, Third Edition, 2006
3. Software Automation Testing Tools for Beginners, Rahul Shedye
4. A Practitioner's Guide to Software Test Design, Lee Copeland, 2003

NOTE:
- The new version of HP QTP is being called HP Unified Functional Testing (UFT). UFT is actually a combination of HP QTP (for testing GUI) and HP Service Test (for testing API).
- HP Application Life Cycle Management (ALM) is the latest incarnation of flagship test management tool Quality Center (QC). These tutorials are based on HP ALM.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)	**L**	**T/P**	**C**
	0	**4**	**2**

(18PW4SE02) MINI-PROJECT

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Understand the formulated industry / technical / societal problems
CO-2: Analyze and / or develop models for providing solution to industry / technical / societal problems
CO-3: Interpret and arrive at conclusions from the project carried out
CO-4: Demonstrate effective communication skills through oral presentation
CO-5: Engage in effective written communication through project report

COURSE OUTLINE:

- A student shall undergo a mini-project during II semester of the M.Tech. programme.
- A student, under the supervision of a faculty member, shall collect literature on an allotted project topic of his / her choice, critically review the literature, carry out the mini-project, submit it to the department in a prescribed report form.
- Evaluation of the mini-project shall be done by a Project Review Committee (PRC) consisting of the Head of the Department, faculty supervisor and a senior faculty member of the specialization / department.
- Prior to the submission of mini-project report to the PRC, its soft copy shall be submitted to the PG Coordinator for PLAGIARISM check.
- The mini-project report shall be accepted for submission to the PRC only upon meeting the prescribed similarity index.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. II Semester (SE)

L	T/P	C
2	0	0

(18AU5EN01) ENGLISH FOR ACADEMIC AND RESEARCH WRITING

COURSE OBJECTIVES:

- To understand the usage of appropriate vocabulary (Formal, Informal, Gender Insensitive etc.)
- To understand the features and processes of academic writing
- To identify the resources
- To understand standard documentation styles

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

CO-1: Use appropriate vocabulary (Formal, Informal, Slang, Gender Insensitive etc.)

CO-2: Employ processes of academic writing

CO-3: Identify the resources

CO-4: Understand standard documentation styles

UNIT- I:

Introduction to Research:

i. Identifying the topic
ii. Identifying Sources; Finding Sources
iii. Defining the broad area; Defining the specific area; Difference between a broad area and specific area
iv. Choosing a topic
v. Mechanics of Writing – Language, Tone, Style, Ethics

UNIT-II:

Referencing & Library Skills:

i. Literature Survey
ii. Writing Objectives
iii. Hypothesis
iv. Methodology
v. Prospects for Future Research

UNIT-III:

Academic Writing Skills:

i. Paraphrasing
ii. Summarizing
iii. Quoting
iv. Rewriting
v. Expansion

UNIT-IV:

Kinds of Academic Writing:

i. Essays
ii. Reports
iii. Reviews
iv. SOPs
v. Abstracts
vi. Proposals

UNIT-V:

Research Process:

i. Selection of Topic
ii. Formulation of Hypothesis
iii. Collection of Data
iv. Analysis of Data
v. Interpretation of Data
vi. Presentation of Data

UNIT-VI:

i. Title
ii. Abstract
iii. Introduction
iv. Literature Survey
v. Methodology
vi. Discussion
vii. Findings/Results
viii. Conclusion
ix. Documenting Sources

TEXT BOOKS:

1. Writing for Science, Goldbort R., Yale University Press (available on Google Books), 2006
2. Handbook of Writing for the Mathematical Sciences, Highman N., SIAM. Highman's Book, 1998

REFERENCES:

1. How to Write and Publish a Scientific Paper, Day R., Cambridge University Press, 2006
2. English for Writing Research Papers, Adrian Wall Work, Springer New York Dordrecht Heidelberg London, 2011
3. MLA Handbook for Research

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)

L	T/P	C
3	0	3

(18PE1CP11) BLOCKCHAIN TECHNOLOGY

COURSE OBJECTIVES:
- To get the terminologies and overview of Blockchain technology
- To study the concepts and foundation of Blockchain Technology
- To understand security mechanism and Consensus in Blockchain
- To design use cases and architecture Blockchain Technology
- To study benefits, limitations and identify application area of Blockchain Technology

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Gain a clear understanding of the concepts that underlie digital distributed ledger
CO-2: Understand key mechanisms like decentralization, transparency and trust, Immutability
CO-3: Understand and apply the concept of Hash Function and Related Hash algorithms for high secure and availability of systems
CO-4: Learn how to design and implement any application in Blockchain Technology

UNIT-I:
Introduction to Blockchain Part I: Introduction to centralized decentralized and distributed system, History of Blockchain, Various technical definitions of Blockchain.
Introduction to Blockchain Technology Part II: Generic elements of a blockchain: Block, Transaction, Peer-to-peer network, Node, Smart contract, Why It's Called "Blockchain".

UNIT-II:
Concept of Blockchain Technology Part I: Cryptography, Hashing, Nonce, Distributed database, Consensus, Smart Contract, Component of block, Structure of Blockchain.
Concept of Blockchain Technology Part II: Applications of blockchain technology, Tiers of blockchain technology Bolckchain 0, Bolckchain 1, Bolckchain 2, Bolckchain 3, Generation of Blockchain X.

UNIT-III:
Cryptography and Technical Foundations: Cryptography, Confidentiality, Integrity, Authentication, Cryptographic primitives, Public and private keys, RSA, Discrete logarithm problem,
Hash Function: Message Digest (MD), Secure Hash Algorithms (SHAs), Design of Secure Hash Algorithms (SHA), SHA-256, Design of SHA3, Elliptic Curve Digital signature algorithm.

UNIT-IV:
Types of Blockchain: Public blockchains, Private blockchains, Semi-private blockchains, Sidechains, Permissioned ledger Distributed ledger Shared ledger Fully private and proprietary blockchains, Tokenized blockchains, Tokenless blockchains, CAP theorem and blockchain

UNIT-V:
Financial markets and trading, Trading, Exchanges, Trade life cycle, Order anticipators, Market manipulation.
Crypto Currency: Bitcoin, Bitcoin definition, Keys and addresses, Public keys in Bitcoin, Private keys in Bitcoin, Bitcoin currency units

UNIT-VI:
Implementation Platforms: Hyperledger as a protocol, Reference architecture, Hyperledger Fabric, Transaction Flow, Hyperledger Fabric Details, Fabric Membership, Fabric Membership

TEXT BOOK:

1. Mastering Blockchain, Imaran Bashir

REFERENCES:

1. Blockchain for Dummies®, IBM Limited Edition, John Wiley & Sons, Inc.

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE10) SOFTWARE DEFINED NETWORKS

COURSE OBJECTIVES:
- To provide a comprehensive introduction to Software Defined Networking (SDN)
- To explain the importance of network virtualization over legacy network
- To identify the impact of SDN on traffic management and the potential on service growth, and discuss few potential issues
- To provide an overview on OpenFlow protocol, and their packet format

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Understand the concepts of Software Defined Networking (SDN) and its network services and challenges
CO-2: Understand the business case and technology motivations for considering SDN solutions
CO-3: Learn how to identify the impact of SDN on traffic management and the potential for network service growth
CO-4: Learn the concepts of visualization, particularly the impact of visualization on servers and networks, service providers, legacy networks, and network vendors

UNIT-I:
Introduction to Software Defined Networking: Virtualization, Virtual Memory, Virtual Memory Operation, Virtual and Physical Memory Mapping, Server Virtualization, Storage Virtualization, Software Defined Networking, Network Limitations, Network Control Plane.

UNIT-II:
SDN Implementation: Introduction, SDN Implementation, SDN Design, Separation of the Control and Data Planes, Edge-Oriented Networking, SDN Operation, Service Providers and SDN

UNIT-III:
Openflow: Introduction, Overview of the OpenFlow Switch Specification, OpenFlow Ports, OpenFlow Packet-Processing Pipeline, OpenFlow Channel, Message Handling, OpenFlow Channel Connections, Controller Modes, Auxiliary Connection Use for Performance and Reliability, Flow Table Synchronization, Bundle Messages, OpenFlow Configuration-and-Management Protocol, Remote Configuration and The OpenFlow Conformance Testing Program

UNIT-IV:
SDN Controllers: Network Programmability, The Management Interface, The Application-Network Divide Modern Programmatic Interfaces, Virtualization and Data Plane I/O, Services Engineered Path, Service Locations and Chaining.

UNIT-V:
SDN Evolution: Introduction, SDN and Enterprise Networks, SDN and Transport Networks, SDN and Optical Transport Networks, Increasing WAN Utilization with SDN,SDN Scalability Issues, Controller Designs for Scalability, Potential SDN Scalability Issues, Network Types

UNIT-VI:
SDN Management: Load Adaptation, Google and SDN, Google's G-Scale Network, Google's G-Scale Network Hardware, Google SDN Deployment, Implementation Challenges

TEXT BOOKS:

1. Software Defined Networking: Design and Deployment, 1st Edition, Patricia A. Morreale and James M. Anderson, CRC Press
2. SDN: Software Defined Networks, Thomas D. Nadeau and Ken Gray, Orielly Media

REFERENCES:

1. Software Defined Networking with OpenFlow, Siamak Azodolmolky, Wiley Publications
2. Software Defined Networks: A Comprehensive Approach, Paul Goransson, Chuck Black, Morgan Kaufmann

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18PE1SE11) SOFTWARE METRICS

COURSE OBJECTIVES:
- To be aware of core metrics for product, quality, process
- To familiarize with the concepts of software quality and tools for quality metrics
- To learn more about software liability
- To understand important concepts of complexity metrics and OO metrics

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Be aware of connections of Software engineering measurements with the multidisciplinary history of measurement theory
CO-2: Be aware of the social and ethical issues associated with human performance measurement
CO-3: Assess the quality of a proposed metric
CO-4: Understand the commercial and organizational contexts of any metric

UNIT-I:
The History and Evolution of Software Metrics: Evolution of the Software industry and evolution of Software measurements – The cost of counting function point metrics – The paradox of reversed productivity for high-Level languages- The Varieties of functional metrics – Variations in application size and productivity rates – Future Technical Developments in Functional Metrics- Software measures and metrics not based on function points.

UNIT-II:
Measuring Software Quality: Quality control and international competition – Defining quality for measurement and estimation – Five steps to Software quality control- Measuring Software defect removal- Measuring Defect removal efficiency – Measuring the costs of defect removal – Evaluating defect prevention methods – Measuring customer reported defects- Measuring invalid defects.

UNIT-III:
Process Metrics: In-Process Metrics for Software Testing - Test Progress S Curve - Testing Defect Arrivals Over Time - Product Size Over Time - CPU Utilization - Effort/Outcome Model. Complexity Metrics and Models - Lines of Code - Halstead's Software Science - Cyclomatic Complexity. - Syntactic Constructs - Structure Metrics.

UNIT-IV:
Mechanics of Measurement: Software Assessments – Software Baselines – Software Benchmarks- What a Baseline analysis covers – Developing or Acquiring a baseline data collection Instrument – Administering the data collection questionnaire – Analysis and aggregation of the Baseline data.

UNIT-V:
Measurement and Data Assessment: Measuring and Analyzing Customer Satisfaction - Surveys - Data Collection - Sampling Methods - Analyzing Satisfaction Data. Conducting In-Process Quality Assessments - Preparation - Evaluation - Quantitative Data - Qualitative Data - Evaluation Criteria - Overall Assessment.

UNIT-VI:
Measurements, Metrics and Industry Leadership: Measures and metrics of industry leaders – Measures, metrics and innovation – Measurements, metrics and outsource litigation – Measurements, metrics and behavioral changes – Commercial Software measurement tools.

Measuring Process Maturity - Process Capability - Value of Process Improvement - Process Adoption – Process Compliance

TEXT BOOKS:

1. Applied Software Measurement: Global Analysis of Productivity and Quality, Caper Jones, Third Edition, McGraw-Hill Companies, 2008
2. Metrics and Models in Software Quality Engineering, Stephen H. Kan, Addison Wesley, 2011

REFERENCES:

1. Object-Oriented Software Metrics, Mark Lorenz, Jeff Kidd, Prentice Hall, 2000
2. Software Testing Principles and Practices, Naresh Chauhan, Oxford University Press, 2010

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18OE1CN01) BUSINESS ANALYTICS

COURSE OBJECTIVES:

- To understand the role of business analytics within an organization and to analyze data using statistical and data mining techniques and understand relationships between the underlying business processes of an organization
- To gain an understanding of how managers use business analytics to formulate and solve business problems and to support managerial decision making and to become familiar with processes needed to develop, report, and analyze business data
- To use decision-making tools/Operations research techniques and to manage business process using analytical and management tools
- To analyze and solve problems from different industries such as manufacturing, service, retail, software, banking and finance, sports, pharmaceutical, aerospace etc.

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Apply knowledge of data analytics
CO-2: Think critically in making decisions based on data and deep analytics
CO-3: Use technical skills in predicative and prescriptive modeling to support business decision-making
CO-4: Translate data into clear, actionable insights

UNIT-I:
Business Analytics: Overview of Business analytics, Scope of Business analytics, Business Analytics Process, Relationship of Business Analytics Process and organization, competitive advantages of Business Analytics.
Statistical Tools: Statistical Notation, Descriptive Statistical methods, Review of probability distribution and data modelling, sampling and estimation methods overview.

UNIT-II:
Trendiness and Regression Analysis: Modelling Relationships and Trends in Data, simple Linear Regression. Important Resources, Business Analytics Personnel, Data and models for Business analytics, problem solving, Visualizing and Exploring Data Business Analytics Technology.

UNIT-III:
Organization Structures of Business analytics, Team management, Management Issues, Designing Information Policy, Outsourcing, Ensuring Data Quality, Measuring contribution of Business analytics, Managing Changes.
Descriptive Analytics, predictive analytics, predicative Modelling, Predictive analytics analysis, Data Mining, Data Mining Methodologies, Prescriptive analytics and its step in the business analytics Process, Prescriptive Modelling, nonlinear Optimization.

UNIT-IV:
Forecasting Techniques: Qualitative and Judgmental Forecasting, Statistical Forecasting Models, Forecasting Models for Stationary Time Series, Forecasting Models for Time Series with a Linear Trend, Forecasting Time Series with Seasonality, Regression Forecasting with Casual Variables, Selecting Appropriate Forecasting Models.
Monte Carlo Simulation and Risk Analysis: Monte Carle Simulation Using Analytic Solver Platform, New-Product Development Model, Newsvendor Model, Overbooking Model, Cash Budget Model.

UNIT-V:

Decision Analysis: Formulating Decision Problems, Decision Strategies without Outcome Probabilities, Decision Trees, The Value of Information, Utility and Decision Making.

UNIT-VI:

Recent trends in Embedded and collaborative business intelligence, Visual data recovery, Data Storytelling and Data journalism.

TEXT BOOKS:

1. Business Analytics-Principles, Concepts, and Applications, Marc J. Schniederjans, Dara G. Schniederjans, Christopher M. Starkey, Pearson FT Press
2. Business Analytics, James Evans, Pearson Education
3. Business Analytics, Purba Halady Rao, PHI, 2013

REFERENCES:

1. Business Analytics for Managers: Taking Business Intelligence Beyond Reporting, Gert H. N. Laursen, Jesper Thorlund, 2nd Edition, Wiley Publications
2. Business Analytics: Data Analysis & Decision Making, S. Christian Albright, Wayne L. Winston 5th Edition, 2015
3. Business Intelligence Guidebook: From Data Integration to Analytics, Rick Sherman Elsevier, 2014

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)	**L**	**T/P**	**C**
	3	**0**	**3**

(18OE1AM01) INDUSTRIAL SAFETY

COURSE PRE-REQUISITES: Elements of Mechanical, Civil, Electrical and Industrial Engineering

COURSE OBJECTIVES:

- To achieve an understanding of principles, various functions and activities of safety management
- To communicate effectively information on Health safety and environment facilitating collaboration with experts across various disciplines so as to create and execute safe methodology in complex engineering activities
- To anticipate, recognize, and evaluate hazardous conditions and practices affecting people, property and the environment, develop and evaluate appropriate strategies designed to mitigate risk
- To develop professional and ethical attitude with awareness of current legal issues by rendering expertise to wide range of industries

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

CO-1: Apply risk management principles to anticipate, identify, evaluate and control physical, chemical, biological and psychosocial hazards

CO-2: Communicate effectively on health and safety matters among the employees and with society at large

CO-3: Demonstrate the use of state-of-the-art occupational health and safety practices in controlling risks of complex engineering activities and understand their limitations

CO-4: Interpret and apply legislative / Legal requirements, industry standards, and best practices in accident prevention programmes in a variety of workplaces

UNIT-I:

Safety Management: Evaluation of modern safety concepts – Safety management functions – safety organization, safety department – safety committee, safety audit - performance measurements and motivation – employee participation in safety and productivity.

UNIT-II:

Operational Safety: Hot metal Operation – Boiler, pressure vessels – heat treatment shop - gas furnace operation-electroplating-hot bending pipes – Safety in welding and cutting. Cold-metal Operation- Safety in Machine shop-Cold bending and chamfering of pipes – metal cutting – shot blasting, grinding, painting – power press and other machines

UNIT-III:

Safety Measures: Layout design and material handling - Use of electricity – Management of toxic gases and chemicals – Industrial fires and prevention – Road safety– Safety of sewage disposal and cleaning – Control of environmental pollution – Managing emergencies in industrial hazards.

UNIT-IV:

Accident Prevention: Human side of safety – personal protective equipment – Causes and cost of accidents. Accident prevention programmes - Specific hazard control strategies - HAZOP – Training and development of employees – First Aid – Firefighting devices – Accident reporting investigation.

UNIT-V:

Safety, Health, Welfare & Laws: Safety and health standards – Industrial hygiene – occupational diseases prevention - Welfare facilities – History of legislations related to safety–

pressure vessel act- Indian boiler act- The environmental protection act – Electricity act - Explosive act.

UNIT-VI:

Safe Handling and Storage: Material Handling, Compressed Gas Cylinders, Corrosive Substances, Hydrocarbons, Waste Drums and Containers

TEXT BOOKS:

1. Safety Management, John V. Grimaldi and Rollin H. Simonds, All India Travellers Bookseller, New Delhi, 1989.
2. Safety Management in Industry, Krishnan N. V., Jaico Publishing House, 1996.

REFERENCES:

1. Occupational Safety Manual, BHEL
2. Industrial Safety and The Law, P. M. C. Nair Publisher's, Trivandrum
3. Managing Emergencies in Industries, Loss Prevention of India Ltd., Proceedings, 1999
4. Safety Security and Risk Management, U. K. Singh & J. M. Dewan, A. P. H. Publishing Company, New Delhi, 1996
5. Industrial Safety Management: Hazard Identification and Risk Control, L. M. Deshmukh, McGraw-Hill Education (India) Private Limited, 2005

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)

L	T/P	C
3	0	3

(18OE1AM02) OPERATIONS RESEARCH

COURSE PRE-REQUISITES: Mathematics, Industrial Engineering

COURSE OBJECTIVES:
- To analyze linear programming models in practical and their practical use
- To apply the transportation, assignment and sequencing models and their solution methodology for solving problems
- To apply the theory of games, replacement, inventory and queuing models and their solution methodology for solving problems
- To evaluate the dynamic programming and simulation models

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Apply and solve the dynamic programming problems
CO-2: Apply the concept of non-linear programming
CO-3: Carry out sensitivity analysis
CO-4: Model the real world problem and simulate it

UNIT-I:
Introduction to Operations Research-Definitions of OR, Characteristics of OR, Scope of OR, Classification of Optimization Techniques, models in OR, General L.P Formulation, Graphical solution, Simplex Techniques.

UNIT-II:
Revised simplex method - duality theory - dual simplex method – sensitivity or post optimality analysis - parametric programming

UNIT-III:
Nonlinear programming problem - Kuhn-Tucker condition, min cost flow problem - max flow problem - CPM/PERT

UNIT-IV:
Scheduling and sequencing, Inventory models, deterministic inventory, models - Probabilistic inventory control models - Geometric Programming.

UNIT-V:
Waiting line Models, Single and Multi-channel Problems, Dynamic Programming, Game Theory, Simulation.

UNIT-VI:
Introduction to Genetic Algorithms, Operators, applications to engineering optimization, Problems.

TEXT BOOKS:
1. Operations Research, S. D. Sharma, Kedarnath Ramnath, Meerut, New Delhi
2. Engineering Optimization, S. S. Rao, New Age International Publications, 2014
3. Introduction to Genetic Algorithms, S. N. Sivanandam, Springer

REFERENCES:
1. Operations Research-An Introduction, H. A. Taha, PHI, 2008
2. Principles of Operations Research, H. M. Wagner, PHI, Delhi, 1982
3. Introduction to Optimization: Operations Research, J. C. Pant, Jain Brothers, Delhi, 2008

4. Operations Research, Hitler Liebermann McGraw-Hill Pub., 2009
5. Operations Research, Pannerselvam, Prentice Hall of India, 2010

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)

L	T/P	C
3	0	3

(18OE1AM03) COMPOSITE MATERIALS

COURSE PRE-REQUISITES: Maths, Physics, Chemistry, Engineering Mechanics, Mechanics of Solids

COURSE OBJECTIVES:
- To understand composite materials and their properties, relationship between them and manufacturing methods
- To understand the principles of material science applied to composite materials
- To study the equations to analyze problems by making good assumptions and learn systematic engineering methods to solve practical composite mechanics problems

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Apply fundamental knowledge of mathematics to modeling and analysis of composite materials
CO-2: Understand the manufacturing methods of various composite materials
CO-3: Analyze the failure modes of composites

UNIT-I:
Introduction: Definition – Classification and characteristics of Composite materials. Advantages and application of composites. Functional requirements of reinforcement and matrix. Effect of reinforcement (size, shape, distribution, volume fraction) on overall composite performance.

UNIT-II:
Reinforcements: Preparation-layup, curing, properties and applications of glass fibers, carbon fibers, Kevlar fibers and Boron fibers. Properties and applications of whiskers, particle reinforcements.
Mechanical Behavior of Composites: Rule of mixtures, Inverse rule of mixtures. Isostrain and Isostress conditions.

UNIT-III:
Manufacturing of Metal Matrix Composites: Casting – Solid State diffusion technique, Cladding – Hot isostatic pressing. Properties and applications.
Manufacturing of Ceramic Matrix Composites: Liquid Metal Infiltration – Liquid phase sintering. Manufacturing of Carbon – Carbon composites: Knitting, Braiding, Weaving. Properties and applications.

UNIT-IV:
Manufacturing of Polymer Matrix Composites: Preparation of Moulding compounds and prepregs – hand layup method – Autoclave method – Filament winding method – Compression moulding – Reaction injection moulding. Properties and applications, Introduction to Machining of Composites.

UNIT-V:
Elastic Behavior of Laminate: Basic assumptions, Strain-displacement relations, Stress-strain relation of layer within a laminate, Force and moment resultant, General load–deformation relations, Analysis of different types of laminates

UNIT-VI:
Strength: Laminar Failure Criteria-strength ratio, maximum stress criteria, maximum strain criteria, interacting failure criteria, hygrothermal failure. Laminate first ply failure-insight

strength; Laminate strength-ply discount truncated maximum strain criterion; strength design using caplet plots; stress concentrations.

TEXT BOOKS:

1. Material Science and Technology, Vol. 13–Composites, R. W. Cahn – VCH, West Germany
2. Analysis and Performance of Fiber Composites, Third Edition, B. D. Agarwal, Wiley Publishers

REFERENCES:

1. Mechanics of Composite Materials, Second Edition. Robert M. Jones, Scripta Book Company
2. Materials Science and Engineering-An Introduction, W. D. Callister Jr., Adapted by R. Bala Subramaniam, John Wiley & Sons, NY, Indian Edition, 2007
3. Composite Materials, K. K. Chawla
4. Composite Materials Science and Applications, Deborah D. L. Chung
5. Composite Materials Design and Applications, Danial Gay, Suong V. Hoa and Stephen W. Tasi

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)

L	T/P	C
3	0	3

(18OE1PS01) WASTE TO ENERGY

COURSE PRE-REQUISITES: None

COURSE OBJECTIVES:
- To create awareness in students of energy conservation
- To identify the use of different types of Bio waste energy resources
- To understand different types of bio waste energy conservations
- To detect different waste conversion into different forms of energy

COURSE OUTCOMES: After completion of the course, students should be able to
CO1: Find different types of energy from waste to produce electrical power
CO2: Estimate the use of bio waste to produce electrical energy
CO3: Understand different types of bio waste and its energy conversions
CO4: Analyze the bio waste utilization to avoid the environmental pollution

UNIT-I:
Introduction to Energy from Waste: Classification of waste as fuel, Agro based, Forest residue, Industrial waste, MSW (Municipal solid waste) – Conversion devices – Incinerators, Gasifiers, Digestors

UNIT-II:
Biomass Pyrolysis: Pyrolysis – Types, slow fast – Manufacture of charcoal – Methods - Yields and application – Manufacture of pyrolytic oils and gases, yields and applications.

UNIT-III:
Biomass Gasification: Gasifiers – Fixed bed system – Downdraft and updraft gasifiers – Fluidized bed gasifiers – Design, construction and operation – Gasifier burner arrangement for thermal heating – Gasifier engine arrangement and electrical power – Equilibrium and kinetic consideration in gasifier operation.

UNIT-IV:
Biomass Combustion: Biomass stoves – Improved chullahs, types, some exotic designs, Fixed bed combustors, Types, inclined grate combustors, Fluidized bed combustors, Design, construction and operation - Operation of all the above biomass combustors.

UNIT-V:
Biogas: Properties of biogas (Calorific value and composition) - Biogas plant technology and status - Bio energy system - Design and constructional features - Biomass resources and their classification - Biomass conversion processes - Thermo chemical conversion - Direct combustion - biomass gasification - pyrolysis and liquefaction - biochemical conversion - anaerobic digestion.

UNIT-VI:
Types of biogas Plants – Applications - Alcohol production from biomass - Bio diesel production - Urban waste to energy conversion - Biomass energy programme in India.

TEXT BOOKS:
1. Biogas Technology-Transfer and Diffusion, M. M. EL-Halwagi, Elsevier Applied Science Publisher, New York, 1984
2. Introduction to Biomass Energy Conversions, Sergio Capareda

REFERENCES:

1. Non-Conventional Energy, Desai Ashok V., Wiley Eastern Ltd., 1990
2. Biogas Technology - A Practical Handbook, Khandelwal, K. C. and Mahdi, S. S., Vol. I & II, Tata McGraw-Hill Publishing Co. Ltd., 1983
3. Food, Feed and Fuel from Biomass, Challal, D. S., IBH Publishing Co. Pvt. Ltd., 1991
4. Biomass Conversion and Technology, C. Y. WereKo-Brobby and E. B. Hagan, John Wiley & Sons, 1996

VNR VIGNANA JYOTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

M.Tech. III Semester (SE)	**L**	**T/P**	**C**
	0	**16**	**8**

(18PW4SE03) PROJECT PART-I

M.Tech. IV Semester (SE)	**L**	**T/P**	**C**
	0	**28**	**14**

(18PW4SE04) PROJECT PART-II

COURSE OUTCOMES: After completion of the course, students should be able to
CO-1: Identify and formulate the problem (Industry/technical/societal)
CO-2: Analyze, design and develop a solution to industry/technical/societal problems
CO-3: Implement and execute the solution
CO-4: Demonstrate effective communication skills through oral presentation
CO-5: Engage in effective written communication through project report

COURSE OUTLINE:

- M.Tech. project work shall be for a minimum duration of 40 weeks spread over two semesters i.e., Project Part-I in III semester and Project Part-II in IV semester.
- A student shall be permitted to register for the major project after satisfying the attendance requirement in all the courses, i.e., theory and practical courses.
- Project reviews namely Project Review I and Project Review II in III semester and Project Review III and Project Pre-submission Seminar in IV semester shall be conducted during the course of Project work.
- A Project Review Committee (PRC) consisting of the Head of the Department as Chairperson and PG Coordinator, Project Supervisor and one senior faculty member of the Department offering the M. Tech. programme as members shall evaluate the progress of project work.
- In Project Review I, a student, in consultation with his Project Supervisor, shall present the title, objective and plan of action of his/her project work to the PRC for approval within four weeks from the commencement of III semester.
- A student can initiate the project work only after obtaining the approval of the PRC.
- The work on the project shall be initiated at the beginning of the III semester.
- Project Review II shall be conducted and evaluated at the end of the III semester.
- Project Review III shall be conducted during IV semester to examine the overall progress of the project work.
- A project pre-submission seminar shall be conducted to decide whether or not the project is eligible for final submission.
- After approval from the PRC, a soft copy of the thesis shall be submitted for PLAGIARISM check to the Examination Branch.
- At the end of IV semester upon fulfilling the above conditions, project viva-voce shall be conducted.
- A student shall submit project progress in prescribed report format during each of the project reviews.