



**PUNE VIDYARTHI GRIHA'S  
COLLEGE OF ENGINEERING AND TECHNOLOGY, PUNE-9  
(AFFILIATED TO UNIVERSITY OF PUNE, PUNE)**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CURRICULUM BOOK**

**ACADEMIC YEAR: 2016-17**

**FOR THE PROGRAMME  
T. E. (INFORMATION TECHNOLOGY)**



**PUNE VIDYARTHI GRIHA'S  
COLLEGE OF ENGINEERING AND TECHNOLOGY**

### **VISION**

**TO ACHIEVE EXCELLENCE IN ENGINEERING EDUCATION**

### **MISSION**

- **To satisfy all stakeholders**
- **To develop ethical, highly motivated engineering professionals with good human values, requisite skills and competencies**
- **To adopt innovative teaching mechanisms**
- **To promote research culture**
- **To contribute to country's economic development**
- **To be responsive to changes in technology, socio-economic and environmental conditions**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**VISION**

**To Empower Students to Face the Technological Challenges of 21<sup>st</sup> Century  
by Imparting Quality Education in the Field of Information Technology**

**MISSION**

- 1) To impart knowledge through innovative teaching-learning process to cater the needs of industries and higher education.**
- 2) To inculcate good human values, professional competencies and create awareness about global technologies in the field of Computer Engineering.**
- 3) To respond to rapid changes in the field of Information Technology.**

**PROGRAM EDUCATIONAL OBJECTIVES**

**PEO1:** Possess strong fundamental concepts in Engineering Science and Technology to address future technological challenges of Information Technology.

**PEO2:** Possess knowledge and skills in the field of Information Technology for engineering problems with innovative approaches.

**PEO3:** . Possess behavioral aspects for research, entrepreneurship and higher studies in the field of Computer Science and Information Technology.

**PEO4:** Have commitment to ethical practices in the field of Information Technology and, societal contributions through communities and life-long learning.

**PEO5:** Possess better interpersonal and presentation skills to cope up with the rapid changes in the field of Information Technology at global level.

## Curriculum book of TE (IT)

### PROGRAMME OUTCOMES

**The Programme Outcomes of the Department of Information Technology are:**

- PO1 an ability to apply knowledge of computing, engineering mathematics, statistics, science, and engineering and technology;
- PO2 an ability to identify and analyze the problem, provide a systematic solution by conducting experiments, interpreting the data and drawing substantial conclusion;
- PO3 an ability to design, implement, and evaluate a software or a software/hardware system, component, or process to meet desired needs within realistic constraints;
- PO4 an ability to identify, formulate, and provide systematic solutions to complex engineering problems and validate the solution;
- PO5 an ability to apply appropriate resources, skills, modern engineering tools and technologies necessary for practice as a IT professional;
- PO6 an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems with necessary constraints and assumptions;
- PO7 an ability to analyze the local and global impact of computing on individuals, organizations and society;
- PO8 an ability to understand professional, ethical, legal, security and social issues and responsibilities;
- PO9 an ability to function effectively as an individual or as a team member to accomplish a desired goal(s) in multidisciplinary environment;
- PO10 an ability to engage in life-long learning and continuing professional development to cope up with fast changes in the technologies/tools with the help of open electives, professional organizations and extra-curricular activities;
- PO11 an ability to communicate effectively in engineering community at large by means of effective presentations, report writing, paper publications, demonstrations;
- PO12 an ability to understand engineering, management, financial aspects, performance, optimizations and time complexity necessary for professional practice;
- PO13 an ability to apply design and development principles in the construction of software systems of varying complexity.

## **INDEX**

<b>Sr. No.</b>	<b>Titles</b>	<b>Page No.</b>
<b>1.</b>	<b>TE IT Course Structure</b>	<b>01</b>
<b>2.</b>	<b>Course in TE IT Semester I</b>	<b>02</b>
2.1	Computer Network Technology	03
2.2	Theory Of Computation	07
2.3	Database Management Systems	09
2.4	Software Engineering	14
2.5	Web Engineering and Technology	17
<b>3.</b>	<b>Course in TE IT Semester II</b>	<b>21</b>
3.1	Design and Analysis of Algorithms	22
3.2	Systems Programming	25
3.3	Operating System	28
3.4	Multimedia Technologies	31
3.5	Information Tech Project Management	34

## Curriculum book of TE (IT)

### Syllabus Structure of Savitribai Phule Pune University, Pune

#### Course Structure T. E. (INFORMATION TECHNOLOGY) 2012 Course

Course Code	Course	Teaching Scheme Hrs/Week			Examination Scheme					Marks Total
		L	T	P	Theory		T W	P	O	
					In Sem	Theory				
<b>Legends:</b>										
L: Lectures    T: Tutorial    P: Practical    TW: Term Work    O: Oral										
<b>Semester -III</b>										
314441	Computer Network Technology	3			30	70				100
314442	Theory Of Computation	4			30	70				100
314443	Database Management Systems	4			30	70				100
314444	Software Engineering	3			30	70				100
314445	Web Engineering and Technology	3			30	70				100
314446	Software Laboratory- I			4			50	50		100
314447	Database Management Systems Laboratory			4				50	50	100
314448	Employability Skill Development Laboratory	1		2			50			50
		18		10	150		100	100	50	750
<b>Semester-IV</b>										
314449	Design and Analysis of Algorithms	4			30	70				100
314450	Systems Programming	4			30	70				100
314451	Operating System	4			30	70				100
314452	Multimedia Technologies	3			30	70				100
314453	Information Tech Project Management	3			30	70				100
314454	Operating System Laboratory			4			50	50		100
314455	Software Laboratory - II			4				50	50	100
314456	Seminar and Technical Communication Lab			2			50			50
		18	10		150	100	100	50	350	750

# TE IT

## Semester I

**Computer Network Technology**

<b>Course Title:</b>	<b>Computer Network Technology</b>	<b>Course Number:</b>	<b>314441</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 3Hrs/Week		<b>Laboratories:</b> 2 Hrs/Week	
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Term-work	Practical/Oral
	<b>Indirect Methods</b>	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
<b>Prerequisites</b>	Foundation of Computer Networks		
<b>Course Objectives</b>			
1	To understand services offered by important layers of OSI model		
2	To provide routing and network management techniques		
3	To understand various application layer protocols and its applications in client / server environment		
4	To understand various WAN technologies in computer networks		
<b>Course Outcomes</b>			
CO1	Students will be able to understand the OSI model and its layer responsibilities in detail		
CO2	Students will be able to explain various routing protocols and techniques and its related management issues at large		
CO3	Students will be able to understand working principle of client/server application with respect application layer protocols		
CO4	Students will obtain thorough knowledge of various Wireless technologies		
<b>Course Contents</b>			
<b>Unit-I</b>	<b>NETWORK LAYER 6 Hours</b>		
	Packet Switching, Virtual Circuits, Datagram Routing Algorithms: Optimality Principle, Shortest path routing- Dijkstra's!lgorithms, Distance Vector Routing, Link State Routing, Counting to infinity problem, RIP, OSPF, BGP IP Addressing IPv4, IP Address Classes, Subnetting, CIDR/Supernetting, IP Fragmentation, ARP, DHCP, RARP, ICMP, IPv6.		
	<b>Practical</b>		
	1.Installation of Local Area Network (cables, connectors, topologies, switches/hubs, crimping tool, IP addressing scheme, Subnetting, College Network Design) a. Cross-over cable b. Star Topology c. Hierarchical Topology		
	2.Using a Network Simulator (e.g. packet tracer) configure		





	c. DHCP		
<b>Unit-IV</b>	<b>WIRELESS LANS, PANS AND MANS</b>		<b>6 Hours</b>
	Introduction (Infrastructure and Ad-hoc Networks), Comparison of Wireless Networks in ISM Band Fundamentals of WLAN – technical issues, Network Architecture, IEEE 802.11- physical layer, Mac Layer Mechanism, CSMA/CA, Bluetooth - Specification, Transport Layer, Middleware Protocol Group, Bluetooth Profiles, IEEE 802.16 –differences between IEEE 802.11 and 802.16, Physical Layer, Data Link Layer		
	<b>Practical</b>		
	1. Using Network Simulator 2/ OMNET simulate a. Local Area Network b. WLAN		
<b>Unit- V</b>	<b>AD-HOC NETWORKS AND SENSOR NETWORKS</b>		<b>6 Hours</b>
	Introduction to MANETs, Sensor Networks, Operating Environment Constraints, Protocols supported by Wireless Networks Applications of Sensor Networks Sensor Node Architecture (hardware components) Sensor Network Architectures (Concept of sink and source, Topologies, Design Principles) Radio Propagation and Propagation Impairments MAC Protocol: Fundamentals, STEM, S-MAC, LEACH, IEEE 802.15.4		
	<b>Practical</b>		
	Using Network Simulator 2/ OMNET simulate a. MANET b. WSN		
<b>Unit-VI</b>	<b>ROUTING IN SENSOR NETWORKS AND RECENT TRENDS</b>		<b>6 Hours</b>
	Routing in MANET : AODV, DSDV, DSR Naming and Addressing in WSN : Basic concepts, MAC address, Distributed assignment of locally unique addresses, Content-based and geographic addressing Routing in Sensor Networks : Challenges and design issues Routing Protocols for WSNs: Flooding, SPIN, PEGASIS, Directed Diffusion, Geographic Routing Recent Trends (References web, no formal text) Software Defined Networking Wi-FiOffloads 100G Ethernet and its variants Internet of Things (IoT) and Web of Things Bring your own Device (BYOD)		
	<b>Practical</b>		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	Andrew S. Tanenbaum	Computer Networks	PHI, Fifth Edition
T2	KazemSohraby, Daniel Minoli, TaiebZnati	Wireless Sensor Network	Wiley
T3	C. Siva Ram Murthy and B. S. Manoj	Ad Hoc Wireless Networks: Architectures and Protocols	Prentice Hall
<b>Reference Books</b>			

PVG's COET, PUNE-9  
DEPARTMENT OF C  
**Curriculum book of TE (IT)**

**2016 -  
17**

R1	James F. Kurose and Keith W. Ross	Computer Networking: A Top-Down Approach Featuring the Internet	Pearson Education
R2	Holger Karl and Andreas Willig	Protocols and Architectures for Wireless Sensor Networks	Wiley
R3	Feng Zhao and Leonidas J Guibas	Wireless Sensor Networks: In Information Processing Approach	Morgan Kaufmann
R4	Natalia Olifer, Victor Olifer	Computer Network Principles Technologies and Protocols for Network Design	Wiley
<b>Self-Learning Facilities</b>	NPTEL Lecture Series		
<b>Web Resources</b>	<a href="http://www.computernetwokingnotes.com">www.computernetwokingnotes.com</a>		
<b>Research papers for reference</b>	<b>Author</b>	<b>Title of Paper</b>	<b>Journal/Transaction</b>
1	Aarti, Dr.S.S.Tyagi	Study of MANET: Characteristics, challenges, Application and Security Attacks	International Journal of Adv Research in CS & SE.
2	Mohit Kumar , Rashmi Mishra	An overviews of MANET: Histry, challenges, and Applications	IJCSE
<b>Contents beyond Syllabus</b>			
<b>Additional Experiments</b>	1. Socket Chatting Application		
	2. Protocol headers using Wire shark		

**THEORY OF COMPUTATION**

<b>Course Title:</b>	<b>THEORY OF COMPUTATION</b>	<b>Course Number:</b>	<b>314442</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 4 Hrs/Week	<b>Laboratories:</b> ----		
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination:30 Marks Unit tests	End Semester Examination: 70 Marks
	<b>Indirect Methods</b>	Assignments, problem solving	Quiz, Q&A session
<b>Prerequisites</b>	Discrete Structure		
<b>Introduction of Course</b>			
<b>Course Objectives</b>			
1	To Study computing machines by describing, classifying and comparing different types of computational models.		
2	To study & develop fundamentals for 'Computational Theory		
3	To develop as ability to apply mathematical knowledge and logic in solving problems		
<b>Course Outcomes</b>			
CO1	Students should be able to understand different computational models		
CO2	Students should be able to design Regular Grammar, Finite Automata, Context Free Grammar, Pushdown Automata, Post Machines, and Turing Machines		
CO3	Students should be able to perform inter-conversions of different computing models		
CO4	Students should be able to understand Pumping Lemma, Properties of Regular Languages and Context Free Languages		
CO5	Students should be able to understand Decidable Languages and Turing Reducibility		
CO6			
<b>Course Contents</b>			
<b>Unit-I</b>	<b>INTRODUCTION</b>		
	. Basic Concepts: Symbols, Strings, Language, Formal Language, Natural Language. Basic Machine And Finite State Machine. Finite Automata: Definition And Construction - DFA , NFA, NFA with epsilon-Moves, Minimization Of FA, Equivalence of NFA and DFA, Conversion of NFA with epsilon moves to NFA, Conversion of NFA With epsilon moves to DFA, Moore and Mealy Machines, Inter-conversion between Moore and Mealy Machines.		
<b>Unit-II</b>	<b>REGULAR EXPRESSIONS, REGULAR GRAMMAR AND LANGUAGES</b>		
	Definition and Identities of Regular Expressions, Regular Grammar and Finite Automata: FA to RG and RG to FA, Left Linear and Right Linear Grammar and		

	Inter-conversion between them. Closure Properties of RLs, Pumping Lemma for RL.		
<b>Unit-III</b>	<b>CONTEXT FREE GRAMMAR AND LANGUAGES</b>		
	Definition and Construction of CFG, Definition and Generation of CFL from CFG. Ambiguous Grammar and Removal of Ambiguity. Simplification of Grammar. Normal Forms of Grammar: CNF and GNF. Chomsky Hierarchy.		
<b>Unit-IV</b>	<b>PUSHDOWN AUTOMATA</b>		
	Definition and Construction of DPDA and NPDA. Equivalence of PDAs and CFGs, Closure Properties Of CFLs, Concept of Post Machines		
<b>Unit- V</b>	<b>TURING MACHINES</b>		
	Definition and Construction of Turing Machines. Languages of TM. Types of TM. Time Complexity of TM, Halting Problem, Church's Turing Hypothesis, Comparison And Applications of DFA, PDA and TM.		
<b>Unit-VI</b>	<b>DECIDABILITY AND REDUCIBILITY</b>		
	Decidable Languages, Decidable Problems Concerning Regular Languages, Decidable Problems Concerning Context-Free Languages. Decidable Problems With The TM, Turing Reducibility.		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	John C. martin	Introduction to Language and Theory of Computation	TMH, Third Edition. 978-0-07-066048-9.
T2	Michel Sipser	Introduction to Theory of Computation	Thomson Course Technology, Second Edition 0-534-95097-3.
	Kavi Mahesh	Theory of Computation	Wiley-India, ISBN : 978-81-265-3311-4
<b>Reference Books</b>			
R1	Hopcroft Ulman	Introduction To Automata Theory, Languages And Computations	Pearson Education Asia, 2nd Edition
R2	Daniel I.A. Cohen	Introduction to Computer Theory	Wiley-India, ISBN: 978-81-265-1334-5
R3	E V Krishnamurthy	Introduction to Theory of Computer Science	EWP Second 2nd Edition
R4	K.L.P Mishra, N. Chandrasekaran	Theory Of Computer Science(Automata, Languages and Computation)	Prentice Hall India, 2nd Edition
R5	Daniel I.A. Cohen	Introduction to Automata Theory Languages and Computations	Pearson Education Asia, Second Edition.

**DATABASE MANAGEMENT SYSTEMS**

<b>Course Title:</b>	<b>DATABASE MANAGEMENT SYSTEMS</b>	<b>Course Number:</b>	<b>314443</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 3 Hrs/Week		<b>Laboratories:</b> 8 Hrs/Week	
<b>Course Assessment Methods</b>	<b>Direct methods</b>	On-line/In-sem Examination: 50/30 Marks	Theory/End Semester Examination: 50/70 Marks
		Term-work	Practical/Oral
	<b>Indirect Methods</b>	Assignments, Presentations, Case Study	Seminars, Quiz, Q&A session, Group Discussion
<b>Introduction of Course</b>			
<b>Course Objectives</b>			
1	To learn and understand the fundamental concepts of database management		
2	To learn and understand strong formal foundation in database concepts, technology and practice.		
3	To learn and understand systematic database design approaches covering conceptual design, logical design and an overview of physical design.		
4	To learn basic issues of transaction processing and concurrency control.		
5	To learn and understand various Database Architectures and Applications.		
6	To learn and Understand how analytics and big data affect various functions now and in the future		
<b>Course Outcomes</b>			
CO1	1. Define basic functions of DBMS & RDBMS.		
CO2	2. Analyze database models & entity relationship models.		
CO3	3. Design and implement a database schema for a given problem-domain.		
CO4	4. Appreciate the impact of analytics and big data on the information industry and the external ecosystem for analytical and data services.		
CO5	5. Programming PL/SQL including stored procedures, stored functions, cursors and packages		
CO6	6. Appreciate the impact of analytics and big data on the information industry and the external		
<b>Course Contents</b>			
<b>Unit-I</b>	<b>UNIT - I INTRODUCTION</b>		
	Database Concepts, Database System Architecture, Data Modeling : Data Models, Basic Concepts, entity, attributes, relationships, constraints, keys, E-R and EER diagrams: Components of E-R Model, conventions, converting E-R diagram into tables, EER Model components, converting EER diagram into tables, legacy system model, Relational Model: Basic concepts, Attributes and Domains, Codd's Rules, Relational Integrity: Domain, Entity, Referential Integrities, Enterprise Constraints, Views, Schema		

	Diagram, Database Design: Functional Dependency, Purpose of Normalization, Data Redundancy and Update Anomalies, Functional Dependency-Single Valued Dependencies. Single Valued Normalization: 1NF, 2NF, 3NF, BCNF. Decomposition: lossless join decomposition and dependency preservation, Multi valued Normalization (4NF), Join Dependencies and the Fifth Normal Form.
<b>Unit-II</b>	<b>UNIT - II SQL AND PL/SQL</b>
	<p>Introduction to SQL: Characteristics and advantages, SQL Data Types and Literals, DDL, DML, DCL, TCL, SQL Operators, Tables: Creating, Modifying, Deleting, Views: Creating, Dropping, Updating using Views, Indexes, Nulls</p> <p>SQL DML Queries: SELECT Query and clauses, Set Operations, Predicates and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate Functions, Nested Queries, Database Modification using SQL Insert, Update and Delete Queries, concept of Stored Procedures, Cursors, Triggers, assertions, roles and privileges</p> <p>Programmatic SQL: Embedded SQL, Dynamic SQL, Advanced SQL-Programming in MYSQL, SQL 2.0/SQL for OODB, No SQL- MongoDB</p> <p><b>Practical</b></p> <p>Design and Develop SQL/NoSQL DDL Statements which demonstrate the use of SQL objects such as Table, View.</p> <p>Design at least 10 SQL/NoSQL queries for suitable database application using SQL/NoSQL DML Statements: Insert, Select, Update, Delete with operators, functions, and set operator.</p> <p>Design at least 10 SQL/NoSQL queries for suitable database application using SQL DML statements: all types of Join, Sub-Query and View.</p> <p>Write a PL/SQL block or Stored Procedure to calculate the Grade of minimum 10 students.</p> <p>Write a cursor in which percentages for each student are calculated. If it is greater than or equal to 60, make that entry into stud_First (Exam_no, Name, Sub1, Sub2, Sub3, Percent) table. If less then 60, make that entry into stud_Pass (Exam_no, Name, Sub1, Sub2, Sub3, Percent) table. If student is fail in any subject (less than 40 in any subject), make that entry into stud_fail (Exam_no, Name, sub1, sub2, sub3) table.</p> <p>Write a PL/SQL function which accepts basic salary of an employee and returns a Gross salary of an employee. <math display="block">\text{Gross salary} = \text{Basic} + \text{HRA} + \text{DA} + \text{TA}</math></p> <p>Write a Trigger that maintains a log of a account (acc_no, name, balance) table. Whenever there is any update of account balance, LogTable maintains AccountNo, Name, amount that is deposited or withdrawn, Type of transaction- if deposite then 'D' – if withdraw then 'W', and Timestamp which includes Day, Date, Time.</p>
<b>Unit-III</b>	<b>DATABASE TRANSACTIONS</b>
	Basic concept of a Transaction, Transaction Management, Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability: Conflict and View, Cascaded Aborts, Recoverable and Non-recoverable Schedules, Concurrency Control: Need, Locking Methods, Deadlocks, Time-stamping



	<p>Methods, Optimistic Techniques, Recovery methods: Shadow-Paging and Log-Based Recovery, Checkpoints, Performance Tuning, Query Optimization with respect to No SQL Database</p> <p><b>Practical</b></p> <p>Implement any five queries on MongoDB Database.</p> <p>Design and Implement Student Data Management System using Java &amp; MySQL database Connectivity. Student data consist of Sid, Sname, Sub1, Sub2, Sub3 marks. And perform Insert, Delete, Update &amp; Search Operation</p> <p>Design and Implement Student Data Management System to collect marks of all the students and store it in database. After collecting marks of all students execute a Stored Procedure from Java to Calculate the Percentage marks and store it back to the database.</p>
<b>Unit-IV</b>	<b>ADVANCE AND EMERGING DATABASE CONCEPT</b>
	<p>Database Architectures: Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Introduction to Parallel Databases, Key elements of Parallel Database Processing, Architecture of Parallel Databases, Introduction to Distributed Databases, Architecture of Distributed Databases, Distributed Database Design, Internet Databases, Database Connectivity using Mongo databases</p> <p><b>Practical</b></p> <p>Implement aggregation with employee collection using MongoDB.</p> <p>Implement Map Reduce operation with following example using MongoDB.</p> <p>Design and Implement Student Data Management System using Java &amp; MongoDB database Connectivity. Student data consist of Sid, Sname, Class, City, Marks. and perform Insert, Delete, Update &amp; Search Operation</p>
<b>Unit- V</b>	<b>LARGE SCALE DATA MANAGEMENT</b>
	<p>Introduction to Big Data, XML: DTD, XML Schemas, XQuery, XPath, JSON: Overview, Data Types, Objects, Schema, JSON with Java/PHP/Ruby/Python, Hadoop: HDFS, HBase: Overview, HBase Data Model, HBase Region, Hive, SSD</p> <p><b>Practical</b></p> <p>Create simple JSONObjects and JSONArray objects using JAVA</p> <p>Implement Encode and Decode JSON Objects using Java</p>
<b>Unit-VI</b>	<b>DATA WAREHOUSING AND DATA MINING</b>
	<p>Teradata RDBMS, Teradata Technology, Data Warehousing : Introduction, Evolution of Data Warehouse, Characteristics, Benefits, Limitation of Data Warehousing, Main Components of Data Warehouse, Conceptual Models, Data Mart, OLAP, Data Mining : Process, Knowledge Discovery, Goals of Data Mining, Data Mining Tasks, Machine learning for Big Data, Business Intelligence, Business analytics.</p> <p>Emerging Database Technologies: Introduction, Cloud Computing and Data Management, Mobile Databases, Dealing with Massive Datasets-Map Reduce and Hadoop. Introduction to SQLite database, XML databases.</p>



	<b>Practical</b>		
	Implement any Machine learning algorithm for BIG data		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	Silberschatz A., Korth H., Sudarshan S.	Database Management Systems	MCGraw Hill
T2	S.K.Singh	Database Systems: Concepts, Design and Application	Pearson
T3	Connally T., Begg C	Database Systems	Pearson Education
<b>Reference Books</b>			
R1	Kristina Chodorow, Michael Dirolf	MangoDB: The Definitive Guide	O'Reilly Publications
R2	Mario Piattini, Oscar Diaz Tom White	Hadoop: The Definitive Guide	O'Reilly Publications
R3	Jiawei Han, Micheline Kamber, Jian Pei	Data Mining: Concepts and Techniques	Elsevier
R4	Bill Schmarzo	Big Data: Understanding How Data Powers Big Business	Wiley
R5	Alex Holmes	Hadoop in Practice", DreamTech Press	Wiley-Dreamtech Press
<b>Self-Learning Facilities</b>	NPTEL Lecture Series		
<b>Web Resources</b>	<a href="http://nosql-database.org">http://nosql-database.org</a>		
<b>Contents beyond Syllabus</b>			
<b>Additional Experiments</b>	Design and Implement Student Data Management System using Java & MySQL database Connectivity. Student data consist of Sid, Sname, Sub1, Sub2, Sub3 marks. And perform Insert, Delete, Update & Search Operation		
	Write a Trigger that maintains a log of a account (acc_no, name, balance) table. Whenever there is any update of account balance, LogTable maintains AccountNo, Name, amount that is deposited or withdrawn, Type of transaction- if deposite then 'D' – if withdraw then 'W', and Timestamp which includes Day, Date, Time.		
<b>Assignments</b>			
1	Design and Develop SQL/NoSQL DDL Statements which demonstrate the use of SQL objects such as Table, View.		
2	Design at least 10 SQL/NoSQL queries for suitable database application using SQL/NoSQL DML Statements: Insert, Select, Update, Delete with operators, functions, and set operator.		
3	Design at least 10 SQL/NoSQL queries for suitable database application using SQL DML statements: all types of Join, Sub-Query and View.		

4	Write a PL/SQL block or Stored Procedure to calculate the Grade of minimum 10 students.
5	Write a cursor in which percentages for each student are calculated. If it is greater than or equal to 60, make that entry into stud_First (Exam_no, Name, Sub1, Sub2,Sub3, Percent) table. If less then 60, make that entry into stud_Pass (Exam_no, Name, Sub1, Sub2, Sub3, Percent) table. If student is fail in any subject(less than 40 in any subject), make that entry into stud_fail(Exam_no,Name,sub1,sub2,sub3) table.
6	Write a PL/SQL function which accepts basic salary of an employee and returns a Gross salary of an employee. $\text{Gross salary} = \text{Basic} + \text{HRA} + \text{DA} + \text{TA}$
7	Write a Trigger that maintains a log of a account (acc_no, name, balance) table. Whenever there is any update of account balance, LogTable maintains AccountNo, Name, amount that is deposited or withdrawn, Type of transaction- if deposite then 'D' – if withdraw then 'W', and Timestamp which includes Day, Date, Time.
8	Implement aggregation with employee collection using MongoDB.
9	Implement Map Reduce operation with following example using MongoDB.
10	Implement any five queries on MongoDB Database.
11	Create simple JSONObjects and JSONArray objects using JAVA
12	Implement Encode and Decode JSON Objects using Java
13	Implement any Machine learning algorithm for BIG data
14	Design and Implement Student Data Management System using Java & MySQL database Connectivity. Student data consist of Sid, Sname, Sub1, Sub2, Sub3 marks. And perform Insert, Delete, Update & Search Operation
15	Design and Implement Student Data Management System to collect marks of all the students and store it in database. After collecting marks of all students execute a Stored Procedure from Java to Calculate the Percentage marks and store it back to the database.
16	Design and Implement Student Data Management System using Java & MongoDB database Connectivity. Student data consist of Sid, Sname, Class, City, Marks. and perform Insert, Delete, Update & Search Operation

**Software Engineering**

<b>Course Title:</b>	<b>Software Engineering</b>	<b>Course Number: 314444</b>	
<b>Designation of Course</b>	Core		
<b>Teaching Scheme:</b> 3 Hrs/Week		<b>Laboratories:</b> --	
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Term-work : NIL	Practical/Oral : NIL
	<b>Indirect Methods</b>	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
<b>Course Objectives</b>			
1	To understand the nature of software complexity in various application domains, disciplined way of software development and software lifecycle process models.		
2	To introduce principles of agile software development, the SCRUM process and agile practices.		
3	To know methods of capturing, specifying, visualizing and analyzing software requirements.		
4	To understand concepts and principles of software design and architecture.		
5	To understand user-centeredness approach and principles of designing effective user interfaces.		
6	To present formal methods, automation and recent trends in software engineering.		
<b>Course Outcomes</b>			
CO1	At the end of course students will Identify unique features of various software application domains and classify software applications.		
CO2	Choose and apply appropriate lifecycle model of software development.		
CO3	Describe principles of agile development, discuss the SCRUM process and distinguish agile process model from other process models.		
CO4	Identify user needs and formulate software specifications.		
CO5	Analyze software requirements by applying various modeling techniques.		
CO6	Translate the requirements model into the design model		
CO7	Describe user-interface design principles.		
CO8	Explain the cleanroom design method.		
CO9	List and classify CASE tools and discuss recent trends and research in software engineering.		

## Curriculum book of TE (IT)

### Course Contents

<b>Unit-I</b>	<b>SOFTWARE ENGINEERING PROCESS</b>
	Nature of Software – Application domains, web-apps, mobile-apps, cloud computing, product line software Introduction to Software Engineering – The discipline, layers, the process (guiding principles), the practice (guiding principles) and myths Process Models – Generic process model, process assessment and improvement, prescriptive models, specialized models, unified process, product and process
<b>Unit-II</b>	<b>AGILE DEVELOPMENT PROCESS</b>
	Agile Development – Agile manifesto, agility and cost of change, agility principles, myth of planned development, toolset for the agile process Extreme Programming – XP values, process, industrial XP SCRUM – process flow, scrum roles, scrum cycle description, product backlog, sprint planning meeting, sprint backlog, sprint execution, daily scrum meeting, maintaining sprint backlog and burn-down chart, sprint review and retrospective Agile Practices - test driven development, refactoring, pair programming, continuous integration, exploratory testing versus scripted testing
<b>Unit-III</b>	<b>REQUIREMENTS ENGINEERING</b>
	Requirements Capturing - requirements engineering (elicitation, specification, validation, negotiation), eliciting requirements, elicitation techniques, developing use cases, building requirements model, negotiating requirements, requirements monitoring, validating requirements, prioritizing requirements (kano diagram) , Requirements Analysis – basics, scenario based modeling, UML models, data modeling, data and control flow model, behavioral modeling using state diagrams Agile Requirements - user stories, 3 Cs of user story, INVEST characteristics
<b>Unit-IV</b>	<b>SOFTWARE DESIGN</b>
	Software Design – definition of design, translating requirements model to design model, design considerations (quality guidelines and attributes), design concepts, design model, design strategies or methods (function-oriented, data-flow-oriented, object-oriented, data-structure-centered, aspect-oriented), design methods classification , design trade-offs, Software architecture, architectural styles (data-centered, data-flow, call and return, layered, peer-to-peer, publish-subscribe, event-based, client-server), architectural trade-off analysis method (ATAM), domain-specific architectures and product-lines
<b>Unit- V</b>	<b>USER INTERFACE DESIGN</b>
	User Interface – Seeheim model and definition of user interface User-centeredness in design - dealing with different types of users, collecting user-requirements, building narratives, creating personas and scenarios Interface design principles–place the user in control, reduce user’s memory load, make interface consistent, Shneiderman's 8 Golden Rules UI Analysis – context of use, user analysis, task analysis Interface design steps – user interface design process, applying design steps, interface design issues Usability - characteristics (ISO, Shneiderman, Nielson) , principles ( principle of proximity, visibility, visual feedback, visual prominence, mental models and metaphors, consistency, affordance and constraints, confirmation, Hick’s law, Fitt’s law)

<b>Unit-VI</b>	<b>FORMAL METHODS, AUTOMATION AND TRENDS IN SOFTWARE ENGINEERING</b>		
	<p>Cleanroom Design – cleanroom strategy, process model, black-box, state-box, clear-box specifications, design refinement and verification, cleanroom testing            Software configuration management – SCM basics, SCM repository, SCM process            CASE – taxonomy, tool-kits, workbenches, environments, components of CASE, categories(upper, lower and integrated CASE tools)            Emerging software engineering trends – technology evolution, process trends, collaborative development, model-driven development, test-driven development , challenges of global software development</p>		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	Pressman, R.	Software Engineering: A Practitioner's Approach, 7th or 8th Ed	McGraw Hill
T2	Schwaber, K. and Beedle, M.	Agile Software Development with SCRUM, 1st Ed.	Pearson
T3	Lowdermilk, T.	User-Centered Design, 1st Ed.	O'Reilly Media
<b>Reference Books</b>			
R1	Vliet, H.	Software Engineering: Principles and Practice, 3rd Ed.	New Delhi: Wiley India Pvt Ltd
R2	Somerville, I.	Software Engineering, 9th Ed.	Pearson Education
R3	Mall, R.	Fundamentals of Software Engineering, 3rd Ed.	Prentice Hall India
R4	Cohn, M	Succeeding with Agile: Software Development Using Scrum	Pearson Education
R5			
<b>Self-Learning Facilities</b>	NPTEL Lecture Series		
<b>Web Resources</b>	<ol style="list-style-type: none"> <li>1. <a href="http://www.rspa.com/spi/index.html#webe">http://www.rspa.com/spi/index.html#webe</a></li> <li>2. <a href="http://www-itec.uni-klu.ac.at/~harald/proseminar/web11.pdf">http://www-itec.uni-klu.ac.at/~harald/proseminar/web11.pdf</a></li> <li>3. <a href="http://58.59.135.118:8081/BOOKS%5C026%5C21%5CHXYWPJH144310.pdf">http://58.59.135.118:8081/BOOKS%5C026%5C21%5CHXYWPJH144310.pdf</a></li> <li>4. <a href="http://www.scrum.org">www.scrum.org</a></li> </ol>		

**WEB ENGINEERING AND TECHNOLOGY**

<b>Course Title:</b>	<b>WEB ENGINEERING AND TECHNOLOGY</b>	<b>Course Number:</b>	<b>314445</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 3 Hrs/Week		<b>Laboratories:</b> 4 Hrs/Week	
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Oral Mock exam	Practical Mock Exam
	<b>Indirect Methods</b>	Assignments	Practicals
<b>Prerequisites</b>	Basic computer network and Database Management System		
<b>Introduction of Course</b>			
<b>WEB ENGINEERING AND TECHNOLOGY</b>			
<b>Course Objectives</b>			
1	To understand the concepts, principles, strategies, and methodologies of Web applications and Development.		
2	To apply current Web technologies to understand current Web business models.		
3	To understand and apply Web development processes.		
4	To understand the engineering aspect of web technology.		
<b>Course Outcomes :</b> Students will be able to			
CO1	Apply the concepts, principles and methods of Web engineering.		
CO2	Have a sufficient theoretical knowledge and analytical skills to develop Web applications.		
CO3	Apply the described concepts, principles and methods to development of complex Web Applications.		
CO4	Design and develop website using current Web technologies.		
CO5	Model, visualize and document the analysis and design of Web applications.		
<b>Course Contents</b>			
<b>Unit-I</b>	<b>WEB ENGINEERING INTRODUCTION</b>		
	Introduction - What is Web Engineering? Web Development History, Motivation, Categories of Web Applications, Web Applications Characteristics. Evolution and Need for Web Engineering, Web Engineering Models, Software Engineering v/s Web Engineering World Wide Web: Introduction to TCP/IP and WAP, DNS, Email, TelNet, HTTP and FTP. Introduction to Browser and search engines, Search fundamentals, Search strategies, Directories search engines and Meta search engines, Working of the search engines , Miscellaneous Web Browser details, Introduction to Web Servers: Features of web servers, caching, case study-IIS, Apache, Configuring web servers.		
	<b>Practical</b>		
	Configuring web servers: Apache.		
<b>Unit-II</b>	<b>BASIC MODELS AND ARCHITECTURES</b>		
	The role of the Information Architect, Collaboration and Communication, Organizing Information,		

	<p>Organizational Challenges, Organizing Web sites parameters and Intranets Creating Cohesive Websites: Conceptual Overview Website Development, Website Design issues, Conceptual Design, High-Level Design, Indexing the Right Stuff, Grouping Content. Architectural Page Mockups, Design Sketches, Navigation Systems. Searching Systems Good &amp; bad web design, Process of Web Publishing. Phases of Web Site development, enhancing your web-site, submission of website to search engines Web security issues, security audit of websites, Web effort estimation, Productivity, Measurement, Quality usability and reliability. Requirements Engineering for Web Applications: Introduction, Fundamentals, Requirement Source, Type, Notations Tools. Principles Requirements Engineering Activities, Adapting RE Methods to Web Application.</p> <p><b>Practical</b></p>
<b>Unit-III</b>	<b>TECHNOLOGIES FOR WEB APPLICATIONS</b>
	<p>Introduction HTML: HTML and DHTML, HTML Basic Concepts, Static and dynamic HTML, Structure of HTML documents, HTML Elements, Linking in HTML, Anchor Attributes, Image Maps, Meta Information, Image Preliminaries, Layouts, Backgrounds, Colors and Text, Fonts, Tables, Frames and layers, Audio and Video Support with HTML Database integration, CSS, Positioning with Style sheets, Forms Control, Form. Elements. Difference between HTML and HTML5</p> <p>Applying Styles, values, selectors, class, ids, inheritance, layout, backgrounds, borders, margin, padding, lists, fonts, text formatting, positioning, Dynamic HTML, DHTML with CSS.</p> <p><b>Practical</b></p> <ol style="list-style-type: none"> <li>1. Create table in HTML.</li> <li>2. Create a registration form using HTML form input elements viz. textbox, text area, radio button and drop down menu, check box, submit, file and reset button.</li> <li>3. Create a HTML programs using frames.</li> <li>4. Create a horizontal navigation bar in DIV using external CSS which contain home, about, gallery, enquiry, contacts menus. Also create the same bar in vertical alignment in another DIV in same page.</li> </ol>
<b>Unit-IV</b>	<b>JavaScript &amp; PHP MySQL ADMIN</b>
	<p>Introduction to JavaScript: Introduction to Scripting, Difference between Java and JavaScript, JavaScript Characteristics, JavaScript and Common Programming Concepts Introduction, Benefits Of Using PHP, MySQL, Server/Client Environment, Development Concept: How PHP Script Work, PHP Syntax, Embed PHP In HTML/HTML In PHP, PHP Data Types, Variable In PHP, Operator in PHP, Control Structure, Looping Structure, Function, File Inclusion: Include()/ Require(), Array, String Function: chr()/ strlen()/ strpos()/strcmp(), State Management : Cookies, Session management.</p> <p>Working with PHP My Admin: Types Data Type, Creating Database &amp; Tables, Dropping Database &amp; Tables, Adding Fields, Selecting Table MySQL Function in PHP : Database Connections, Managing Database Connections, Performing Queries, Closing Connection.</p> <p><b>Practical</b></p> <p><b>PHP:</b></p>



## Curriculum book of TE (IT)

	<p>1. Create a PHP program in which two values submitted using form and calculate its addition, subtraction, multiplication, modulation, average and division on the same page. Find the greatest number between them and square of each of them using PHP function.</p> <p>2. Write following program in PHP:</p> <ol style="list-style-type: none"> <li>Validating given email address</li> <li>Change background color based on hour of a day.</li> <li>Print Fibonacci Series.</li> </ol> <p><b>JavaScript:</b></p> <p>3. Write a Java script program to create a simple calculator.</p> <p>4. Write following Java script program:</p> <ol style="list-style-type: none"> <li>Create form validation program that checks the empty values from that form and alert back using alert function. Use at least 5 components.</li> <li>Display a live clock in Java script.</li> </ol> <p><b>PHP MYSQL Admin:</b></p> <ol style="list-style-type: none"> <li>Create feedback form in PHP which contains first name, last name, address, email, comment and mobile number and store that information in database. Also create a page which displays submitted feedbacks in tabular form.</li> <li>Create a login form using session handling in PHP. After successful login display name, address and other details in tabular format of logged user. Create 5 users. Also create a 'Logout' option .Store the data of user, login and password in the database.</li> <li>Create a simple address book in PHP using MySQL database which contains Adding new person with address (name, phone number, email, permanent address and temporary address etc.),updating their address, deleting him from record and view all records in table.</li> </ol>
<b>Unit- V</b>	<b>JAVA SERVLETS</b>
	<p>The J2EE and its multitier architecture, The enterprise application, Java Servlet and CGI, A simple servlet, life cycle of servlet, anatomy of servlet, javax.servlet package, Generic servlet, Http Servlet, reading data from client, HTTP request headers, sending data to client, HTTP response header, cookies, session management.</p> <p>Introduction of XML, Validation of XML documents, DTD, Ways to use XML, XML for data files, HTML VsXML, Embedding XML into HTML documents, Converting XML to HTML for Display, Displaying XML usingCSS and XSL, Rewriting HTML as XML, Relationship between HTML, SGML and XML, webpersonalization , Semantic web, Semantic Web Services, Ontology.</p> <p><b>Practical</b></p> <p><b>Java Servlet:</b></p> <ol style="list-style-type: none"> <li>Create two textboxes on the HTML page named login and password. After clicking on the 'login' button the servlet will be displayed. It will show 'login successful' upon correct Password else 'authentication failure' will be displayed. Make the use of HTTP Servlet or Generic Servlet.</li> <li>Write a program to demonstrate the use of servlet request and response as well as doGet ( ) and doPost ( ) methods. (The subject teacher should provide the appropriate problem statement for this).</li> </ol>
<b>Unit-VI</b>	<b>CONTENT MANAGEMENT SYSTEMS</b>
	<p>Introduction to CMS, advantages using CMS, CMS development tools: Wordpress, Drupal, Joomla.</p>



	<p>Wordpress: content and conversion, directory, file structure, local working, component administration, core, loop, data management, Wordpress as CMS, Wordpress in enterprise. Website Deployment: Domain registration, Domain hosting, parking websites, uploading data using FTP, email configuration. AJAX</p>		
	<b>Practical</b>		
	<b>WordPress</b>		
	Design a website using Content management system of WordPress. Make the use of different plugins and themes of the WordPress .		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	Ivan Bayross	Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP 4th Edition	BPB Publications.
T2	Jim Keogh	J2EE: The Complete Reference, 1st Edition	Tata McGraw Hill Publishing Company
T3	Jason Hunter	Java Servlet Programming 2nd Edition	O'reilly Publications.
T4	Brad Williams, David Damstra, Hal Stern	Professional WordPress: Design and Development	Wrox publications
<b>Reference Books</b>			
R1	Steven M. Schafer	HTML, XHTML and CSS, Fourth Edition .	Wiley India Edition
R2		Web Technologies Black Book: HTML, JavaScript, PHP, Java, JSP, XML and AJAX .	Kogent Learning Solutions Inc
R3	B. V. Kumar, S. Sangeetha, S. V. Subrahmanya	J2EE Architecture, an illustrative gateway to enterprise solutions: concept to Application Design and deployment	Tata McGraw Hill Publishing Company.
R4	Stephanie Leary	Wordpress for Web developers: An introduction to web professionals	Apress Publications.
<b>Web Resources</b>	<a href="http://www.w3schools.com">www.w3schools.com</a> <a href="http://www.cs.iit.edu">www.cs.iit.edu</a> /		
<b>Tutorials</b>	Assignment 1 on unit 1 Assignment 2 on unit 2		

# TE IT

## Semester II

**Design and Analysis of Algorithms**

PVG's COET, PUNE-9  
DEPARTMENT OF C  
**Curriculum book of TE (IT)**

**2016 -  
17**

<b>Course Title:</b>	Design and Analysis of Algorithms	<b>Course Number:</b>	314449
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 4 Hrs/Week	<b>Laboratories:</b> 2 Hrs/Week		
<b>Course Assessment Methods</b>	<b>Direct methods</b>	On-line/In-sem Examination: 50/30 Marks	Theory/End Semester Examination: 50/70 Marks
		Term-work	Practical/Oral
	<b>Indirect Methods</b>	Assignments, problem Solving	Objective test, Group Discussion
<b>Prerequisites</b>	Data Structures, Basic mathematics: Induction, probability theory and Basic searching and sorting algorithms		
<b>Introduction of Course</b>			
<b>Course Objectives</b>			
1	To know the basics of computational complexity analysis and various algorithm design paradigms.		
2	Provide students with solid foundations to deal with a wide variety of computational problems.		
3	To provide a thorough knowledge of the most common algorithms and data structures.		
4	To analyze a problem and identify the computing requirements appropriate for its solutions		
<b>Course Outcomes</b>			
CO1	Apply Knowledge of Mathematics to perform asymptotic analysis of algorithms		
CO2	Demonstrate a familiarity with major algorithms and data structures		
CO3	Apply important algorithmic design paradigms and methods of analysis		
CO4	Synthesize efficient algorithms in common engineering design situations		
<b>Course Contents</b>			
<b>Unit-I</b>	<b>INTRODUCTION</b>		
	Analysis of Algorithm, Efficiency- Analysis framework, asymptotic notations – big O, theta and omega. Analysis of Non-recursive and recursive algorithms, Amortized Analysis. Solving Recurrence Equations (Homogeneous and non-homogeneous) Proof Techniques: Minimum 2 examples of each: Contradiction, Mathematical Induction – Tiling Problem, Direct proofs, Proof by counterexample, Proof by contraposition.		
<b>Unit-II</b>	<b>DIVIDE AND CONQUER AND GREEDY</b>		
	Divide & Conquer: General method, Control abstraction, Merge sort, Quick Sort – Worst, Best and average case. Binary search, Large integer Multiplication, Strassen's Matrix multiplication. (for all above algorithms analysis to be done with recurrence) Greedy Method: General method and characteristics, Prim's method for MST , Kruskal method for MST (using $n \log n$ complexity), Dijkstra's Algorithm,		

	Huffman Trees ( $n \log n$ complexity), Fraction Knapsack problem, Job Sequencing		
	<b>Practical</b>		
	Write a program to implement matrix multiplication using Strassen's method. (Divide and Conquer)		
	Write a program to implement coding and decoding using Huffman method. (Greedy) Students are expected to write the program with $n \log n$ complexity and verify the same.		
<b>Unit-III</b>	<b>DYNAMIC PROGRAMMING</b>		
	General strategy, Principle of optimality, Warshal's and Floyd's Algorithm , Optimal Binary Search Trees, 0/1 knapsack Problem, Travelling Salesman Problem		
	<b>Practical</b>		
	Write a program to print shortest path and cost for the directed graph using Floyd and Warshal method. ( Dynamic Programming) and verify the complexity		
<b>Unit-IV</b>	<b>BACKTRACKING</b>		
	General method, Recursive backtracking algorithm, iterative backtracking method. 8- queens problem, Sum of subsets, Graph coloring, Hamiltonian Cycle , 0/1 Knapsack Problem		
	<b>Practical</b>		
	Write a recursive program to find the solution of placing n queens on chess board so that no queen takes each other (backtracking).		
	Write a non-recursive program to check whether Hamiltonian path exists in undirected graph or not. If exists print it. ( backtracking)		
<b>Unit- V</b>	<b>BRANCH AND BOUND</b>		
	The method, Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Traveling sales person problem		
	<b>Practical</b>		
	Write a program to solve the travelling salesman problem. Print the path and the cost. ( Branch and Bound)		
<b>Unit-VI</b>	<b>COMPUTATIONAL COMPLEXITY AND PARALLEL ALGORITHMS</b>		
	Non Deterministic algorithms, The classes P, NP, NP Complete, NP hard Proofs for NP Complete Problems: Clique, Vertex Cover Parallel Algorithms: Introduction, models for parallel computing, computing with complete binary tree, Pointer doubling algorithm		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	Horowitz and Sahani	Fundamentals of computer Algorithms	Galgotia. ISBN 81-7371-612-9
T2	R.C.T.Lee, S S Tseng, R C Chang, Y T Tsai	Introduction to Design and Analysis of Algorithms: A	Tata McGraw Hill. ISBN-13:978-1-25-902582-2. ISBN-10:1-25-902582-9

		Strategic approach	
T3	Gilles Brassard, Paul Bratle	Fundamentals of Algorithms	Pearson ISBN 978-81-317-1244-3
<b>Reference Books</b>			
R1	Thomas H Cormen and Charles E.L Leiserson	Introduction to Algorithm	PHI, ISBN:81-203-2141-3
R2	Anany Levitin	Levitin,"Introduction to the Design & Analysis of Algorithm	Pearson ISBN 81- 7758-835-4
R3	Steven S Skiena	The Algorithm Design Manual	Springer,2nd edition, ISBN 978-81-8489-865-1
R4	George T. Heineman, Gary Pollice, Stanley Selkow	Algorithms in a Nutshell, A Desktop Quick Reference	O'Reilly, ISBN 13:978-81-8404-608-3
<b>Self-Learning Facilities</b>			
NPTEL Lecture Series , Lecture series from IIT Kharagpur			
<b>Web Resources</b>			
<a href="http://www.cc.ntut.edu.tw/~cmliu/Alg/NTUT_Algorithm_S08g/index.htm">http://www.cc.ntut.edu.tw/~cmliu/Alg/NTUT_Algorithm_S08g/index.htm</a> <a href="http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html">http://www.personal.kent.edu/~rmuhamma/Algorithms/algorithm.html</a> weighted union and collapse find rule strassen's matrix multiplication <a href="http://www.geeksforgeeks.org/strassens-matrix-multiplication/">http://www.geeksforgeeks.org/strassens-matrix-multiplication/</a>			
<b>Assignments</b>			
1	Write a program to implement matrix multiplication using Strassen's method. (Divide and Conquer)		
2	Write a program to implement coding and decoding using Huffman method. (Greedy) Students are expected to write the program with nlogn complexity and verify the same.		
3	Write a program to print shortest path and cost for the directed graph using Floyd and Warshal method. ( Dynamic Programming) and verify the complexity		
4	Write a recursive program to find the solution of placing n queens on chess board so that no queen takes each other (backtracking).		
5	Write a non-recursive program to check whether Hamiltonian path exists in undirected graph or not. If exists print it. ( backtracking)		
	Write a program to solve the travelling salesman problem. Print the path and the cost. ( Branch and Bound)		

**SYSTEMS PROGRAMMING**

<b>Course Title:</b>	<b>SYSTEMS PROGRAMMING(314450)</b>	<b>Course Number:</b>	<b>(314450) 2012</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 4 Hrs/Week		<b>Laboratories:</b> 4 Hrs/Week	
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
	<b>Indirect Methods</b>	Assignments, Presentations, Class Test	Practical and Oral(SL-II) Seminars, Quiz, Q&A session, Group Discussion
<b>Prerequisites</b>	<ul style="list-style-type: none"> <li>Data Structures</li> <li>Computer Organization</li> <li>Microprocessors</li> <li>Basic searching and sorting Algorithms</li> </ul>		
<b>Course Objectives</b>			
1	1. To learn and understand fundamentals of System Software Programs as Assembler, Macroprocessor, Linkers and Loaders.		
2	2. To learn how to design and develop various System Software Programs		
3	3. To study phases of compiler in detail.		
<b>Course Outcomes</b>			
CO1	Design & implement System Programs as Assembler, Macroprocessor.		
CO2	Use tool Lex for generation of Lexical Analyzer.		
CO3	Use tool YACC for generation of Syntax Analyzer		
<b>Course Contents</b>			
<b>Unit-I</b>	<p><b>Introduction:</b> Components of System Software, Language Processing Activities, Fundamentals of Language Processing</p> <p><b>Assemblers:</b> Elements of Assembly Language Programming A simple Assembly Scheme, Pass structure of Assemblers, Design of Two Pass Assembler, Single pass assembler</p> <p><b>Macro Processor:</b> Macro Definition and call, Macro Expansion, Nested Macro Calls and definition, Advanced Macro Facilities, Design of Macro Processor</p>		
	<p><b>Practical:</b></p> <p>1. Write a program to implement II pass assembler. ( For hypothetical instruction set from Dhamdhere)</p> <p>a. Consider following cases only (Literal processing not expected)</p> <p>b. Forward references</p> <p>c. DS and DC statement</p> <p>d. START, EQU</p> <p>e. Error handling: symbol used but not defined, invalid instruction/register etc.</p>		

	<p>2. Write a program to implement II pass assembler. (For hypothetical instruction set from Dhamdhare). Consider Literal processing , forward references not expected</p> <p>a. Use of literals and not symbols b. LORG, END c. Error handling</p> <p>3. Write a program to implement Macro Processor. (AIF and AGO not expected, nested macro calls not expected) with Error Handling: Macro Duplicate definition, Parameter Mismatching etc.</p>
<b>Unit-II</b>	<p><b>LOADERS AND LINKERS</b> <b>Loaders:</b> Loader Schemes, Compile and Go, General Loader Scheme, Absolute Loader Scheme, Subroutine Linkages, Relocation and linking concepts, Self-relocating programs, Relocating Loaders, Direct Linking Loaders, Overlay Structure, Design of absolute and direct linking loader.</p>
<b>Unit-III</b>	<p><b>INTRODUCTION TO COMPILER</b> Phase structure of Compiler Lexical Analyzer: The Role of the Lexical Analyzer, Input Buffering. Specification of Tokens, Recognition of Tokens, Lexical Analyzer, RE to DFA Conversion, Lexical Errors. LEX: LEX Specification, Generation of Lexical Analyzer by LEX.</p> <p><b>Practical:</b> 4. Write a program to convert RE to DFA. (Compiler point of view, RE to DFA direct method from Uho Ulman Sethi).</p>
<b>Unit-IV</b>	<p><b>PARSERS</b> Role of parsers, Top down parsers, recursive descent parser, predictive parser, LL(K) parsers, Bottom up Parsers - Operator Precedence Parser, Shift Reduce - SLR, LR(K), LALR, Error Detection and Recovery in Parser, YACC specification and features Automatic construction of Parser(YACC).</p> <p><b>Practical:</b> 5. Write a program to implement calculator using LEX and YACC.</p>
<b>Unit- V</b>	<p><b>SEMANTIC ANALYSIS 8 Hours</b> Need, Syntax Directed Translation, Type Checking, Type conversion, Syntax Directed Definitions, Translation of assignment Statements, iterative statements, Boolean expressions, and conditional statements. Intermediate Code Formats: Postfix notation, Parse and syntax trees, Three address code, quadruples and triples. Quadruple generation for 1D and 2D arrays <b>Practical:</b> 6. Intermediate code generation using LEX &amp; YACC for Control Flow and Switch Case statements.</p>
<b>Unit-VI</b>	<p><b>STORAGE ALLOCATION, CODE GENERATION AND OPTIMIZATION</b> Storage allocation, code generation and Code optimization:</p>

	Storage Allocation: Run time storage organization, storage organization and allocation strategies. Code Generation: Code generation Issues, A Simple Code Generator Code Optimization: Machine Independent: Common Sub-expression elimination, Removing of loop invariants, Reduction in strengths. Machine dependent Issues: Assignment and use of registers, Rearrangement of Quadruples for code optimization.		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	D.M. Dhamdhere	,"Systems Programming and Operating Systems"	Tata McGraw-Hill, ISBN-13:978-0-07-463579-7
T2	Alfred V. Aho, Ravi Sethi, Reffrey D. Ullman,	"Compilers Principles, Techniques, and Tools"	Addison Wesley, ISBN 981-235-885-4
T3	John J Donovan	"Systems Programming"	Tata McGraw-Hill Edition 1991, ISBN 0-07-460482-1
<b>Reference Books</b>			
R1	Leland L. Beck,	"System Software An Introduction to Systems Programming" 3rd Edition,	Person Education, ISBN 81-7808-036-2
R2	Adam Hoover	,"System Programming with C and Unix",	Pearson,2010
R3	Terence Parr,	"Language Implementation Patterns",	SPD,2009
<b>Self-Learning Facilities</b>	Books		
<b>Contents beyond Syllabus</b>	Implementation Of Lexical Analyzer		
<b>Additional Experiments</b>	Lexical Analyzer		



**OPERATING SYSTEM**

<b>Course Title:</b>	<b>OPERATING SYSTEM</b>	<b>Course Number:</b>	<b>314451</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b>	4 Hrs/Week	<b>Laboratories:</b>	4 Hrs/Week
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Oral Mock exam	Practical Mock Exam
	<b>Indirect Methods</b>	Assignments	Practicals
<b>Prerequisites</b>	Computer Organization and Data Structure		

**Introduction of Course**

**OPERATING SYSTEM**

**Course Objectives**

1	To Introduce basic concepts and functions of modern operating systems.
2	To Understand the concept of process, and thread management.
3	To Understand how the resources are scheduled and managed.
4	To Understand the concepts of process synchronization and deadlock.
5	To know the concept of I/O and File management.
6	To Understand various Memory management techniques.
7	To be aware of latest trends in Operating Systems.

**Course Outcomes :** Students will be able to

CO1	Possess knowledge of the role of Operating Systems and their types.
CO2	Apply the concept of a process, thread and scheduling algorithms.
CO3	Apply the concepts of process synchronization and how it is achieved.
CO4	Realize the concept of deadlock and different ways to handle it.
CO5	Realize various memory management techniques.
CO6	Realize the concept of I/O management and File system.

**Course Contents**

<b>Unit-I</b>	<b>OPERATING SYSTEM OVERVIEW</b>
	Operating System Objectives and Functions, The Evolution of Operating Systems, Developments Leading to Modern Operating Systems, Virtual Machines OS Design Considerations for Multiprocessor and Multicore architectures, Microsoft Windows Overview, Modern UNIX Systems, Linux, Android. Booting Process of all the above operating systems.
<b>Unit-II</b>	<b>PROCESS DESCRIPTION AND CONTROL</b>
	Process: Concept of a Process, Process States, Process Description, Process Control (Process creation, Waiting for the process/processes, Loading programs into processes and Process Termination), Execution of the Operating System. Threads: Processes and Threads, Concept of Multithreading, Types of Threads, Thread programming Using pthreads, Multicore processors and threads , Linux Process and Thread Management, Android Process and Thread Management Scheduling: Uniprocessor Scheduling - Types of Scheduling, Scheduling Algorithms, and Thread Scheduling, An introduction to Multiprocessor and Real-Time Scheduling, Traditional UNIX Scheduling, Linux Scheduling.

## Curriculum book of TE (IT)

	<p><b>Practical</b></p> <p>(1) Process control system calls: The demonstration of fork, execve and wait system calls along with zombie and orphan states.</p> <p>(2) Thread management using pthread library.</p> <p>(3) Implementing a CPU scheduling policy in a Linux OS.</p>
<b>Unit-III</b>	<b>CONCURRENCY: MUTUAL EXCLUSION AND SYNCHRONIZATION</b>
	<p>Concurrency: Process/thread Synchronization and Mutual Exclusion Principles of Concurrency, Requirements for Mutual Exclusion, Mutual Exclusion: Hardware Support, Operating System Support (Semaphores and Mutex), Programming Language Support (Monitors), Classical synchronization problems: Readers/Writers Problem, Producer and Consumer problem.</p> <p>Concurrency : Deadlock and Starvation Principles of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock detection, An Integrated Deadlock Strategy, Example: Dining Philosophers Problem, Linux inter-process communication and concurrency mechanisms, Android Inter-process communication mechanisms and concurrency mechanisms</p>
	<p><b>Practical</b></p> <p>(1).Thread synchronization using counting semaphores and mutual exclusion using mutex. Application to demonstrate: producer-consumer problem with counting semaphores and mutex.</p> <p>(2).Deadlock Avoidance Using Semaphores: Implement the deadlock-free solution to Dining Philosophers problem to illustrate the problem of deadlock and/or starvation that can occur when many synchronized threads are competing for limited resources.</p> <p>(3) Inter process communication in Linux using PIPE, FIFO and SIGNALS.</p>
<b>Unit-IV</b>	<b>MEMORY MANAGEMENT</b>
	<p>Memory Management: Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Buddy System, Relocation, Paging, Segmentation. Virtual Memory: Hardware and Control Structures, Operating System Software, Linux Memory Management, Windows Memory Management, Android Memory Management.</p>
	<p><b>Practical</b></p> <p>Implementing a memory management policy in a Linux OS.</p>
<b>Unit- V</b>	<b>INPUT/OUTPUT AND FILES</b>
	<p>I/O Management and Disk Scheduling: I/O Devices, Organization of the I/O Function, Operating System Design Issues, I/O Buffering, Disk Scheduling, Disk Cache, Linux I/O. File Management: Overview, File Organization and Access, File Directories, File Sharing, Record Blocking, Secondary Storage Management, Linux Virtual File System, Android File Management.</p>
	<p><b>Practical</b></p> <p>Implementing a file system in a Linux OS</p>
<b>Unit-VI</b>	<b>RECENT AND FUTURE TRENDS IN OS</b>
	<p>Linux Kernel Module Programming, Embedded Operating Systems: Characteristics of Embedded Systems, Embedded Linux, and Application specific OS. Basic services of NACH Operating System. Introduction to Service Oriented Operating System (SOOS), Introduction to Ubuntu EDGE OS, etc.</p>

	<b>Practical</b>		
	<p>(1) Linux Kernel configuration, compilation and rebooting from the newly compiled kernel.</p> <p>(2) Kernel space programming: Implement and add a loadable kernel module to Linux kernel, demonstrate using insmod, lsmod and rmmod commands. A sample kernel space program should print the "Hello World" while loading the kernel module and "Goodbye World" while unloading the kernel module.</p> <p>(3) Implement a new system call, add this new system call in the Linux kernel (any kernel source, any architecture and any Linux kernel distribution) and demonstrate the use of same.</p>		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	William Stallings	Operating System: Internals and Design Principles, 8th Edition.	Prentice Hall
T2	Abraham Silberschatz, Peter Baer Galvin and Greg Gagne	Operating System Concepts, 9th Edition.	John Wiley & Sons
T3	Maurice J. Bach	Design of UNIX Operating System	PHI
<b>Reference Books</b>			
R1	Tom Adelstein and Bill Lubanovic	Linux System Administration, Inc., 1st Edition, 2007	O'Reilly Media
R2	Harvey M. Deitel	Operating Systems, , 3rd Edition.	Prentice Hall
R3	Andrew S. Tanenbaum	Modern Operating System, , 3rd Edition.	Prentice Hall
R4	Thomson	Operating System in depth by Thomson	Thomson.
<b>Tutorials</b>	Assignment 1 on unit 1		
	Assignment 2 on unit 2		

**Multimedia Technologies**

<b>Course Title:</b>	<b>Multimedia Technologies</b>	<b>Course Number:</b>	<b>314452</b>
<b>Designation of Course</b>	Professional Application		
<b>Teaching Scheme:</b> 3 Hrs/Week	<b>Laboratories:</b> --		
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Term-work : NIL	Practical/Oral : NIL
	<b>Indirect Methods</b>	Assignments, Presentations	Seminars, Quiz, Q&A session, Group Discussion
<b>Prerequisites</b>	Data Structures and Files and Basics of computer graphics and animation		
<b>Introduction of Course</b>			
<b>Course Objectives</b>			
1	To learn 5 basic components of multimedia (text, image, audio, video and animation)		
2	To learn the advance graphics		
3	To learn compression techniques for various multimedia components		
4	To learn Gaming and animation		
<b>Course Outcomes</b>			
CO1	At the end of this course students will be able to to create their own file formats for specific application		
CO2	To do some projects based on current trends in multimedia		
CO3	To use of open sources for authoring tool for animation and presentations		
CO4	To develop simple games and animation		
<b>Course Contents</b>			
<b>Unit-I</b>	<b>INTRODUCTION TO MULTIMEDIA</b>		
	Goals, objectives, and characteristics of multimedia, Multimedia building blocks, Multimedia architecture, hardware support Distributed multimedia applications, streaming technologies, multimedia database systems Multimedia authoring tools, overview of multimedia software tools, Multimedia Applications Media Entertainment, Media consumption, web-based applications, e-learning and education Text: Types of text, Text compression: Huffman coding, LZ & LZW Text file formats: TXT, DOC; RTF, PDF, PS		
<b>Unit-II</b>	<b>DIGITAL IMAGE</b>		
	Basic Image fundamentals, image File formats - (BMP, TIFF, JPEG, GIF) Image acquisition, storage processing, Communication, and display Image Enhancement: Enhancement by point processing, Spatial filtering Image		

	Compression: Types of Compression: Lossy & Lossless, Symmetrical & Asymmetrical, Intra-frame & Inter-frame Hybrid JPEG, Lossless: RLE, Shannon - Fano algorithm, Arithmetic coding. Lossy: Vector quantization, Fractal Compression Technique, Transform Coding, Psycho-analysis, and inter-frame Correlation. Hybrid: JPEG-DCT		
<b>Unit-III</b>	<b>AUDIO AND AUDIO COMPRESSION</b>		
	Nature of sound waves, characteristics of sound waves, psycho-acoustic, MIDI, digital audio, CD formats. Audio file formats: WAV, AIFF, VOC, AVI, MPEG Audio File formats, RMF, WMA Audio compression techniques : DM, ADPCM and MPEG Audio file conversions Multimedia Supported audio formats in Android, Media Playback.		
<b>Unit-IV</b>	<b>VIDEO</b>		
	Video signal formats, Video transmission standards: EDTV, CCIR, CIF, SIF, HDTV, digitization of video Video file formats: MOV, Real Video, H-261, H-263, Cinepack, Nerodigital, Video editing, DVD formats, MPEG, Video streaming Multimedia Supported video formats in Android, Media Playback		
<b>Unit- V</b>	<b>ANIMATION AND OPENGL</b>		
	Animation: Basics of animation, types of animation, principles of animation, techniques of animation, Creating animation OpenGL: Open GL over windows/Linux, Extension, programming languages, SDK, shadowing techniques, rendering		
<b>Unit-VI</b>	<b>ADVANCES IN MULTIMEDIA</b>		
	Virtual Reality : Concept, Forms of VR, VR applications, VR devices: Hand Gloves, Head mounted tracking system, VR chair, CCD, VCR, 3D Sound system, Head mounted display Synchronization: Multimedia Communication and applications, Study of Multimedia networking, Quality of data transmission, Multimedia over IP, Media on Demand Multimedia in Android: Android Multimedia Framework Architecture, GStreamer :Introduction, GStreamer Based Multimedia Framework, Open Core MultimediaEngine		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	Ralf Steinmetz and Klara Nahrstedt	Multimedia Computing, Communication and Applications	Pearson Education
T2	K;R; Rao	Multimedia Communication Systems: Techniques, Standards, and Networks	TMH
T3	Ranjan Parekh	Principles of Multimedia	Tata McGraw-Hill
<b>Reference Books</b>			
R1	Ashok Banerji, Inanda Ghosh	Multimedia Technologies	
R2	Gonzalez, Woods	Digital Image Processing	Addison Wesley

R3	Ze-Nian Li, Marks S. Drew	Fundamentals of Multimedia	Pearson Education
R4	Edward !ngel	OpenGL: ! Primer	Addison-Wesley
R5	Parag Havaladar, Gerard Medioni	Multimedia Systems	Cengage Learning
<b>Self-Learning Facilities</b>	NPTEL Lecture Series		
<b>Web Resources</b>	<ul style="list-style-type: none"> <li>• Programming with OpenGL : Tom McRenolds Silicon Graphics Computer Systems</li> <li>• Multimedia Power point presentations for audio, image compression etc.</li> </ul>		

**Information Technology Project Management**

<b>Course Title:</b>	<b>Information Technology Project Management</b>	<b>Course Number:</b>	<b>314453</b>
<b>Designation of Course</b>	Professional Core		
<b>Teaching Scheme:</b> 3 Hrs/Week		<b>Laboratories:</b> --	
<b>Course Assessment Methods</b>	<b>Direct methods</b>	In-sem Examination: 30 Marks	Theory/End Semester Examination: 70 Marks
		Term-work ---	Practical/Oral ---
	<b>Indirect Methods</b>	Class Room Exam	
<b>Course Objectives</b>			
1	Explain the need for Engineers in Management .		
2	To understand basic managerial and behavioural traits essential in industry.		
3	To understand project management through life cycle of the project.		
4	To understand project planning, execution, tracking, audit and closure of project.		
5	To understand basics of ERP systems and its deployment.		
6	To understand current and future trends and practices in the IT industry.		
<b>Course Outcomes</b>			
CO1	Students will learn and understand basic essential managerial qualities.		
CO2	They can understand importance of engineer's role in management.		
CO3	They will understand IT project management through life cycle of the project.		
CO4	They will learn about project planning, execution, tracking, audit and closure of project.		
CO5	They can understand processes in different departments of IT and non-IT industries		
CO6	They can understand current technologies and future trends in IT Project Management.		
<b>Course Contents</b>			
<b>Unit-I</b>	<b>MANAGEMENT BASICS AND OVERVIEW OF PROJECT MANAGEMENT</b>		
	Management concepts, levels of management, functions of management- POSDCORB, Management approaches by Taylor and Fayol		
	SWOT analysis, Importance of Strategy, Project concepts, proposal and contract, project management lifecycle, Project stake holders, project processes, groups and interactions		
	Case studies		
<b>Unit-II</b>	<b>PRODUCT MANAGEMENT AND SCHEDULING</b>		
	Project scope management, Time management: Activity planning, duration estimation, critical path finding.		
	Cost Estimation and management, Effort estimation, PERT, CPM, Case studies		
<b>Unit-III</b>	<b>HR AND COMMUNICATION MANAGEMENT</b>		
	Project Team: Roles and responsibilities of individuals in IT Project, Business ethics		
	Role of Project Manager: Motivation, Team development and team building, Conflict management, negotiation, decision making		

	Communication planning, information distribution, performance reporting, administrative closure, Case studies		
<b>Unit-IV</b>	<b>PROJECT EXECUTION AND RISK MANAGEMENT</b>		
	Project monitoring and control: tools for project management, Software tools like Microsoft project management or any other open source tool,		
	Risk management planning, Risk identification, Qualitative and quantitative analysis, Risk response planning, Risk monitoring and control, Case studies		
<b>Unit- V</b>	<b>QUALITY MANAGEMENT</b>		
	Quality planning, Quality assurance, Quality control, Project auditing, ISO 9000, CMM I, Six sigma, Benchmarking, Case studies		
<b>Unit-VI</b>	<b>PROJECT MANAGEMENT RECENT TRENDS</b>		
	SCM, CRM, ERP: Basic concepts, Advantages and limitations, SAP		
	IP : Patent and copyright, Agile software development, Business process reengineering, International Project Management, Case studies		
<b>Text Books</b>	<b>Author</b>	<b>Title of Book</b>	<b>Publication</b>
T1	E book	A Guide to Project management	Project Management Institute
T2	Jack R Meredith	Project Management A managerial Approach	Wiley & Sons
<b>Reference Books</b>			
R1	P.C. Tripathi	Principles of Management	Tata McGraw Hill Education Private Limited
R2	John Nichololas	Project Management for Business, Engineering, and Technology	Butterworth-Heinemann an imprint of Elsevier
<b>Self-Learning Facilities</b>	NPTEL Lecture Series		