SYLLABUS

THIRD YEAR (COMPUTER ENGINEERING)
COURSE 2012
(EFFECTIVE FROM JUNE 2014)

University of Pune

Course Structure for TE Computer Engineering

2012 Course (w.e.f. June 2014)

Subject Code	Subject	Teaching Scheme Hrs/Week			Examination Scheme					Mark
		Lect.	Tutorials	Pract	In-Semester Assessment	Tw	Pr	OR	End Semester Exam	Total
SEM – I										
310241	Theory of Computation	4	—	—	30	—	—	—	70	100
310242	Operating Systems Design	4	—	—	30	—	—	—	70	100
310243	Data Communication and Wireless Sensor Networks	4	—	—	30	—	—	—	70	100
310244	Database Management Systems Applications	3	—	—	30	—	—	—	70	100
310245	Computer Forensic and Cyber Applications	3	—	—	30	—	—	—	70	100
310246	Programming Lab-I	—	—	4	—		50	50	—	100
310247	Programming Lab-II	—	—	4	—	50	—	50	—	100
310248	Employability Skills Development Lab	—	—	2	—	50	—	—	—	50
	Total of Semester – I	**18**	—	**10**	**150**	**100**	**50**	**100**	**350**	**750**

Course Structure for TE Computer Engineering

2012 Course (w.e.f. June 2014)

SEM – II										
Subject Code	**Subject**	**Teaching Scheme Hrs/Week**			**Examination Scheme**					**Mark**
		Lect.	**Tutorials**	**Pract**	**In-Semester Assessment**	**Tw**	**Pr**	**Or**	**End Semester Assessment**	**Total**
310249	Principles of Concurrent and Distributed Programming	4	—	—	30	—	—	—	70	100
310250	Embedded Operating Systems	4	—	—	30	—	—	—	70	100
310251	Computer Networks	4	—	—	30	—	—	—	70	100
310252	Software Engineering	3	—	—	30	—	—	—	70	100
310253	Digital Signal Processing Applications	3	—	—	30	—	—	—	70	100
310254	Programming Laboratory-III	—	—	4	—	—	50	50	—	100
310255	Programming Laboratory-IV	—	—	4	—	50	—	50	—	100
310256	Seminar and Technical Communication Laboratory	—	—	2	—	50	—	—	—	50
	Total of Semester – II	**18**	—	**10**	**150**	**100**	**50**	**100**	**350**	**750**

310241 Theory of Computation

Teaching Scheme

Theory: 4 Hrs/ Week

Examination Scheme:
In Semester Assessment: 30 Marks
End Semester Assessment: 70 Marks

<table>
<tr><td colspan="3"><b>Course Objectives:</b>
• To learn formal Programming Language Theory
• To learn Grammar and Turing Machine Designing</td></tr>
<tr><td colspan="3"><b>Course Outcomes:</b>
• Ability to subdivide problems space based on input subdivision using constraints, grammar
• Ability to design deterministic turing machine for all input all output , NP Complete
• Ability to design non deterministic turing machine for all input all output, NP Hard</td></tr>
<tr><td>Unit I</td><td><b>Basic Concepts and Formal Language theory:</b>
Languages in abstract, Defining languages, Klenne closure, Symbol/alphabets, string/word, Formal
Introduction, mathematical foundation. Mathematical Formal Language Theory Representation for Formal Languages: Sets, Logic, Functions, Relations, Graphs,Proof Techniques-Formal Proofs, Inductive Proofs, Strings & Languages, examples, Basic Machine: Functionality and Limitations.
Importance of Automata Theory. Automata, Automata- Formal Definition & Designing Finite Automata examples, Simplified Notation, Nondeterminism-Formal Definition & Designing Nondeterministic Finite Automata, Computability & Complexity, Pattern Matching. Language Acceptor: Concept, Machine as a language acceptor, example, Machine as a string processor. Finite Automata- Formal Definition & Designing Finite Automata –basic examples, Simplified Notation.
<b>Regular Expressions and Languages:</b>
Recursive definition of regular expression, regular set, identities of regular expressions, regular expressions, examples and FA. Equivalence of R.E.-examples. Identity Rules And Algebraic laws for R.E.
Regular languages and examples. Pumping lemma for regular languages.
Limitations of R.E.</td><td>8 hrs</td></tr>
<tr><td>Unit II</td><td><b>Deterministic and Non deterministic Finite Automata:</b> DFA: Definition and description of DFA, Transition Function of a DFA, NFA: Definition and description of DFA, Transition Function of a NFA, Є-NFA: Definition and description of NFA, Transition Function of a NFA, Language acceptance by a FA(NFA , DFA) and string acceptance, Conversion of NFA with Є to NFA without Є, Conversion of NFA without Є to DFA, Conversion of NFA with Є to DFA (direct method and subset construction method), Minimization of a DFA. Inter-conversion RE and FA:
Construction of FA equivalent to RE using state loop elimination method. Construction of FA equivalent to RE using Andrsen’s Theorem. Construction of RE equivalent to FA(RE to Є-NFA, Є-NFA to DFA).
FA with output: Moore and Mealy machines -Definition, models, inter-conversion.Pumping Lemma for Regular languages, Properties of Regular Languages and FA: Closure and Decision properties, Limitations of FA.</td><td>8 Hrs</td></tr>
<tr><td>Unit III</td><td><b>Grammar:</b> Grammar- Definition, representation of grammar, Chomsky hierarchy, Context Free Grammar- Definition, Derivation, sentential form, parse tree, inference, derivation, parse tree, ambiguity in grammar and language- ambiguous Grammar and Push Down grammar, removing ambiguity from grammar, Normal Forms- Chomsky normal form, Greibach normal form, Closure properties of CFL, Decision property of</td><td>6 Hrs</td></tr>
</table>

	CFL. Automata Regular grammar- Definition, left linear, right linear grammar, FA to RG and RG to FA, Application of grammar. ambiguous grammar. Recursive, Recursively Enumerable Languages: A Language that is not recursively enumerable, An Undecidable Problem that is RE Recursive Languages, The Universal Language. Introduction to concurrent grammar. Concurrent Grammar, Formal methods in concurrency,Graph Grammar, Aspect of Concurrency in Graph Grammar, set theoretic approaches to Graph Grammar, Graph Grammar for parallel computation	
Unit IV	**Turing machines:** Turing machines (TMs): TM Model and conventions, Formal Definition, TM Instantaneous Description (ID), Transition Function, Languages of TM, Equivalence of final state and halting state (TM and halting), TM and Computers: Simulating a TM by computer, Simulating a computer by TM, Types of TM: Deterministic Turing Machines (DTM) and Non-deterministic Turing Machines (NTM), Extension to Basic TM: TM with Multiple tracks, Multitape TMs, Universal TM (UTM), Church-Turing hypothesis , Post Machines: Introduction to Post Machines (PMs), Comparison between FA, PDA, PM and TM Concurrency and parallel machines considerations while designing Turing Machine. Examples of Concurrent Turing Machines. problem in TM: Undecidable problems about Turing Machines, Reduction, Post Correspondence Problem(PCP, NPCP) : Definition Modified PCP, Other Undecidable Problems, Non-deterministic Turing Machine.	8 Hrs
Unit V	**Push Down Automata**- Definition, Notation, acceptance by final state, acceptance by empty stack, Equivalence of PDA and CFG- Grammar to PDA, PDA to Grammar, Deterministic PDA and Non Deterministic PDA. Parsing and PDA. Application of PDA. Non deterministic PDA (NPDA). Introduction to Post Machines (PMs)	4 Hrs
Unit VI	**Tractable & Intractable:** The Classes P and NP : Problems Solvable in Polynomial Time, An Example: Kruskal's Algorithm, Nondeterministic Polynomial Time, An NP Example: The Traveling Salesman Problem, Polynomial-Time Reductions NP-Complete Problems, An NP-Complete Problem: The Satisfiability Problem, Tractable and Intractable Representing SAT Instances, NP-Completeness of the SAT Problem, A Restricted Satisfiability Problem: Normal Forms for Boolean Expressions, Converting Expressions to CNF,.The Problem of Independent Sets, The Node-Cover Problem, The Directed Hamilton-Circuit Problem, Undirected Hamilton Circuits.	6 Hrs
Text Books		
1."Introduction to Automata Theory Languages And Computation” By John E. Hopcoroft, Rajeev Motwani, Jeffrey D-Ullman, LPE		
2.“Theory Of Computer Science “ By K.L.P. Mishra & Chandrashekharan		
3.“Introduction to The Theory of Computation” By Michael Sipser, ISE		
Reference Book		
1.“Introduction to Languages of The Theory Of Computation” By Martin		
2. “Theory of Computation”, Vivek Kulkarni, Oxford University Press, ISBN 0-19-808458-7		
3. “Introduction to Languages and Theory of Computation”, John Martin McGrawHill, 978-0-07-066048-9		
Digital Content: Content Developed by BoS		

310242 Operating Systems Design

Theory: 4 Hrs/ Week **Examination Scheme:**

In Semester Assessment: 30 Marks

End Semester Assessment: 70 Marks

Course Objectives: • To learn the Operating System Booting Process • To learn advance file system and operating system management • To learn *init()* process and other essential boot processes • **To learn use of GRUB2**		
Course Outcomes: • Ability to use EFI based x64 Operating Systems • Ability to use x64 based File Systems and Managers		
Unit I	**Foundation of Unix Operating Systems:** what is kernel, Types of kernel(monolithic,micro,exo),Operating system booting process Grub-I, Grub-II, Buffer management in Unix/Linux,Buffer Cache, Internal Representation of Files, Systems calls for the files systems,File management, File Concept , Access methods, Free Space Management, Disk management, Swap space.	8 Hrs
Unit II	**Process & Threads:** The Structure of Processes: Process States and Transitions, Layout of System Memory, The context of a Process, Saving the Context of a Process, Manipulation of the Process Address space, Sleep, Process creation,Signals,Process termination, A waiting process termination, Invoking other programs,The user ID of a process, Changing the size of a process, The shell, Basic Shell script Programming. System boot and the *init()* process Process scheduling, System calls for time,Clock Threads,Concept of threads, Linux processes & thread management, Introduction to threads(advantages of threads, Implementation of threads) Multithreading models (M to 1, 1:1:1, MM model), Threading issues(Fork() and exec() system calls, Thread Cancellation ,thread specific data). Deadlocks: Resources, Principles of deadlock, The ostrich algorithm, Detection and Recovery,Prevention,Avoidance. Banker's Algorithm. Init Process in Android Mobiles(Latest Android Version).	10 Hrs
Unit III	**Memory management and virtual memory:** Swapping,Demand paging, a hybrid System with swapping and demand paging, memory management requirements,Memory partitioning, Paging, Segmentation, Security Issues, Hardware and control structures, Operating system software, Linux memory management,Windows8 memory management, Android Memory Management	6 Hrs
Unit IV	**Inter-process Communication:** Process Tracing, system V IPC, Network communications, pipes, Sockets, Multiprocessor systems: Problem of Multiprocessor Systems, Solution with Master and Slave Processors, Solution with Semaphores, The Tunis System, Performance Limitations	6 Hrs
Unit V	**Advance Tools and Technologies:** Make tool: *make, nmake,cmake,* AWK tools, *grep egrep, fgrep,*sorting tool, UEFI Boot,UBoot, USB BOOT, Case study of Fedora-19 EFI files: *gcdx64.efi, grub.cfg, grubx64.efi, MokManager.efi, shim.efi, shim-fedora.efi*	6 Hrs

310242 Operating Systems Design

Teaching Scheme: **Examination Scheme:**
Theory: 4Hrs/Week **In Semester : 30 Marks**
End Semester Assessment: 70 Marks

Unit VI	**Advance Tools and Technologies(and problem solving in the OS)** Multiptocessor scheduling,Real time scheduling,Linux scheduling, UNIX free BSD scheduling,Windows vista scheduling,Windows-8 ReFS, Embedded System, Handheld Operating systems: Handheld systems-Requirements, Technology Overview, PalmOS, Microsoft Windows CE, Microsoft Windows Mobile, Google Android, Other Handheld Operating systems, Securing Handheld systems, Frame of Reference for Handheld Systems.	6 Hrs

Text Books

1. Pramod Chandra P. Bhatt "An Introduction to Operating Systems: Concepts and Practices", 4th edition, PHI, ISBN 978-81-203-4836-3
2. Maurice J.Bach "The Design of The Unix Operating system", PHI, ISBN 978-81-203-0516-8

Reference Books:

1. Evi Nemeth, Garth Snyder, Tren Hein, Ben Whaley, Unix and Linux System Administration Handbook, Fourth Edition, ISBN: 978-81-317-6177-9, 2011
2. Dhanajay M. Dhamdhere, "Operating Systems: A Concept Based Approach", 3rd Edition, McGrawHill Education, ISBN-13: 978-1-25-900558-9, ISBN-10: 1-25-900558-5

Digital Content:

1. MSDN 2013 versions
2. Fedora19 Development
3. Content Developed by the BoS

310243 Data Communication and Wireless Sensor Networks

Teaching Scheme: **Examination Scheme:**

Theory: 4 Hrs/Week **In Semester Assessment: 30 Marks**

End Semester Assessment: 70 Marks

<table>
<tr><td colspan="3"><b>Course Objectives:</b>
• To learn Data Communication Methods and Algorithm
• To learn setup, installation,configuration of WSN
• To study different programming tools</td></tr>
<tr><td colspan="3"><b>Course Outcomes:</b>
• Ability to program using data communication methods and algorithm
• Ability to setup, configure and program WSN
• Ability to use different programming application for WSN, BIGDATA</td></tr>
<tr><td>Unit I</td><td><b>Basics of communication:</b>
Introduction, Mathematical Foundation, Basic Problem Solving, Communication System, Baseband, Broadband and Carrier Communication, transmission modes, Baud rate, bit rate, SNR, Channel Bandwidth and rate of communication.
Digital Modulation Techniques: PCM, PCM Encoder and Decoder, DPCM, ADPCM, Delta modulation,Adaptive Delta Modulation, RS-232C,Bandwidth requirement of digital modulation techniques, quantization noise Line Coding techniques: Bipolar, Unipolar, RZ, NRZ, Manchester, AMI, B8ZS, Block coding techniques.
Multiplexing techniques: TDM, FDM, WDM, and CDMA, TD-SCDMA, LTE-TDD, LTE-FDD,. LAN standards: Ethernet, Wireless LAN, WiMax, ZigBee, Bluetooth,Infrastructure based (satellite n/w, Cellular n/w) and Infrastructure less (Adhoc n/w) wireless topologies. VLAN, Basics of VPN, VPN tools and Applications.</td><td>10 Hrs</td></tr>
<tr><td>Unit II</td><td><b>Overview of Wireless Networks:</b>
Wireless Transmission: Electromagnetic Spectrum, Radio, Micro Waves, Infrared, Lightwave, Spread Spectrum Systems, modem Switching Techniques: Circuit Switching, Packet Switching and Message Switching, Hardware Components: Transceivers, Access Points and wireless routers, Data link layer design issues: Services, Framing, Error and flow control, Stop-and-Wait protocol, Sliding Window protocol, Medium access control sub layer, Channel allocation: Static and Dynamic allocation, Multiple Access Protocols: ALOHA, CSMA, CSMA/CD , CSMA/CA</td><td>8Hrs</td></tr>
<tr><td>Unit III</td><td><b>Basic Concepts of WSN:</b> Background of Sensor Network Technology, Applications: Building Automation, Sensors and Robots, Health Care and Military Applications. WSN Architecture: Sensor Type and Technology, Sensor Network Organization and Tracking. RFID based data communication, Architecture,</td><td>4 Hrs</td></tr>
<tr><td>Unit IV</td><td><b>Data link layer protocols</b>
Link Layer: Error control, Framing, Link management.
MAC Layer: Low duty cycle protocols and wakeup concepts, Contention-based protocols, Schedule-based protocols, Networking Sensors: ZigBee, Sensor MAC(S-MAC) protocol for WSN, Naming and Addressing: Fundamentals, Address and name management in wireless sensor networks</td><td>6Hrs</td></tr>
<tr><td>Unit V</td><td><b>Routing Protocols for WSN</b>
Data Dissemination and Gathering, Routing Challenges and Design Issues in WSN, Routing Strategies (Proactive and Reactive) in WSN. Geographic and Energy aware routing, Attribute based routing,Routing Techniques: Flooding, SPIN, Infrastructure Establishment:Topology Control, Clustering, time</td><td>6 Hrs</td></tr>
</table>

	synchronization, localization and services, Low energy adaptive Clustering, Power efficient gathering in sensor information system,case study of Data Communication from HDMI Camera in host and accessory modes, PICONET, Tunneling Protocols.	
Unit VI	**Infrastructure Establishment for WSN** Localization and Positioning, tracking: Properties of positioning, Possible approaches, Task driven Sensing, Rolls of Sensor nodes and utilities, Information based sensor tracking, joint routing and information aggregation, Sensor Network Databases-BIGDATA, Sensor network platforms and tools, Single-hop localization, Positioning in multi-hop environments,Impact of anchor placement, Operating Systems for WSN: OS Design Issues, Examples of OS(Architecture, Design Issues, Functions): Tiny OS, Mate, Magnet OS, MANTIS, Nano-RK OS Architecture Block Diagram, LiteOS Architectural Block Diagram,LiteFS Architectural Block Diagram, Content delivery networks. Introduction to Internet of Things(IoT)	6 Hrs

Text Books:

1. Kurose, Ross “Computer Networking a Top Down Approach Featuring the Internet”, Pearson; 6th edition (March 5, 2012), ISBN-10: 0132856204,ISBN-13: 978-0132856201 (pdf available)
2. Fang Zhaho, Leonidas Guibas, “Wireless Sensor Networks: An information Processing Approach”, Elsevier ISBN: 978-81-8147-642-5

Reference Books:

1. Dipankar Raychaudhari, Mario Cerla,”Emerging Wireless Technologies and the Future Mobile Internet”, Cambridge University Press, ISBN-13: 978-1-107-67864-4(Paperback)
2. Lyla B. Das, “Embedded Systems: An Integrated Approach” Pearson, ISBN:978-81-317-8766-3
3. Kazim Sohraby, Daniel Minoli, Taieb Znati, “Wireless Sensor Networks: Technology, Protocols and Applications”, Wiley ISBN: 978-81-265-2730-4 (Students Edition)
4. Robert Faludi, “Wireless Sensor Networks”, O'REILLY, ISBN 13: 978-93-5023-289-7

Digital Content:

1. **IoT PDF: Converging Technologies for Smart Environments and Integrated Ecosystems IERC Book Open Access_2013**
2. **Content Developed by the BoS**

310244 Database Management Systems Applications

Teaching Scheme:
Theory: 3 Hrs/Week

Examination Scheme:
In Semester Assessment: 30 Marks
End Semester Assessment: 70 Marks

Course Objectives:

- To learn Advance techniques in Database Management Systems
- To learn advanced storage technologies
- To learn Database Programming

Course Outcomes:

- Ability to handle Advance Databases
- Ability to use advanced storage technologies,BIGDATA
- Ability to program databases

Unit I	**Introduction to Databases:** Database Concepts, Data models, legacy relational model-R-Model, Database concepts, Data Model and types, ER Modeling, Concept of normalization. Introduction to SQL (OODB) using MYSQL: SQL, DDL, DML, DCL, TCL, View, Index	4 Hrs
Unit II	**Advanced Database Techniques:** Structured verses unstructured data, Concept of NOSQL database, Comparative study of SQL and NOSQL, Databases types, NOSQL Data Modeling, Benefits of NOSQL, NOSQL using MongoDB- mongoDB shell, data types, manipulation(insert, update, delete documents), querying, aggregation, indexing, croud-sourcing.	6 Hrs
Unit III	**Database Transactions- Transaction Management:** Transaction Management and Concurrency Control, Performance tuning and query optimization of SQL and NoSQL Databases.	6 Hrs
Unit IV	**Database Architecture:** Client-server model: two tire, three tire, Parallel Databases, Distributed Databases, Database connectivity and Web Technologies,Database administration and Management, Connectivity using MongoDB, Cassandra.	8Hrs
Unit V	**Big data management and Programming** XML: XML-Introduction ,XML DTD's Domain specific DTD's, Querying XML data, JSON, HADOOP-Introduction, Building blocks of hadoop, components of hadoop (HDFS, Mapreduce) , HBASE, HIVE, SSD., Cloudera, Oracle Cloud,Oracle BDB, MongoDB, BIGDATA, R programming	8Hrs
Unit VI	**Advances in Databases-Data:** Data Warehouse:- data warehouse introduction , difference between operational database and data warehouse, architecture of data warehouse. Introduction to Data mining techniques: A/B Testing, classification, regression, prediction, clustering, summarization, association rule, sequence discovery, time series analysis Machine learning for big data:- introduction to machine learning, type of machine learning algorithm, supervised and unsupervised algorithm, Recommendation algorithm. Business Intelligence: Introduction, BIS components, Business models, Business analysis framework from DW.	8 Hrs

Text Books

1. Abraham Silberschatz ,Henry Korth , S.Sudarshan,"Database System concepts",5th Edition ,McGraw Hill International Edition
2. Database Principals: Fundamentals of Design, Implementation and Management, Coronel, Morris and Rob, 9th edition, Cengage Learning, ISBN: 978-81-315-1736-9
3. MongoDB: The Definitive Guide by Kristina Chodorow
4. Big Data Analytics with R and Hadoop by Vignesh Prajapati
5. Jiawei han, Micheline Kamber, "Data mining :Concepts and systems ",Morgan Kaufmann publishers
6. Margaret H. Dunham, “Data mining”, Pearson publication

Reference Books

1. http://docs.mongodb.org/manual/
2. Chuck Lam, “Hadoop in action”

Digital Content

1. Big Data: The next frontier for innovation competition and productivity (pdf published by Mckinsey)
2. Chamber J, “Software for Data Analysis: Programming with R”, Springer, 2008, ISBN : 978-0-387-75935-7 (soft copy)
3. Content Developed by BoS

310245 Forensics and Cyber Applications

Teaching Scheme:	**Examination Scheme:**
Theory: 3 Hrs/Week	**In Semester Assessment : 30 Marks**
	End Semester Assessment: 70 Marks

Course Objectives:

- To learn Forensics and use of Computers
- To learn Tools used in Computer Forensics and Cyber Applications
- To learn programming for Computer Forensics

Course Outcomes:

- To develop Computer Forensics Awareness
- Ability to use Computer Forensics Tools
- Ability to use Computer Forensics Cyber Applications

Unit I	**Basics of Computer Networks:** Protocols and Standards, OSI Model, TCP/IP Model, Network topology (Physical & logical), LAN standards, Ethernet (802.3) Transmission media: Guided transmission media - Twisted Pair, Coaxial and Fiber-optic cables, Switching techniques: Circuit switching, Packet switching and message switching, Network Hardware Components: Connectors, Repeaters, hubs, NICs, Bridges and Switches **Fundamentals of Mac Protocols:** Motivation for a specialized MAC, Fundamentals of MAC protocols, Sensor MAC Case Study (Protocol overview, Periodic listen and sleep operations, Schedule selection and co-ordination, Adaptive listening, Message passing), IEEE 802.15.4 protocol: Physical, MAC layer, naming and addressing, Assignment of MAC addresses, Distributed assignment of locally unique addresses, content based and geographic addressing	10 Hours
Unit II	Foundations of digital Forensics,Language of Computer Crime Investigation, Digital Evidence of Courtroom, Cyber crime Law: United State Perspective, Indian Perspective, Indian IT Act, conductive Digital Investigation, Handling a Digital Crime Scene: Principles, Preservation, Modus Operandi, Motive, and Technology .	6 Hours
Unit III	Violent Crime and Digital Evidence, Digital Evidence as Alibi, Gender Offenders on the Internet,Computer Intrusions,	6 Hours
Unit IV	Cyber stalking, Computer Basics for Digital Investigators, Applying Forensic Science to Computers	4 Hours
Unit V	Digital Evidence on Windows Systems, Digital Evidence on UNIX Systems,Digital Evidence on Mobile Devices, Intellectual Property Rights	6 Hours
Unit VI	Network Basics for Digital Investigators,Applying Forensic Science to Networks, Digital Evidence on the Internet, Digital Evidence on Physical and Data-Link Layers,Digital Evidence at the Network and Transport Layers, Security and Fraud detection in Mobile and wireless networks.	8 Hours

	Text Books
1	Digital Evidence& Computer Crime, Eoghan Casey Bs Ma Ac, ELSEVIER-Academic Press,Third Edition, ISBN 13 : 978-0123742681, ISBN 10 : 0123742684

2	Unix and Linux System Administration Handbook, Evi Nemeth,Garth Snyder, et al, Person Publication,
3	Kurose, Ross “Computer Networking a Top Down Approach Featuring the Internet”, Pearson; 6th edition (March 5, 2012), ISBN-10: 0132856204,ISBN-13: 978-0132856201 (pdf available)
	Reference Books
1.	Guide to Computer Forensics & Investigation, Bill Nelson, Amelia Phillips, christopher Steuart, Cengage Learning, Fourth Edition, ISBN 13 : 978-1435498839, ISBN 10 : 1435498836
2.	Ivan Stojmenovic, Handbook of Wireless Networks and Mobile Computing, Wiley India Student Edition, ISBN 978-81-265-0768-9
	Digital Contents
1	Computer Forensics and Cyber Crime Notes Prof. Sarang Joshi

310246 Programming Laboratory I

Teaching Scheme: **Examination Scheme:**
Practical: 4 Hrs/Week **Practical : 50 Marks**
Oral: 50 Marks

Course Objectives:

- To develop Database programming skill with multi-core programming
- To develop Operating Systems programming and administrative skills
- To develop use data storage devices and related programming and Management skills

Course Outcomes:

- Ability to write programs at systems level operating system modules
- Ability of problem solving using multi-core, advanced databases techniques and tools
- Ability to handle and programming of storage devices

Tools: Latest version of 64 Bit Operating Systems Open Source Fedora-19 or Higher equivalent with LAMP tools or Windows-8, with Multicore CPU equivalent to Intel i5/i7 4th generation onwards supporting Virtualization and Multi-Threading, 8 GB RAM, 500GB/1TB HDD, Latest versions of 64-Bit Programming languages such as Microsoft Visual Studio(ver. 12 or Higher) or equivalent open source, Eclipse 64-bit Platform, 64-bit Database Client-Server architecture equivalent to IBM 3250 or higher equivalent with latest versions of 64-bit Databases Oracle MySQL, MongoDB, CauchDB or equivalent Open Source Databases, Android 4.4, adb, Wireless Network supporting High End data traffic Tools: 64-bit or latest operating systems like Fedora or equivalent,64-bit Multicore Server Machine with SSD, LAMP, MongoDB, BIG DATA Tools, Hadoop Mapreduce, Kiggle or open source equivalent, Eclipse, Python, Java, Android, OpenCL, (Jullia: Download from Intel Science and Technology Center for BigDATA) Jullia For Open BIGDATA Computation, R Programming, CUDA tool kit for Fedora 19 or equivalent, Tablet and Mobile development tools and other tools published by the BoS time to time.

Documentation:
64 bit LATXT generated PDF with title colors and content beautification as per the template and revisions published by the BoS time-to-time.

Assignments Group A (Mandatory)

1	DBMS using connections(Client-Data sever, two tier) Oracle/MySQL (ODBC/JDBC), SQL prompt to create data base tables insert, update data values, delete table, use table, select queries with/without where clause. ,demonstrate use of stored procedure / function (create procedure at the data side and make use of it on the client side)	
2	DBMS using connections(Client-application server-Data sever, three tier) Oracle/MySQL (ODBC/JDBC), SQL Joints, prompt.	
3	Design and Develop SQL DDL statements which demonstrate the use of SQL objects such as Table, View , Index using Client-Data sever(two tier)	
4	Write a program in Python/C++ to read display the i-node information for a given text file, image file.	
5	Write an IPC program using pipe. Process A accepts a character string and Process B inverses the string. Pipe is used to establish communication between A and B processes using Python or C++.	

6	Use Python for Socket Programming to connect two or more PCs to share a text file.	
Assignments Group B (Any Six Assignments, All assignments to be covered in a Batch)		
1	Design at least 10 SQL queries for suitable database application using SQL DML statements: Insert, Select, Update, Delete Clauses using distinct, count, aggregation on Client-Data sever(three tier)	
2	Implement database with suitable example using MongoDB and implement all basic operations and administration commands using two tier architecture.	
3	Use MongoDB to process semi structured and unstructured data collections such as Rfid, images, blogs use python/Java MongoDB interface.	
4	Write an python/Java application using MongoDB to maintain the blog for composing the blog consists of text columns, images and videos also calculate the hit or users visited by drawing 2D graphs.	
5	Write a program in Python/C++ to test that computer is booted with Legacy Boot ROM-BIOS or UEFI.	
6	Write a program in C++ to create a RAMDRIVE and associate an acyclic directory structure to it. Use this RAMDRIVE to store input, out files to run a calculator program.	
7	Write a program in C++ to develop a tool using GRUB2 or GRUBx64.rfi select and display a boot partition. (use appropriate overloading)	
8	Write a Python/Java/C+ program to verify the operating system name and version of Mobile devices.	
9	Write a program using MongoDB to compose a web news-letter consisting of videos, images, text use python MongoDB interface.	
10	Create a iso boot image using open source tools.	
11	Write a python program for creating virtual file system on Linux environment.	
12	Write a program in C++ to make USB Device Bootable by installing required system files	
13	Write a program in python for USB Device File Management. Check usefulness of command *e2fsck* for different file systems mounted on computer.	
14	Aggregation and indexing with suitable example using Cassendra and RdfID based employees attendance system.	
15	Aggregation and indexing with suitable example using MongoDB.	
16	Map reduce operation with suitable example using MongoDB.	
17	Indexing and querying with MongoDB using suitable example.	
18	Connectivity with MongoDB using any Java application.	
19	Using MongoDB create a database of employee performance, employee attendance on the workstation. Perform statistical analysis for the results of the products produced by employees rated as passed ok, damaged products (5 samples per batch size 1000) and the portion covered in the training and absentee of the employees during training. Use programming language R. (or R-Python/R-Java) or equivalent assignment using R Programming Language for BiGDATA computing.	
	Assignment Group C: Advance Technology Assignments (Any One, all three to be covered in a Batch)	
1	BIG DATA applications using Hadoop	

2	BIG DATA applications using Blogs	
3	Big Data Predictive Machine Learning	
4	Create and test functioning of Windows-8 ReFS (Resilient File System)	
	Laboratory Manual to be prepared by the college teachers and get it approved by the BoS	
	Applicable PR/OR/TW Examination shall be conducted by the pair of examiners as per university rules and shall be on the experiments performed by the students. During Examination the student must write the Mathematical modeling of the problem statement before attempting the software solution. The software solution must be as per mathematical modeling. The student must demonstrate the running output with and without debug mode. The scheme of assessment shall be published by the BoS.	

310247 Programming Laboratory II

Teaching Scheme:	**Examination Scheme:**
Practical: 4 Hrs/Week	**Term Work: 50 Marks**
	Oral: 50 Marks

Course Objectives:

- To develop Programming Skills in Data Communication, Wireless Sensor Networks and Computer Forensics and Cyber applications using multi-core freaturs.
- To develop skills use professionally special tools used for Data Communication, WSN and Forensics

Course Outcomes:

- The Students must be able perform programming for Data communication
- The Students must be able perform programming using Wireless Sensor Networks using multicore programming features.
- The Students must be able perform programming for Computer Forensics Cyber Applications

Tools: Latest version of 64 Bit Operating Systems Open Source Fedora-20 or Higher equivalent with LAMP tools or Windows 8, with Multicore CPU equivalent to Intel i5/7 4th generation on-wards supporting Virtualization and Multi-Threading, 8 GB RAM, 500GB/1TB HDD, Latest versions of 64-Bit Programming languages such as Microsoft Visual Studio(ver. 12 or Higher) or equivalent open source, Eclipse/QT 64-bit Platform, Gigabit Switch and Router, ADSL routers, Optical router, VoIP setup, SAN, IP based printer, Camera,Two server class of machines (64 bit) , NS3, Fire Sensors

Tools : CAINE 5.0 (Computer Aided Investigative Environment) or latest 64 bit version or equivalent, SIFT 3.0 or open-source equivalent 64 bit, Digital Forensics Framework, Open Computer Forensics Architecture 2.3 or open-source equivalent and other tools published by the BoS time to time. Rubi or RailsWeb tools.

Documentation:

32/64 bit LATXT PDF with title colors and content beautification as per the template and revisions published by the BoS time-to-time. The problem Solving description must be done with mathematical modeling and Theory of Computing.

Assignments Group A (Mandatory)

1	Implementation of following spoofing assignments using C++ multicore Programming a) IP Spoofing b) Web spoofing.	
2	A fire is to be detected using relevant wireless sensor network installed in a remote location to communicate the data to the central server for the monitoring purpose and detection of the fire. Write a program to implement the system using WSN and Different data communication strategies/ algorithms (at least two) to compare the reliability of the data received and efficient timing. Use of Fort Forwarding/Tunneling Protocol is expected.	
3	Write a computer forensic application program in Java/Python/C++ for Recovering Deleted Files and Deleted Partitions	
4	A person on a nearby road is trying to enter into a WiFi network by trying to crack the	

	Password to use the IP Printer resource; write a program in Java/Python/C++ to detect such attempt and prohibit the access. Develop the necessary scenario by Using an IEEE 802.11, configure a Wi-Fi adapter and Access Point.	
5	Write a program to implement Pulse Code Modulation Technique to transfer the data to other computer.	
6	Write a program in C++ /Python to analyze email header.	
Assignments Group B (Any Six Assignments, All assignments to be covered in the Batch)		
1	Develop a GUI and write a Java/Pythn/C++ program to monitor Network Forensics, Investigating Logs and Investigating Network Traffic.	
2	Write a program in Python for Investigating Wireless Attacks using Multi-core programming.	
3	Write a program in Python for Investigating Web Attacks. Finding originator's IP, Subnet Mask and Default gateway where a Web Server is connected using Optical Router.	
4	Create a Scenario and write a program for overcoming a Website hacking problems and identifying hacker machine using Java/Python/C++. Develop a prototype website using Ruby on rails.	
5	Write a program in C++ for Tracking Emails and Investigating Email Crimes	
6	Install and use Android Mobile Forensics Open Source Tools.	
7	Write a program to Implement a packet sniffing tool in C++/Java/Python.	
8	Write a program in C++ to implement to identify DOS attack on a wireless cluster of servers.	
9	Install and use open source tools to Identifying various types of WiFi attacks. Write a C++/ Java/Python program to identify atleast one such attack.	
10	Install and use a open source tool to Identifying MMS attacks, create necessary Scenario.	
11	Design and implementation of Honeypot	
12	Write a program to identifying private data acquisition of digital evidence using Java in a WiFi system, use SAN storage(BIGDATA)	
13	Write a program to Implement a packet sniffing tool in C++	
14	Write a program to Implement a fingerprint recognition using Java Programming	
15	Write a program for identifying the image tampering, voice data (recorded/Blogged/twitted/Social Web Sites) tampering Python Programming. use SAN storage(BIGDATA)	
16	Write a program for identifying the voice data (recorded/ blogged Video/twitted/ Social Web Sites) tampering, where a Mic is attached through WSN. use SAN storage(BIGDATA)	
17	Write a program for Identifying the tampering of digital signature using Python	
18	Write a C++/Java program for Log Capturing and Event Correlation.	
19	Write a tool to detect and prevent Capturing mobile messages in Python/Java.	
Assignment Group C: Advance Technology Assignments (Any One)		
1	Implementation of Steganography program.	
2	Implement a program to generate and verify CAPTCHA image.	

3	Intrusion detection system	
4	Write a program to detect and prevent windows 8 registry Hacks and Twicks	
5	Simulate the performance of DSDV, AODV and DSR routing protocols over the WSN. Installation and configuration of WSN using ZigBee protocol	
6	Set up a small wireless sensor network of few nodes and show communication between two nodes using NS3 or equivalent	

Note: While Conducting the Term work and Oral Examination Students must demonstrate the running Program to the examiners. Laboratory Manual to be prepared by the college teachers and get it approved by the BoS.

Applicable PR/OR/TW Examination shall be conducted by the pair of examiners as per university rules and shall be on the experiments performed by the students. During Examination the student must write the Mathematical modeling of the problem statement before attempting the oral/termwork solution. The solution must be as per mathematical modeling. The student must demonstrate the running output with and without debug mode. The scheme of assessment shall be published by the BoS.

310248 Employability Skills Laboratory

Teaching Scheme: **Examination Scheme:**
Practical: 2 Hrs/Week **Term Work: 50 Marks**

Course Objectives:

- To learn use of advance programming, documentation, presentation and communication Tools
- To learn use to group discussions in problem solving
- To learn technology and group leadership skills
- To learn administrative skills and responsibilities in teamwork
- To learn quantitative skills
- To learn technology skills

Course Outcomes:

- Ability to understand need of technical competence required for problem solving
- Ability to understand employers requirements
- Ability to understand professional and group behavioral ethics

Tools: Latest version of 64 Bit Operating Systems Open Source Fedora-19 or Higher equivalent with LAMP tools or Windows 8, with Multicore CPU equivalent to Intel i5/7 4th generation onwards supporting Virtualization and Multi-Threading, 8 GB RAM, 500GB/1TB HDD, Latest versions of 64-Bit Programming languages such as Microsoft Visual Studio(ver. 12 or Higher) or equivalent open source, Eclipse 64-bit Platform, 64-bit Database Client-Server architecture equivalent to IBM 3250 or higher equivalent with latest versions of 64-bit Databases Oracle MySQL, MongoDB, CauchDB or equivalent Open Source Databases, Wireless Network supporting High End data traffic, sensors and other tools published by the BoS time to time.

Documentation:
32/64 bit LATEX PDF with title colors and content beautification as per the template and revisions published by the BoS time-to-time.

Course Write-up Theory shall include:

Understanding problem solving requirements: Customer Requirements, Employer requirements and technology requirements.

Competency assessment using:

Employer's perspective: Reliability, Integrity, Teamwork, Willingness to learn, Entrepreneurship, Self-discipline, Communication, Self-motivation, Flexibility, Technical leadership

Students Perspective: Basic computer skills,Technical skills, Use of modern tools, Advanced computer skills, System design, Communication and gestures, Responsibility, Verbal communication, Application of knowledge, Creativity, Gender Co-existence, respect, social and ethical responsibilities.

Use of Supporting Technology Perspective: In addition to development tools it is very important to use group communication and information sharing technologies. Students are expected to acquire following skills:
Posting a question on the blog or forum, writing and maintaining a mailing list mails, Writing outputs or

logs using PostBin tools or equivalent, use of GIT(refer to github Web site) for revision control or open source equivalent, Forwarding a resume for the Job as per the advertisement (Referring to the following assignment)		
	A customer wants to run Unique-identity project for the residents bearing its nationality as national-resident, NRI, NRE and migrated along with personal information, gender information with family tree(Maternal and paternal), photographs, finger-prints, crime records, earning methods, Bank Accounts, marital status, relationships,Diseases, Medical Insurance details, Social Contributions, Driving license and RfiD tags for self owned Vehicles. The customer wants to find and monitor using reports the status of ethical values in the society, growth in wealth, poverty line issues, health issues, increase/decrease in crimes and family issues and such more issues. Or equivalent assignment covering BIGDATA, WSN and OS. Laboratory Teacher(i.e. a customer role and an employer role) is expected to give above problem solving assignment to the batch of students and observe the students performance for Employer's perspective and Students Perspectives and grade every student on the scale of Strong, Good, Fair, Poor, Needs Improvement. Also, special efforts (create documentary evidence of weekly progress for the same) are to be taken for students getting Poor and Needs improvement remarks. Also a teacher shall get the GAP analysis done from the students with respect to employers perspective and Students perspective.	
	Note: Any Equivalent Assignment must be approved by the Board of Studies. Laboratory Manual to be prepared by the college teachers and get it approved by the BoS	

310249 Principles of Concurrent and Distributed Programming

Teaching Scheme:	**Examination Scheme:**
Theory: 4 Hrs/Week	**In Semester Assessment: 30 Marks**
	End Semester Assessment: 70 Marks

Course Objectives:

- To learn concurrent architecture and programming
- To learn distributed programming
- To use concurrent and parallel programming tools and applications

Course Outcomes:

- Ability to perform concurrent programming
- Ability to perform distributed programming
- Ability to use concurrent and parallel programming using GPU

Unit I	**Concepts, Overview, Programming environments, –**Computation Models, Distributed programming languages LISP, YACC, Programming environmental tools Open GL, MPI Java	6 Hrs
Unit II	Concurrent grammar, communication and synchronization of concurrent tasks process/Thread System process migration, shared memory, Concurrent LISP, Concurrent YACC, Concurrent java	8 Hrs
Unit III	The death of single core solution, NVIDA and CUDA, GPU hardware, alternatives to CUDA, Understanding parallelism with GPUs, CUDA hardware overview, Parallel architectures and Programming principles-Parallel computing, Parallel architecture, Architectural classification scheme, Parallel programming models, parallel algorithms, performance analysis of parallel algorithms.	8 Hrs
Unit IV	Distributed Computing Systems, models, Issues in designing distributed operating systems, DCE	6 Hrs
Unit V	Virtualization and programming for Xen-Overview of virtualization, resource vitalization, need and advantages of Virtualization, Xen-overview & X86 virtualization, Xen & virtualization resources, installation & Configuration, Virtual Machine booting & Configuration	6 Hrs
Unit VI	Cloud and Mobile Computing Principles, CUDA Blocks and Treads, Memory handling with CUDA, Multi-CPU and Multi-GPU solution	6 Hrs

Text Books

1. Concepts, Techniques, and Models of Computer Programming by Peter Van Roy and Seif Haridi - MIT Press Ltd, 2004
2. Distributed Operating Systems by P.K.Sinha,PHI Publications
3. Parallel Computing by M.R.Bhujade,New Age International(p) Ltd. New Age Science, 2nd Edition, **ISBN-10:** 1906574200, **ISBN-13:** 978-1906574208
4. Multicore Programming by Wiley publications

Reference Books

1. Peter Barry, Patric Crowley, “Modern Embedded Computing, Elsevier, ISBN: 978-93-81269-77-0 (Chapter 15 Only,)
2. Shane Cook, “CUDA Programming”, Elsevier, ISBN: 978-0-12-415933-4

Digital Content: Content Developed by the BoS

310250 Embedded Operating Systems

Teaching Scheme:	**Examination Scheme:**
Theory: 4 Hrs/Week	**In Semester Assessment: 30 Marks**
	End Semester Assessment: 70 Marks

Course Objectives:

- To Learn the Concepts of Embedded Systems processors and Operating System
- Develop ability to use Embedded Operating utilities in Embedded Linux

Course Outcomes:

- Ability to write technical content using Embedded Linux
- Ability to write Embedded Programming

Unit I	Operating Systems Concepts, Real-Time Tasks, Real-Time Systems, Types of Real-Time Tasks, Real-Time Operating Systems,	4 Hrs
Unit II	Processor Basics, Integrated Processors: Systems on Chip, ARM Processors history, Hardware Platforms, ARM Architecture, Interrupt Vector Table, Arm Programming, Assembly language, Instruction Set, Arithmetic, Logical and Conditional, load-store instructions,Constants, Readonly and Read-write memory, Multiple Register Load Store. ARM-9, ARM Cortex-M3, Case Study of Begal-Black-Bone: Architecture, Uboot, Interfacing and Programming	8 Hrs
Unit III	LSB, OSDL, OSDL Mobile Linux Initiative, Linux Background, Linux Kernel Construction, Tool Chain, Tools Overview, Kernel Build System, Kernel Initialization: BIOS verses Boot loader, U-Boot, Anatomy of Embedded Systems: POST and Boot Process, Kernel Initialization, *init*, Storage Considerations and memory management, BusyBox, Execution Context,Process Virtual Memory, Cross-Development Environment, Embedded Linux Distributions, Do-It-Yourself Linux Distributions, Initialization Flow of Control, Kernel Command Line Processing, Subsystem Initialization, The *init* Thread, System Initialization,	8 Hrs
Unit IV	Bootloaders, Device Driver Basics: Character Device, PCI Device Drivers, File Systems, Device Tree, MTD Subsystem, Embedded Development Environment,	8 Hrs
Unit V	Development Tools, ssh, Kernel Debugging Techniques, Debugging Embedded Linux Applications, Stepper Motor Controller interfacing using Begal Black Bone Embedded System, Embedded Graphics and Multimedia Tools and Applications.	6 Hrs
Unit VI	Porting Linux, Linux and Real Time, Embedded Android: Bootloader, Kernel, Init, Zygote, System Server, Activity Manager, Launcher (Home), Embedded Android Applications: Calculator, Twitter Search App, Slide Show App	6 Hrs

Text Books

1. Lyla B. Das, “Embedded Systems: An Integrated Approach” Pearson, ISBN:978-81-317-8766-3 (Chapters 7,8,10,11)
2. Christopher Hallinan, “Embedded Linux Primer”,Prentice Hall, ISBN-10: 0-13-167984-8, ISBN-13: 978-0-13-167984-9

Reference Books

1. Peter Barry, Patric Crowley, “Modern Embedded Computing, Elsevier, ISBN: 978-93-81269-77-0 (Chapter 4,5,6,7,8,10,14 Only,)
2. Karim Yaghmour, “Embedded Android”,O'R*eilly, ISBN: 978-1-449-30829-2, Pdf*

Digital Content: Content Developed by the BoS

310251 Computer Networks

Teaching Scheme:	**Examination Scheme:**
Theory: 4 Hrs/Week	**In Semester Assessment: 30 Marks**
	End Semester Assessment: 70 Marks

Course Objectives:

- To understand the Network Architecture.
- To learn and understand various Networking Protocols & Layers
- To learn and understand wireless technologies

Course Outcomes:

- Ability to setup, install and configure networks
- Ability of network programming
- Ability to use network protocols, wireless technologies

Unit I	**Application Layer:** OSI Model Block Diagram, Application layer protocols: Functionality and header formats, HTTP, DNS, FTP, SMTP, TELNET, DHCP	4 Hrs
Unit II	**Transport Layer:** TCP, UDP, Socket Programming, TCP Flow control, TCP congestion control, TCP in wireless network, Real time transport protocol, Stream control transmission protocol (SCTP), Quality of services (QoS), Differentiated services, Integrated services	8 Hrs
Unit III	**Network Layer:** Basics of IPV4, ICMPV4, IPV6 in detail (Motivation, Features, Address representation, Unicast and Multicast addresses, Header format), ARP, RARP, Mobile IP, Distance vector and link state routing algorithms, Routing protocols: RIP, OSPF, BGP	8 Hrs
Unit IV	**Basic Concepts in Wireless LAN:** Wireless Standards (802.11 a/b/g/n/ac/ad), Wireless LAN and Technology, Wireless application protocols(Architecture and application)	6 Hrs
Unit V	**Advanced Wireless Technologies:** VoIP Architecture and Applications, Vehicular network, Delay tolerant network.	6 Hrs
Unit VI	**Advance Network Technologies:** Virtualization, Software defined network, ATM (Overview, Protocol Architecture, AAL), GMPLS, Introduction of optical networks,Propagation of Signals in Optical Fiber, Client Layers of the Optical Layer	8 Hrs

Text Books

1. Kurose, Ross “Computer Networking a Top Down Approach Featuring the Internet”, Pearson; 6th edition (March 5, 2012), ISBN-10: 0132856204,ISBN-13: 978-0132856201 (pdf available)
2. Andrew S. Tenenbaum, “Computer Networks”,4th Edition, PHI, ISBN 81-203-2175-8.
3. Georgios I. Papadimitriou, Andreas S. Pomportsis, P. Nicopolitidis, Mohammed S. Obaidat,“Wireless Networks”, Wiley Student Edition ISBN: 978-0-470-84529-5

Reference Books

1. Fourauzan B., "Data Communications and Networking", 4th edition, Tata McGraw-Hill,Publications, 2006, ISBN 0 - 07 – 0634145
2. Olifer & Olifer ,”Computer Networks-principles, technologies & protocols for network design”, WILEY
3. Rajiv Ramaswami, Kumar Shivarajan, Glan Shasaki, “Optical Networks a Practical Perspective”, Elsevier-Morgan Kaufmann ISBN: 978-0-12-374092-2 pdf
4. Vijay K, Garg, “Wireless Communication and Networking”, Elsevier Morgan Koffman, ISBN: 978-81-312-1889-1

Digital Content: Developed by the Bos

310252 Software Engineering

Teaching Scheme:	**Examination Scheme:**
Theory: 3 Hrs/Week	**In Semester Assessment: 30 Marks**
	End Semester Assessment: 70 Marks

Course Objectives:

- To learn and understand the principles of Software Engineering
- To Learn and understand Software Development Life Cycle
- To apply Project Management and Requirement analysis principles to S/W project development.
- To apply Design and Testing principles to S/W project development.

Course Outcomes: The student will be able to

- Compare and chose a process model for a software project development
- Analyze and model software requirements of a software system
- Design and Modeling of a software system with tools
- Designing test cases of a software system
- Prepare the SRS, Design document, Project plan of a given software system

Unit I	Nature of Software , Software Engineering, The Software Process , Software Myths, A Generic Process Model, Prescriptive Process Models: The Waterfall Model, Incremental Process Models , Evolutionary Process Models ,Concurrent Models , Specialized Process Models , The Formal Methods Model, The Unified Process Personal, Agility Principles , Extreme Programming (XP), Scrum, Introduction to Clean Room Software Engineering	6 Hrs
Unit II	**Requirements Engineering** Requirements Engineering,Eliciting Requirements, Collaborative Requirements Gathering,Quality Function Deployment,Usage Scenarios,Elicitation Work Products, Developing Use Cases, Building the Requirements Model, Negotiating Requirements,Validating Requirements, Analysis: Scenario-Based Modeling,UML Models,Developing an Activity Diagram,Swim-lane Diagrams,Class-Based Modeling,Requirements Modeling Strategies: Flow-Oriented Modeling, Creating a Behavioral Patterns for Requirements Modeling, State Machine Diagram with orthogonal states, Requirements Modeling for Web Apps, SRS	8 Hrs
Unit III	**Design Methods and Models** The Design Process,Concepts of design, Design Quality, Design Principles, Object-Oriented Design Concepts, Design Classes, The Design Model and elements, Software Architecture,Importance, Architectural Styles, Architectural Design, Assessing Alternative Architectural Designs, using Architectural Styles in Designs, Component Design, Class-Based Components, Conducting Component-Level Design, Component-Level Design for WebApps, User Interface Design, The Golden Rules , User Interface Analysis and Design, Interface Analysis Interface Design Steps, WebApp Interface Design, Design Evaluation , Design Document, Modifiability: SAAM Method, ATAM Method, The HASARD Method.	8 Hrs
Unit IV	**Testing Principles** A Strategic Approach to Software Testing,Strategic Issues,Test Strategies for	6 Hrs

	Conventional Software,Test Strategies for Object-Oriented Software,Test Strategies for WebApps, Validation Testing, System Testing, The Art of Debugging, Software Testing Fundamentals, White-Box Testing , Basis Path Testing, Control Structure Testing, Black-Box Testing, Model-Based Testing, Testing for Specialized Environments, Architectures, Object-Oriented Testing Strategies, Object-Oriented Testing Methods, Test Cases and the Class Hierarchy, Testing Concepts for WebApps, Testing Process—An Overview, User Interface Testing , Test plan,Positive Testing Negative Testing.	
Unit V	**Project Planning and management** The Management Spectrum ,Software Scope,Problem Decomposition, Process Decomposition ,Process and project metrics, Size-Oriented Metrics, Function-Oriented Metrics, Reconciling LOC and FP Metrics, Object-Oriented Metrics , Integrating Metrics within the Software Process, Software Project Estimation, Decomposition , Process-Based Estimation, Estimation with Use Cases , Empirical Estimation Models, The Structure of Estimation Models, The COCOMO II Model , Project scheduling: Basic Concepts, Defining a Task Set for the Software Project , Scheduling : Tracking the Schedule, Earned Value Analysis Risk management: Reactive versus Proactive Risk Strategies, Risk Identification, Assessing Overall Project Risk, Risk Projection, Developing a Risk Table, Assessing Risk, Project Plan	8 Hrs
Unit VI	**Advanced Software Engineering** Software Quality, McCall's Quality Factors, ISO 9126 Quality Factors, Software Reliability, Measures of Reliability and Availability, Software Safety, Formal Methods Concepts,Applying Mathematical Notation for Formal Specification, Introduction to Formal Specification Languages : Object Constraint Language (OCL), Z Specification Language, Software reuse, Distributed software engineering, Service-oriented architecture, Embedded software, Aspect-oriented software engineering	6 Hrs
Text Books		
1. Roger S Pressman "Software Engineering : A Practitioner's Approach " 7th Edition Mcgraw-Hill ISBN:0073375977		
2. Ian Sommerville " Software Engineering" 9th edition Pearson Education SBN-13: 978-0-13-703515-1, ISBN-10: 0-13-703515-2 , pdf downloadable		
3. Hong Zhu "Software Design Methodology", Elsevier ISBN: 978-81-312-0356-9		
Reference Books		
1. Pankaj Jalote " An Integrated Approach to Software Engineering" 3rd Edition Narosa Publication ISBN: 81-7319-702-4 pdf down loadable 2. Rajib Mall " Fundamentals of Software Engineering" 3rd edition PHI 3. Pfleejer " Software Engineering- Theory and Practice" 4th edition 4. Martin Fowler "Distilled UML" 3rd edition 5. Hans Van Vilet "Software Engineering Principles and Practice" 3rd edition Wiley		
Digital Content: Developed by the BoS		

310253 Digital Signal Processing Applications

Teaching Scheme:	**Examination Scheme:**
Theory: 3 Hrs/Week	**In Semester Assessment: 30 Marks**
	End Semester Assessment: 70 Marks

Course Objectives:

- Study and understanding of representation of signals and systems.
- To learn and understand different Transforms for Digital Signal Processing
- Design and analysis of Discrete Time signals and systems
- **To Generate foundation for understanding of DSP and its applications like audio, Image, telecommunication and real world**

Course Outcomes:

- Students will understand the mathematical concepts of signal representation and transformations with their analysis.
- Development of ability for generating proper solution to signal processing problems.
- Students will be capable of understanding Digital Signal Processing Applications and implementation of signal processing to various applications.

Unit I	Introduction:- The Breadth and Depth of DSP, Statistics, Probability and Noise, How digital signal is created: ADC and DAC, Signals, Linear Systems, Classification of signals, Properties of DT systems, Mathematical models for representation of DT system: Linear convolution, Linear constant coefficient difference equation, Use of Transducers in Signal Processing, Analog to Digital conversions (ADC), Sampling Process.	8 hrs
Unit II	Fourier Transform: DTFT, Properties, DFT, Circular convolution, DFT Spectral leakage, Efficient computations of DFT, Fast Fourier Transform ,Radix-2 DIT and DIF FFT Algorithms, Application of DFT, Linear filtering.	8 hrs
Unit III	Z-Transform Definition of Z-Transform, ZT and FT, ROC, ZT properties, pole-zero plot, Inverse Z-Transform, Methods, System function H(Z), Analysis of DT LTI systems in Z-domain: DT system representation in time and Z domain. Relationship of FT and ZT	6 hrs
Unit IV	Introduction to Filter Structures, components of digital filters, DT Filters Block diagram representation, equivalent structures, Basic FIR and IIR Filter structures, DT filters as DT systems, Solution of difference equation, FIR and IIR filters direct form structures,	6 hrs
Unit V	DSP Processors:DSP Building Blocks, Data Acquisition, Fix Point and Floating Point Implementation the SHARC floating Point processor, SIMD Micro Architecture and Instructions, Operating systems, Micro-Architecture consideration, Implementation Options, Intrinsic and Data type, OMAP (Open Multimedia Application Platform), DSP Applications: DSP and its benefits, Application areas, Key DSP operations, DSP processors, real world, audio, telecommunication applications and biomedical applications. (Ref: Reference Book No 3, Chapter 1, Page Nos: 1 to 36)	6 hrs
Unit VI	**DSP in Speech Processing & Image Processing:** **Audio Processing**: Human Hearing, Timbre, Sound Quality Versus Data rate, High Fidelity Audio, Companding, Speech Synthesis and Recognition, Non Linear Audio Processing, **Image Foundation and Display**: Digital Image Structure, Cameras and Eyes, Television Video Signals, Other Image Acquisition and display, Brightness and Contrast Adjustments, Gray Scale Transforms, (Ref: Text Book No 1, Chapter 22,23, Page Nos: 351 to 386)	6 hrs

Text Books

1. Steven W. Smith, “The Scientist and Engineer's Guide to Digital Signal Processing”,California Technical Publishing, 2nd Edition, PDF ISBN 0-9660176-6-8

(Colleges are expected to purchase Paperback Edition ISBN 0-9660176-4-1.)

2. “Digital Signal Processing “ , Third Ed. Prentice Hall ISBN 81-203-0720-8
3. “Digital signal processing A practical approach” Second Ed. by Emmanuel Ifeachor , Barrie W Jervis ; Pearson

Reference Books

1. Lyla B. Das Embeded systems and Integrated Approach PEARSON ISBN 978-81-317-8766-3 (Chapter number 15 only)
2. Peter Barry and Patric Crowley Modern Embedded Computing, Elsevier ISBN 978-93-81269-77-0 (Chapter number 11 only)
3. “Digital Signal Processing - A Computer Based Approach”, Sanjit K Mitra Third Ed. TMH
4. “Electrical and Electronic Measurements and Instrumentation “ A. K. Sawhney- Dhanpat Rai and Sons, Delhi -2002 Print

Digital Content: Developed by the BoS

310254 Programming Laboratory-III

Teaching Scheme:	**Examination Scheme:**
Practical: 4 Hrs/Week	**Practical: 50 Marks**
	Oral: 50 Marks

Course Objectives:

- To learn multi-core, Concurrent and Distributed Programming
- To learn Embedded Operating Systems Programming
- To learn writing Software Engineering document
- To learn Embedded/ Concurrent and Distributed Programming

Course Outcomes:

- Ability to perform multi-core, Concurrent and Distributed Programming
- Ability to perform Embedded Operating Systems Programming
- Ability to write Software Engineering Document
- Ability to perform Concurrent and Distributed Programming

Tools: Latest version of 64 Bit Operating Systems Open Source Fedora-19 or Higher equivalent with LAMP tools, Windows 8 with Multicore CPU equivalent to Intel i5/7 4th generation onwards supporting Virtualization and Multi-Threading, 8 GB RAM, 500GB/1TB HDD, CUDA/OpenCL ,GPU/Begal Bone Black(BBB)/Atmel Cortex A5/M4 Mobile-tablet processor, WSN sensor Processor boards, Latest versions of 64-Bit Programming languages such as Microsoft Visual Studio (ver. 12 or Higher) or equivalent open source, Eclipse 64-bit Platform, 64-bit Database Client-Server architecture equivalent to IBM 3250, MySQL, MongoDB, OpenMP, CUDA/OpenCL or equivalent Open Source, Wireless Network supporting High End data traffic and other tools published by the BoS time to time.

Documentation:

32/64 bit LATXT PDF as per the template and revisions published by the BoS time-to-time. It shall Cover Aim, Objectives, Mathematical Modeling covering multi-core and distributed aspects, Efficiency, Data Structures resulted out of Mathematical Modeling, Conclusion.

Assignments Group A (Mandatory)

1	Develop an application using Beeglebone Black/ ARM Cortex A5 development board to simulate the operations of LIFT.	
2	Develop an application using Beeglebone Black/ ARM Cortex A5 development board to simulate the working of signal lights.	
3	Implement an calculator (64 bit Binary Multiplication) application using concurrent lisp	
4	Apply the Following Software Engineering to all assignments(No 1,2,3 of Group A and B). Mathematical Modeling must result into UML Requirements. Apply Assignment No 4a to 4d for all Group A and Group B assignments of Embedded Operating system and Concurrent and Distributed Programming. Use tools Open source tools like ArgoUML, UMLLet, StarUML or equivalent tools for UML models) Or Use Agile or Scrum-Agile methodologies and Tools.Use of Possitive and Negative Testing.	
4a	Design mathematical model of the Application/system using set theory, algebraic system, relations and functions, Deterministic and Non-Deterministic entities..	
4b	Analyze requirements from the Problem statement, mathematical model, Domain requirements and identify Functional, Non functional, Actors, Usecases for the application/system. Create usecase diagram, activity diagram/swimlane diagram for each usecase.	
4c	Design the architecture for the system/application using package diagram , deployment diagram. Design classes using class diagram.	

4d	Design the behavior of the system/application using state machine diagram and sequence diagram.	
5	Create Project plan, SRS, Design document and Test Plan for one group-C assignment from embedded operating system or Concurrent and Distributed Programming	
6	Write an application to parse input text file concurrently and compare the result of concurrent parsing with serial parsing (Use concurrent YACC parser)	
Assignments Group B (Any Six Assignments, All assignments to be covered in the Batch)		
1	Write an application to and demonstrate the change in BeagleBoard/ ARM Cortex A5 /Microprocessor /CPU frequency or square wave of programmable frequency.	
2	Implement a Parallel Quick Sort algorithm using NVIDIA GPU or equivalent ARM board.	
3	Vedic Mathematics method to find square of 2-digit number is used in a distributed programming. Use shared memory and distributed (multi-CPU) programming to complete the task.	
4	Implement a Parallel ODD-Even Sort algorithm using GPU or ARM equivalent.	
5	Implement n-ary search algorithm using OPENMP	
6	Implement concurrent prims algorithm using OPENMP	
7	Implement *nxn* matrix parallel multiplication using CUDA/OpenCL GPU, use shared memory.	
8	Develop a network based application by setting IP address on BeagleBoard/ ARM Cortex A5.	
9	Implement a Multi-threading application for echo server using socket programming in JAVA	
10	Implement Reader-Writer problem using OPENMP	
11	Implement a dinning philosophers problem using OpenCL wherein each philosopher is a distributed computer memory in a cluster.	
12	A text file is stored in a distributed manner on three hard disks on three machines such that consecutive lines, one per hard disk are stored in cyclic manner. Write a program using OpenCL to read/Write/Modify the file.	
13	A file holds a data structure that is written and modified by number of users in a distributed manner. Multiple users on multiple computers use Read-Modify-Write cycle provided resource is available else use use modify once before exit. Write necessary Program using OpenCL.	
14	Perform Assignment No 4 of Group A for Assignment No 12 of Group-B using UMLLet	
15	Perform Assignment No 4 of Group A for Assignment No 13 of Group-B usung concurrent UML.	
Assignment Group C: Advance Technology Assignments (Any One)		
1	Develop Robotics(stepper motor) Application using Beagle Board.	
2	Develop bus arbitration logic using VME/PCI bus for cluster of CPU boards for high performance computing (BIG DATA)	
3	Implement a Distributed matrix multiplication using CUDA / OpenMPI	
Digital Content: Laboratory Manual Developed by college Teachers and get it approved by BoS		
Applicable PR/OR/TW Examination shall be conducted by the pair of examiners as per university rules and shall be on the experiments performed by the students. During Examination the student must write the Mathematical modeling of the problem statement before attempting the solution. The solution must be as per mathematical modeling. The student must demonstrate the running output with and without debug mode. The scheme of assessment shall be published by the BoS.		

310255 Programming Laboratory-IV

Teaching Scheme:	**Examination Scheme:**
Practical: 4 Hrs/Week	**Term Work: 50 Marks**
	Oral: 50 Marks

Course Objectives:

- To learn network programming techniques
- To learn Digital Signal Processing programming
- To develop use of Network Programming and DSP using WSN

Course Outcomes:

- Ability to set-up, install and configure network, WSN
- Ability to perform Concurrent programming for Networking and WSN
- Ability to use different networking protocols and tools

Tools: Latest version of 64 Bit Operating Systems Open Source Fedora-20 or Higher equivalent with LAMP tools, Windows 8 with Multicore CPU equivalent to Intel i5/7 4th generation onwards supporting Virtualization and Multi-Threading, 8 GB RAM, 500GB/1TB HDD, Latest versions of 64-Bit Programming languages such as Microsoft Visual Studio(ver. 12 or Higher) or equivalent open source, Eclipse 64-bit Platform, 64-bit Database Client-Server architecture equivalent to IBM 3250 or equivalent with Wireless Network supporting High End data traffic, Gigabit Switch and Router, ADSL routers, Optical router, VoIP setup, SAN, IP based printer, Camera,Two server class of machines (64 bit) , 64-bit NS3 or equivalent, ARM Cortex M4/A5 and other tools published by the BoS time to time.

Documentation:

32/64 bit LATXT PDF as per the template and revisions published by the BoS time-to-time.

Assignments Group A (Mandatory)

<table>
<tr><td>1</td><td colspan="2">Implementation of Packet sniffer. Program should identify header of each protocol.Use multi-core programming.</td><td></td></tr>
<tr><td>2</td><td colspan="2">Consider the network id 192.168.4.0 or such relevant IP and create four subnets namely A, B, C, D. Assign the subnet mask. Write a Python \ C++ program to Perform the following operations (use overloading if applicable).
a) Ping the machine of same subnet.
b) Ping the machine in subnet A from machine of subnet B.
c) Analyze the output of the above sub assignments.</td><td></td></tr>
<tr><td>3</td><td colspan="2">Write C++ Program with GUI to capture using remotely placed camera and read uncompressed TIFF Image to perform following functions (Menu Driven) Use of Overloading and Morphism is expected. Image Frame1 is used for displaying Original Image and Image Frame 2 is used for displaying the action performed.</td><td></td></tr>
<tr><td></td><td>• Sharpen the Image
• Blur the Image (Programmable rectangular Seed)
• Programmable image Contrast and Brightness
• Rotate image by programmable angle</td><td>• Convolution(overloading: FFT, Other)
• Histogram
• Mean and Standard Deviation of image
• PDF of a Signal acquired through ADC</td><td></td></tr>
<tr><td>4</td><td colspan="2">Write a C++ program to read the HTTP header and analyze the parameters</td><td></td></tr>
<tr><td>5</td><td colspan="2">Installing and configure DHCP server and write a program (C++\Python\Java) to install the software on remote machine.</td><td></td></tr>
</table>

6	Design and Setup LAN with Star topology to access Storage Area Network (SAN). The SAN must have DSP data, Text Data, Multimedia Data available for the access.	
Assignments Group B (Any Six Assignments, All assignments to be covered in the Batch) (within 6 assignments atleast two assignments from DSP)		
1	Write a Python program to grab the image from Camera and apply the edge detection algorithm(overloaded with Sobel variants, Others) to find the edges use BBB / ARM Cortex A5/A9/M4 Mobile Boards. Store the Images in SAN (for BIGDATA analitics)	
2	Implementation of streaming video server and displaying video at client side using Java. Videos are stored using SAN (BIGDATA)	
3	Simulation of WAN (RIP) using packet tracer/Network Simulator 3 (NS3)or higher equivalent.	
4	Study and perform Linux networking commands emulation using Python or C++.	
5	Write FTP/Telnet program using socket programming for TCP using C++	
6	Write TFTP program using socket programming for UDP using C++	
7	Create TCP/IP packet using standard TCP/IP include files and send it to the server using c++.	
8	Implement any congestion control algorithm for TCP using Python	
9	Implementation of Concurrent Text Conferencing application using Python or Java	
10	Implementation of Concurrent Proxy server program using Python or Java	
11	Implementation of Multithreaded web server. Check the functionality of web server in LAN and through the ADSL router using NAT.	
12	Implement a program for remote print manager to print documents on remote printer. (IP based printer) using Python	
13	Implementation of sliding window protocol using C++.	
14	Implementation of distance vector routing algorithm using C++.	
15	Implementing video conferencing system. Use VoIP protocol. Use Python or Java.	
16	Configure and setup network with optical router.	
17	(Refer Assignment 3 of Group A) Write a C++/ Python program to generate a Sign wave of Programmable frequency and capture samples at programmable frequency (Max up as per Nyquist Sampling Theorem) and reconstruct the Sign wave using collected Samples using ARM Cortex A5/A9. Use oscilloscope to calculate signal frequency. Write your observations. Store a Data file in SAN (BIGDATA)	
18	(Refer Assignment 3 of Group A) Write a C++/ Python program to generate a Square wave of programmable frequency. Write a function to generate Pole-Zero Diagram using multicore programming.	
19	(Refer Assignment 3 of Group A) Write a C++/ Python program to capture signal using ARM Cortex A5/A9/M4 ADC and signal generator, generate/construct a Square/Sine wave of programmable frequency and voltage Draw Voltage (y-axis) and Time (x-axis) graph. Write a function to emulate simple RC filter with R being Trim-pot(GUI meter) of 10K and C = 0.1 microFarad. Write a program to generate a Voltage-Time response curve with reference to change in R. Draw the resultant outcome graph. Store the data in SAN (BIGDATA)	
Assignment Group C: Advance Technology Assignments (Any One)		
1	To create a network with three nodes and establish a TCP connection between node 0 and node 1 such that node 0 will send TCP packet to node 2 via node 1 using NS3 or higher	

	equivalent.	
2	To create scenario of different network topology used in LAN using TCP/UDP and analysis how FTP will run over it in NS3 or higher equivalent.	
3	Designing IPv6 network and/or configuration of Dual stack IPv6 and IPv4 network.	
4	Controlling presentation slides with hands by identifying movements through Camera.	
5	Installation and setup to control the remote machine.	
6	Network Boot operations through Programming	
Reference Books: Lab Manual developed by the College Teachers approved by the BoS		
Digital Content:(Down Loadable PDF for students only) 1. Steven W. Smith, “The Scientist and Engineer's Guide to Digital Signal processing”,California Technical Publishing, 2nd Edition, PDF ISBN 0-9660176-6-8 (Colleges are expected to purchase Paperback Edition ISBN 0-9660176-4-1.) 2. Laboratory Manual developed by the College Teacher and get it approved by the BoS.		
Applicable PR/OR/TW Examination shall be conducted by the pair of examiners as per university rules and shall be on the experiments performed by the students. During Examination the student must write the Mathematical modeling of the problem statement before attempting the software solution. The software solution must be as per mathematical modeling. The student must demonstrate the running output with and without debug mode. The scheme of assessment shall be published by the BoS.		

310256 Seminar and Technical Communication Laboratory

Teaching Scheme: **Examination Scheme:**
Theory: 1 Hrs/Week **Term Work: 50 Marks**
Practical: 2 Hrs/Week

Course Objectives:

- To develop ability of thinking and motivation for seminar
- To develop ability to perform literature survey
- To develop ability to generate proof-of-concept
- To develop ability to prepare presentation
- To develop Seminar presentation and Technical Communication Skills

Course Outcomes:

- Motivation Statement
- Survey Documentation
- Proof-of-Concept and related data
- Presentation

Tools: Latest version of 64 Bit Operating Systems Open Source Fedora-19 or Higher equivalent with LAMP tools or Windows 8 with Multicore CPU equivalent to Intel i5/7 4th generation on-wards supporting Virtualization and Multi-Threading, 8 GB RAM, 500GB/1TB HDD, Latest versions of 64-Bit Programming Tools, Intelligent LCD Board Projector or LCD Projector, LATXT and other tools published by the BoS time to time.

Documentation:
64 bit LATXT presentation slides and bibliography as per the template and revisions published by the BoS time-to-time. (Softcopy Submission), GIT

Write-up Theory:
Problem solving techniques, Writing Motivation, Deciding Objectives and Outcomes, Different types of Survey and deriving conclusions from the survey, data collection, implementation technology, deriving Conclusions and presenting conclusions using graphs.
Using above points for technical communication and discussions.

Assignments Group A (Mandatory)

1	Identify the Social Problem to be solved using Computing Algorithms	Within 1st Week from the Start of Semester
2	1st Presentation to the Seminar Guide along with the Identified Problem and Motivation	In the 2nd Week from the Start of Semester
3	2nd Presentation to the Seminar Guide along with the Identified Problem and revised Motivation, Objectives and planned Outcomes using Video-conferencing or Skype like tool or open source equivalent.	In the 3rd Week from the Start of Semester
4	3rd Presentation to the Seminar Guide along with the Identified Problem and revised Motivation, Objectives and planned Outcomes, Algorithmic Survey to finalize the algorithm to be selected to solve the problem for same outcomes.	In the 4th Week from the Start of Semester
5	4th Presentation to the Seminar Guide covering above items1 to 4 along with Survey of the selected Algorithm and results obtained by the other researchers to solve same problem using Video-conferencing or Skype like tool or open source equivalent.	In the 5th Week from the Start of Semester

6	5th Presentation to the Seminar Guide per objective No 1 results of implementation using data tables and comparative outcome graphs with other researchers for same data.	In the 6th Week from the Start of Semester
7	Progress 6th Presentation to the Seminar Guide per objective No 2 results of implementation using data tables and comparative outcome graphs with other researchers for same data.	In the 7th Week from the Start of Semester
8	Progress 7th Presentation to the Seminar Guide per objective No 3 results of implementation using data tables and comparative outcome graphs with other researchers for same data using Video-conferencing or Skype like tool or open source equivalent.	In the 8th Week from the Start of Semester
9	Seminar Documentation including cover Title page, plagiarism assessment report Certificate from Guide, Abstract, list of Figures, List of Tables, Abstract, Technical write-up using Mathematical Modeling for the Problem solved, efficiency obtained, Presentation Slide using Latex including bibliography/references.	In the 9th Week from the Start of Semester
10	Seminar Presentations and evaluation by the pair of Experts including guide using Audio-Video Tools, Intelligent LCD Projector, laser Pointers in a Department seminar/Conference Hall.	In the 10thWeek from the Start of Semester

Marking Scheme:

Final Marks Should include A> Marks Given by Experts (out of 10), B> 10 Marks for the Questions asked by students and quality of answers, C>10 Marks to the report, D>10 Marks to quality of Latex Presentation Slides,E> 10 Marks to the Technical Communication skills.

Final Marks = (A+B+C+D+E) *(No. Of Students actually attended the seminar/ total Strength of the class). No assessment shall be conducted when the students attendance is less than 75% of the total strength of the class. All progress reports along with correction remarks by the guide and the final report to be submitted in softcopy in CD and one copy stored in SAN (for BIG DATA analytics)

Laboratory Manual to be developed by the College Teacher and get it approved by the BoS.