



## GOVERNMENT COLLEGE OF ENGINEERING AURANGABAD

*“In Pursuit of Global Competitiveness”*

(An Autonomous Institute of Government of Maharashtra)

**Department of Information Technology**

**Class TE IT With Effect From 2018-2019**

### Program Educational Objective(s)

After graduation and few years of graduation, the (Information Technology) graduates would

PEO I	Interpret, design and analyze data for effective problem solving
PEO II	Pursue advanced studies to adapt to current trends
PEO III	Attain professional careers and provide services in societal and environmental context for sustainable development
PEO IV	Work successfully with effective communication skills, professionalism, team work and ethical attitude

### Program Outcome(s)

The program enables students to achieve by the time of graduation:

1. An ability to apply principles and methodologies of mathematics, science, and engineering fundamentals
2. An ability to identify, study research literature, formulate the computing requirements appropriate to its solution
3. An ability to design solutions for engineering problems and design systems or processes that meet specified needs with appropriate considerations
4. An ability to conduct investigations of complex problems including design of experiments, analysis and interpretation of data to provide valid conclusions
5. An ability to create, select and apply appropriate techniques, resources, and modern engineering and IT tools with an understanding of the limitations
6. An understanding of professional engineering practice with concern to societal, health, safety and legal responsibilities
7. Recognize the impact of IT solutions in an environmental and societal context and demonstrate knowledge of and need for sustainable development
8. Understand and commit to professional ethics and responsibilities and norms of engineering practice
9. An ability to function effectively as an individual and as a member or leader in a team to accomplish a common goal
10. Communicate effectively to comprehend and write effective reports, design documentation and make effective presentations
11. An ability to engage in independent and life-long learning to enhance their careers for continuing professional development
12. An understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects

### Mapping of PEOs and POs

	<b>Programme Educational Objectives</b>	<b>Program Outcome(s)</b>
<b>PEO I</b>	Interpret, design and analyze data for effective problem solving	1,2,3
<b>PEO II</b>	Pursue advanced studies to adapt to current trends	3,4,5,6,7,9
<b>PEO III</b>	Attain professional careers and provide services in societal and environmental context for sustainable development	4,5,6,7,8,9,10,11,12
<b>PEO IV</b>	Work successfully with effective communication skills, professionalism, team work and ethical attitude	6,8,10,11,12

**Third Year Engineering (IT) Choice Based Credit System  
Semester- V**

Sr No	Subject Code	Subject	Contact Period (Hrs.)			Credits	Continuous Evaluation in terms of Marks							Total (100)
			TH	T	PR		Class Test I	Class Test II	TA	ES E	TW	Practical/Viva-voce		
1	IT3043	Computer Algorithm	3	-	-	3	15	15	10	60	-		100	
2	IT3044	Computer Networks	2	1	-	3	15	15	10	60	-		100	
3	IT3045	Software Engineering and Testing	3	-	-	3	15	15	10	60	-		100	
4	IT3046	Operating System	3	-	-	3	15	15	10	60	-		100	
5	IT3072 IT3073	<b>Professional Elective-I</b> Enterprise Resource Planning Computer Organization	4	1	-	5	15	15	10	60			100	
<b>LAB</b>														
6	IT3047	Lab: Computer Algorithm	-	-	2	1	-	-	-		25	25	50	
7	IT3048	Lab: Computer Networks	-	-	2	1	-	-	-		25	25	50	
8	IT3049	Lab: Software Engineering and Testing	-	-	2	1	-	-	-		25	25	50	
9	IT3050	Lab: Operating System	-	-	2	1	-	-	-		25	25	50	
10	IT3023	Lab: Open Source-I	-	-	2	1	-	-	-		50	-	50	
<b>Total</b>			<b>15</b>	<b>2</b>	<b>10</b>	<b>22</b>	<b>75</b>	<b>75</b>	<b>50</b>	<b>300</b>	<b>150</b>	<b>100</b>	<b>750</b>	

**Third Year Engineering (IT) Choice Based Credit System  
Semester- VI**

Sr No	Subject Code	Subject	Contact Period (Hrs.)				Credits	Continuous Evaluation in terms of Marks					
			TH	T	PR	Class Test I		Class Test II	TA	ESE	TW	Practical/Viva-voce	Total (100)
1	HS3008	Business Intelligence	2	-	-	2	15	15	10	60	-		100
2	IT3051	Theory of Computation	3	1	-	4	15	15	10	60	-		100
3	IT3052	Advanced Database Management System	3	-	-	3	15	15	10	60	-		100
4	IT3053	Mobile Computing	3	-	-	3	15	15	10	60	-		100
5	IT3074 IT3076	<b>PROFESSIONAL ELECTIVE –II</b> Linux Operating System Distributed Systems	4	-	-	4	15	15	10	60	- -		100
6	OEIT3001 OEIT3002	<b>OPEN ELECTIVE</b> Principles of Programming Language Object Oriented Modeling and Design	3	-	-	3	15	15	10	60	-		100
7	IT3054	Lab: Advanced Database Management System	-	-	2	1	-	-	-		25	25	50
8	IT3055	Lab: Mobile Computing	-	-	2	1	-	-	-		25	25	50
9	IT3075 IT3078	<b>Lab PROFESSIONAL ELECTIVE –II</b> Lab: Linux Operating System Lab: Distributed Systems	-	-	2	1	-	-	-		25	25	50
<b>Total</b>			<b>18</b>	<b>01</b>	<b>06</b>	<b>22</b>	<b>90</b>	<b>90</b>	<b>60</b>	<b>360</b>	<b>75</b>	<b>75</b>	<b>750</b>

- Students should undergo Internship/Industrial Training at the end of fourth and sixth semester of total four to six weeks duration. This is Mandatory Audit Course

## IT3043 : Computer Algorithm

### Teaching Scheme

Lectures: 03 Hrs/Week

Credits:03

### Examination Scheme

Test I

: 15 Marks:

Test II

: 15 Marks

Teachers Assessment

: 10 Marks

End Semester Exam

: 60 Marks

**Pre-requisite: IT 2032 Data Structures**

### Course Objectives:

To enable the students to design good efficient solutions to real world problems and understand the complexity of their solution in terms of memory space and computer time

### Course Outcomes:

After completing the course, students will be able to:

- CO1 Appreciate the need for analysis of algorithms.
- CO2 How to analyze the best-case, average-case and the worst-case running times of algorithms using asymptotic analysis.
- CO3 Know the standard design techniques of algorithms and know the conditions in which each particular technique is to be applied based on complexities.
- CO4 Design efficient algorithms for problems encountered in common engineering design situations.
- CO5 Know the limitations on the time complexity of algorithms i.e. the theory of NP Complete problems.

### Detailed Syllabus:

#### Unit 1 Introduction

Algorithm concept, Need for Analysis, Asymptotic notations of analysis of algorithms, analyzing control structures, worst case and average case, best case analysis, amortized analysis, Space complexities, Mathematical foundations, summation of arithmetic and geometric series,  $\sum n$ ,  $\sum n^2$ , recurrence relations and methods.

#### Unit 2 Divide and conquer

Divide and conquer basic strategy and its complexity, Sorting algorithms such as selection sort, insertion sort, bubble sort, heap sort, binary search, quick sort, merge sort, Fast Fourier Transform.

#### Unit 3

**Greedy method** - Basic strategy and its complexity, application to job sequencing with deadlines problem, minimum cost spanning trees, single source shortest path, Knapsack Problem.

**Dynamic Programming** - Basic strategy and its complexity, Principle of optimality, multistage graphs, all pairs shortest path, single source shortest paths, Huffman codes, travelling salesman problem, 0/1 knapsack problem, Matrix-chain multiplication, Longest common subsequences.

#### Unit 4

**Traversal and Search Techniques:** graph traversal, breadth first search and depth first search, connected components, DAG, Backtracking basic strategy, The Method Of

Branch And Bound, 8-Queen's problem, graph coloring, Hamiltonian cycles, Travelling Sales Person Problem Using Branch And Bound

**Unit 5 NP problems:** P and NP concept, decision Vs optimization problems, NP-hard and NP-complete problems, satisfiability, Clique, Node cover, non deterministic algorithms.

**Text Books:**

1. Introduction to Algorithms : Cormen T.H. et.al : Prentice Hall of India
2. Computer Algorithms : Horowitz, Sahani, Rajsekharan , Galgotia Publications Pvt.Ltd
3. Fundamentals of Algorithms : Brassard, Bratley , Prentice Hall

## IT3044 Computer Networks

### Teaching Scheme

Lectures: 2 Hrs/Week

Tutorial : 1 Hr/Week

Total Credits :03

### Examination Scheme

Test 1 : 15 Marks

Test 1 :15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

### Prerequisites: IT2040 Data Communication and Networking

**Course description:** Students learn about the distinct concepts and protocols in network architecture.

### Course Objectives:

1. To discuss issues related to application-layer concepts, including network services required by applications
2. To discuss working of Network layer and Transport Layer
3. To make the students understand Multimedia networking
4. To understand why need to Network Management

### Course Outcomes

After completing the course, students will able to:

- CO1 Explain application-layer concept and working of different protocols of application layer
- CO2 Describe the principle of working of transport layer and implementation of these principles in existing protocols
- CO3 Express working of network layer and routing protocols
- CO4 Explore Architecture and Mechanisms for multimedia networking in a best-effort network
- CO5 The network Management by motivation the need for providing appropriate tools for networking administrator

### Detailed Syllabus:

#### Unit 1 Application Layer

Principles of Network Applications , The Web and HTTP , File Transfer , Electronic Mail in the Internet , DNS, Peer-to-Peer Sharing,Socket Programming With TCP.

#### Unit 2 Transport Layer

Introduction and Transport-Layer Services, Multiplexing and Demultiplexing , Connectionless Transport-UDP, Principles of Reliable Data Transfer , Connection-Oriented Transport , Principles of Congestion Control , TCP Congestion Control

#### Unit 3 The Network Layer

Introduction Forwarding and Routing, Router , The Internet Protocol (IP) , Routing Algorithms, Routing in the Internet , Broadcast and Multicast Routing.

#### Unit 4 Multimedia Networking

Multimedia Networking Applications , Streaming Stored Video , Protocols for Real-Time Interactive Applications,Scheduling and Policing Mechanisms,RSVP.

#### Unit 5 Network Management

What Is Network Management ? , The Infrastructure for Network Management, The Internet-Standard Management Framework,ASN.1.

### **Text and Reference Books**

1. “Computer Networking- a top-down approach featuring the Internet”, by James F. Kurose and Keith W. Ross, Person Education, ISBN-10- 0132856204, 6<sup>th</sup> Edition.
2. “Data Communications and Networking”, by Forouzan B. A, Tata McGraw-Hill Publications, 2006, ISBN-0-07-063414-9, 4th edition.
3. “Computer Networks”, by Tanenbaum A. S., Pearson Education , 2008, ISBN- 978-81-7758-165-2, 4th Edition
4. “Computer Networks and Internet”, by Comer D., Pearson Education, ISBN-81-297- 0330-0, 2nd Edition.
5. “Computer Networks- A Systems Approach”, by Larry L. Peterson and Bruce S. Davie, Morgan Kaufmann, ISBN-978-81-312-1045-1, 4th Edition.



## IT3045: Software Engineering and Testing

### Teaching Scheme

Lectures: 3 Hrs/Week

Total Credits :3

### Examination Scheme

Test 1 : 15 Marks

Test 2 : 15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

### Course Description

The course aims to prepare students for careers in software engineering, software project management, and software development and integration, implementation of testing levels for real time projects.

### Course Educational Objectives:

1. To understand fundamental concepts of software engineering principles.
2. To design and implement the software solutions and methodologies, good test cases
3. To understand the role and contents of testing activities in different life cycle phases.
4. To train the students on basic principles of Software Engineering used in Industry.

### Course Outcomes:

- CO1 Apply principles of software engineering with internal fundamentals.
- CO2 Identify software design solutions for appropriate processes.
- CO3 Select modern IT testing tools as per software levels.
- CO4 Apply appropriate testing tools as per engineering and management principles.
- CO5 Create planning management and bug reporting to function effectively to achieve common goal.

### Detailed Syllabus:

**UNIT-1 Introduction:** Software engineering basics, SDLC Models, Requirement Engineering, Agile methodology, Software Project Planning with Size estimation, Cost estimation Models, The constructive Cost mode COCOMO II.

**UNIT-2 Software Design**  
Design, Modularity, Strategy of design, Function oriented design, IEEE Recommended practice for Software Design Description, Object Oriented Design, Unified Modeling Language diagrams, Software metrics, data structures metrics, Information flow metrics, Metrics analysis, Software quality, software reliability Models, ISO 9000.

**UNIT-3 Basic of software testing:** Quality Concepts, Needs and Objective of testing, Software Development & Software Testing Life Cycle, Testing Standards:-IEEE, CMM, ANSI. Verification and Validation Model, Techniques of Verification, levels of testing,

**UNIT-4 Testing methods and Testing tools:** Black Box methods:-Equivalence partitioning, Boundary-value analysis, Error guessing. White Box methods:-Statement coverage, Decision coverage, Condition coverage.  
Testing Tools:-Win Runner, Load Runner.

**UNIT-5 Test Planning & Documentation:** Testing Strategy, Defect Management and Test Reporting, Software maintenance process and models, estimation of maintenance costs, reverse engineering, software re-engineering, Configuration management, documentation.

### Text books:

1. Pressman R.S., “*Software Engineering*”, McGraw-Hill Publication
2. K.K.Aggarwal, Yogesh Singh, “*Software Engineering*”, New Age International Publishers
3. Ron Patton, “*Software testing*”, Pearson Publications

**Reference Books:**

1. Ian Sommerville , “*Software Engineering*” , Pearson Education
2. Boris Bezier, “*Software testing techniques*”, Dreamtech Publications
3. Meilir Page-Jones, “ *Fundamentals of Object Oriented Design in UML*” Pearson Education
4. Rex Black , “*Software testing*”, Wrox Publications
5. Dr.K.V.K.K. Prasad, “*Software testing tools*”, Dreamtech Publications

## IT3046 : Operating System

### Teaching Scheme

Lectures: 3 Hrs/Week

Total Credits :3

### Examination Scheme

Test 1 : 15 Marks

Test 2 : 15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

### Prerequisites: IT2032 Data Structures

**Course description:** This course provides students with major functions and principles of operating systems using latest OS. Main topics include the structure of operating systems, process management, memory management, I/O management, file management.

### Course Objectives:

1. Understand structure of modern operating systems.
2. Describe the needs for OS.
3. Summarize techniques for concurrency, synchronization, storage.

CO1 Understand the concept of OS internal functions for research literature.

CO2 Examine details of process concepts and algorithms to design solutions

CO3 Apply the concept of deadlock and synchronization for professional development.

CO4 Analyze software projects for performance centric applications.

CO5 Justify I/O management and security for modern engineering techniques.

### Course Outcomes

After completing the course, students will able to:

### Detailed Syllabus:

**Unit 1 Introduction :** Operating system and functions, operational view of a computing system, issues in communication with devices, kernel and shell of an operating system Classification of Operating systems- Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multiprocessor Systems, Multithreaded Systems, Operating System Structure-Layered structure, System Components, Operating System services, System calls, Concept of Virtual machine

**Unit 2 Process Management:** Process Concepts, Schedulers, Process Control Block (PCB), Process address space, Process identification information, Threads and their management, Scheduling Concepts, Scheduling Algorithms, Multiprocessor Scheduling Inter Process Communication models and Schemes, Process generation.

**Unit 3 Process Synchronization:** Concurrent Processes, Principle of Concurrency, Producer / Consumer Problem, Mutual Exclusion, Critical Section Problem, Semaphores, Classical Problem in Concurrency- Dining Philosopher Problem, Monitors, Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.

**Unit 4 Memory Management:** Basics of memory management, memory relocation, linking and loading of memory, processes and primary memory management, Multiprogramming with fixed and variable partitions, Paging, hardware support for paging, segmentation, segmentation with paging, fragmentation, Virtual memory concepts, Demand paging, Performance of demand paging, Page

replacement algorithms, Thrashing, Cache memory organization, Locality of reference.

**Unit 5 File System, Protection & security:** File concept, File system structure, file allocation methods, Disk space management, Swap-space management, I/O Management and disk scheduling. Goals of protection, Access matrix, Implementation of access matrix, The security problem, Authentication, Programming threats, System threats, Threat monitoring, Encryption.

### **Text and Reference Books**

1. Abraham Silberschatz and Peter Barer Galvin, "*Operating System concepts*", 8th Ed. Addison Wesley, 1998
2. Andrew S. Tanenbaum, "*Modern Operating Systems*", Prentice Hall, 1991
3. D M Dhamdhare, "*Operating Systems : A Concept based Approach*", 2nd Edition

## IT3072: Enterprise Resource Planning

### Teaching Scheme

Lectures:04Hrs/Week

Tutorial:01 Hrs/Week

Credits:05

### Examination Scheme

Test-1 : 15 Marks

Test-2 :15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

### Course Objectives:

Understand necessity of ERP in organization. Learn modelling and implementation details of ERP. Learn customization issues in ERP implementations.

### Course Outcomes:

- CO1 Understanding the necessity, challenges associated with managing ERP systems.
- CO2 Learn the concept of supply chain management, relationship management.
- CO3 Students will have an understanding of many approaches that may be drawn upon in the implementation of an ERP system
- CO4 Introduce methods, formalisms and tools necessary for transferring architectural decisions to system designs and to successfully implemented systems.
- CO5 Study the role of ERP in E-commerce and E-Business Intelligence system.

### Detailed Syllabus:

- Unit I Introduction:** Definition / concept of an enterprise, Evolution of ERP, reasons for explosive growth of ERP in the market, tangible and intangible benefits of ERP systems, Limitations of ERP, Concept of business integration and how it is achieved by ERP systems, discussion on whether companies can develop their own ERP packages or should go for ERP implementation
- Unit II ERP and related technologies:** Management Information Systems, Decision Support Systems, Executive Information Systems, Introduction to Data Warehousing, Introduction to Data Mining, Concept of Online Analytical Processing, concept of Supply chain management, concept of Supply Relationship Management, concept of Customer Relationship Management, concept of Product Lifecycle Management.
- Unit III ERP implementation:** ERP implementation process, gap analysis, importance of end user training, concept of business consultants, definition of vendors, concept of domain experts, definition of end users, Concept of customization, ERP Implementation guidelines, reasons why an ERP fails, Post Implementation (Maintenance Mode).
- Unit IV ERP Market:** ERP Market: Introduction, SAP AG, Baan Company, Oracle Corporation, People Soft, JD Edwards World Solutions Company, System Software Associates, Inc. (SSA); QAD; A Comparative Assessment and Selection of ERP Packages and Modules.
- Unit V Electronic commerce & Future Directions:** Electronic commerce, role of ERP in electronic commerce, ERP Case studies, Future of ERP. Impact of Business Intelligence and Cloud Computing on ERP systems.

**Text Books:**

1. Enterprise Resource Planning: Mary Sumner, Pearson Fifth Edition, 2009
2. Enterprise Resource Planning: Alexis Leon Tata McGraw-Hill

**Reference Books:**

1. Concepts in Enterprise Resource Planning: Bret Wagner, Ellen Monk, Cengage Learning India, 2012
2. Enterprise Resource Planning Systems: Daniel E. O'Leary, Cambridge University Press

## **IT3073 : Computer Organization**

### **Teaching Scheme**

**Lectures:04Hrs/Week**

**Tutorial:01 Hrs/Week**

**Credits:05**

### **Examination Scheme**

**Test-1 : 15 Marks**

**Test-2 :15 Marks**

**Teachers Assessment : 10 Marks**

**End Semester Exam : 60 Marks**

**Prerequisite: IT2031 DEMP or Equivalent**

**Course Objectives:** Expose the ideas and techniques that define the art of computer architecture, organization, and design. Provide the students with architectural framework and foundation needed to understand future trends in the design.

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

- CO1 Know about various trends in computer design and architecture of advanced processors
- CO2 Identify where, when and how enhancements of computer performance can be accomplished with external devices.
- CO3 Perform computer arithmetic operations and will also be introduced to more recent applications of computer organization in advanced digital systems.
- CO4 Focus on Addressing modes, instruction format and perform operation on instruction set.
- CO5 Realize about issues related to instruction level execution, pipeline, and control unit operations, memory organization & optimization techniques.

### **Detailed Syllabus:**

- UNIT 1** Introduction ,Organization and Architecture , Computer Evolution and Performance , Computer Components and Function, Bus Interconnection , Cache Memory - Principles, Elements of Cache Design , Pentium 4 Cache Organization , Internal Memory Technology , Semiconductor Main Memory, Advanced DRAM Organization , Cache size vs Block size, mapping functions, replacement algorithms, write policy
- UNIT 2** Input/Output, External Devices, I/O Modules, Programmed I/O,Interrupt-Driven, I/O,Direct Memory Access , I/O Channels and Processors, The External Interface: FireWire and Infiniband
- UNIT 3** Computer Arithmetic, The Arithmetic and Logic Unit (ALU), Integer Representation- Sign Magnitude Representation, Twos Complement Representation , Converting between Different Bit Lengths, Fixed-Point Representation, Integer Arithmetic, Booths Algorithm, Floating-Point Representation, Floating-Point Arithmetic - Addition and Subtraction, Multiplication and Division , Precision Considerations , IEEE Standard for Binary Floating-Point Arithmetic
- UNIT 4** Instruction Sets: Characteristics and Functions, Machine Instruction Characteristics, Types of Operands, Intel x86 and ARM Data Types, Types of Operations, Intel x86 and ARM Operation Types, x86 and ARM Addressing Modes, Instruction Formats, x86 and ARM Instruction Formats
- UNIT 5** Processor Structure and Function : Register Organization, The Instruction Cycle, Instruction Pipelining, pipeline hazards , RISC , CISC, Instruction-Level Parallelism and Superscalar Processors, Control Unit Operation - Micro-operations, Hardwired Implementation , Microinstruction Sequencing, Microinstruction Execution, basic concepts in parallel processing & classification of parallel architectures.

**Text and Reference Books**

1. William Stallings, Computer Organization and Architecture: Designing for Performance, Pearson Education, 8th Edition

**Reference Books:**

1. David A. Patterson and John L. Hennessy, Computer Organization and Design: The Hardware/Software Interface, Elsevier.
2. Carl Hamacher, Zvonco Vranesic and Safwat Zaky, Computer Organization, McGraw Hill.
3. John P. Hayes, Computer Architecture and Organization, McGraw Hill.
4. Vincent P. Heuring and Harry F. Jordan, Computer Systems Design and Architecture, Pearson Education.

**Reference websites:**

1. <http://nptel.ac.in/courses/106103068/>
2. [https://www.tutorialspoint.com/computer\\_organization/index.asp](https://www.tutorialspoint.com/computer_organization/index.asp)
3. [http://www.cse.iitm.ac.in/~vplab/courses/comp\\_org.htm](http://www.cse.iitm.ac.in/~vplab/courses/comp_org.htm)



## IT3047: Lab Computer Algorithm

### Teaching Scheme

Practical: 2 Hrs/Week

Credit : 01

### Examination Scheme

Term Work marks : 25 Marks

Practical Examination  
& Viva Voce: : 25 Marks

### Course Outcomes

As an outcome of completing the Laboratory course, students will able to:

- CO1 Design and implement appropriate data structures for computation
- CO2 Demonstrate algorithms using divide and conquer approach
- CO3 Solve problems using greedy method.
- CO4 Apply dynamic programming techniques.
- CO5 Illustrate backtracking techniques in problem solving

### Suggestive List of Experiments

Sr. No.

Details

#### Low Level

- 1 Implement recursive and iterative algorithms for specific problems.
- 2 Implementation of sorting methods using recursion.
- 3 Write a program for finding maximum and minimum using divide and conquer approach
- 4 Implement merge sort using divide and conquer approach.

#### Medium Level (Any Five)

- 5 Implement the Greedy Knapsack problem.
  - 6 Write a program for finding minimal spanning Trees using Prim's/ Kruscal's Greedy approach.
  - 7 Write a program for finding shortest path using multistage graph problem.
  - 8 Implement the all pairs shortest path problem using dynamic programming approach.
  - 9 Implement the different techniques for tree traversals
  - 10 Implement the graph traversal techniques
- Difficult Level (Any One)
- 11 Illustrate 8-Queens problem using general backtracking method and recursive backtracking method.
  - 12 Implement a program for travelling salesperson problem algorithm using (a) Dynamic programming approach, and (b) Backtracking

## IT3048 Lab Computer Networks

**Teaching Scheme**  
**Practical: 2Hrs/Week**  
**Credits :01**

**Examination Scheme**  
**Term Work : 25 Marks**  
**Practical / Viva Voce : 25 Marks**

### Laboratory Course Outcomes

As an outcome of completing the Laboratory course, students will able to:

- CO1 Demonstrate routing algorithm
- CO2 Exercise socket programming
- CO3 Demonstrate sliding window protocols
- CO4 Design networks using NS too and Virtual Labs
- CO5 Illustrate Packet sniffing and analyze packets using Wireshark

### Suggested list of Practicals

- 1 Write a program for distance vector algorithm to find suitable path for transmission
- 2 Using TCP/IP sockets, write a client-server program file transfer
- 3 Write a program for congestion control using Leaky bucket algorithm
- 4 Write a program for implementation of simple DNS
- 5 Write a program for to simulate a sliding window protocol: Go Back N
- 6 Write a program for to simulate a sliding window protocol: Selective Repeat
- 7 Basics of Network Simulation
- 8 Simulating a Local Area Network
  - a. Study of Ethernet Frame Structure
  - b. Simulating a LAN using Network Simulator 2
- 9 Write a Program of Wireshark
  - a. Platform required
  - b. 10Installation
  - c. Packet sniffing and analysis using Wireshark
- 10 Study of Tcpdump

## **IT3049 Lab: Software Engineering and Testing**

### **Teaching Scheme**

**Practical: 2Hrs/Week**

**Credits :01**

### **Examination Scheme**

**Term Work : 25 Marks**

**Practical / Viva Voce: 25 Marks**

### **Course Outcomes:**

1. Hands on software engineering principles, tools and techniques to develop, maintain and evaluate software systems
2. Ability to design and develop efficient, reliable, robust and cost-effective software solutions
3. Ability to work as an effective member or leader of software engineering teams and meet ethical standards with legal responsibilities

### **Computer Usage / Lab Tool**

QTP/ALM tools/latest open source tools

### **Suggestive List of Experiments:**

- 1 Study Project planning through the study of problem.
- 2 Case study of Software requirement analysis.
- 3 Software design by writing test cases.
- 4 Software development and debugging using tools
- 5 Software verification and validation procedure real time example
- 6 Study of latest Testing tools QTP/ALM
- 7 Introduction to QTP/ALM tools
- 8 Real time application testing by QTP/ALM tools
- 9 Design test cases by boundary value analysis
- 10 Design test cases by equivalence partitioning
- 11 Design test cases by regression testing.

## **IT3050: Lab Operating System**

<b>Teaching Scheme</b>	<b>Examination Scheme</b>	
<b>Practical: 2 Hrs/Week</b>	<b>Term Work</b>	<b>: 25 Marks</b>
<b>Credits: 01</b>	<b>Practical Examination &amp; Viva Voce:</b>	<b>: 25 Marks</b>

### **Laboratory Course Outcomes**

As an outcome of completing the Laboratory course, students will able to:

- CO1 Apply the Knowledge of basic commands for computing requirements.
- CO2 Evaluate design of process component to implement with multithreading.
- CO3 Analyze the need of interprocess communication to select and continue with IT tool
- CO4 Understand the basic difference between primary memory and virtual memory

### **List of Experiments**

<b>Sr. No.</b>	<b>Details</b>
1	Installation Process of modern operating systems and implement various commands
2	Case study of various process states.
3	Write a program using system calls.
4	Write a Program to simulate process scheduling like FCFS, Shortest Job First and Round Robin(any two)
5	Write a program for interprocess communication using Shared memory
6	Write a program for interposes communication using pipes
7	Write a Program for deadlock avoidance(Bankers algorithm)
8	Write a Program to simulate page replacement algorithms like FIFO and LRU.
9	Write a Program to simulate memory allocation using First fit, Best fit, and Worst fit.(any two)
10	Write a program for disk scheduling

## IT3023 Lab Open Source-1

### Python Programming

Teaching Scheme

Practical: 2Hrs/Week

Credits :01

Examination Scheme

Term Work : 50 Marks

**Pre-Requisite Courses: CS1002 Lab: BCOMPIT, IT2021 OOP**

### Course Outcomes

- CO1 Understand Python especially the C and object oriented concepts with problems and design systems or processes.
- CO2 Identify the built in objects of Python and requirements appropriate to its solution
- CO3 Classify the interpretation of loop control statements.
- CO4 Justify to advanced applications such as programming for modern and IT tools.
- CO5 Create data from a text file using Python, select and apply appropriate techniques

**Unit 1** Introduction to Python and Basics of Python, all primitive and Non-Primitive variable

**Unit 2** Operators and Expressions: All type operator and expression, Decision Statements, Loop Control Statements and Function

**Unit 3** Class and Inheritance : Introduction, Class definition, Method Overriding and Overloading, Multiple Inheritance, Abstraction And Encapsulation, Polymorphism. Strings, function strings

**Unit 4** List and Multidimensional List and List Processing

**Unit 5** Sets, Tuples and Dictionary, Graphical User Interface in Python, Python- Database Access MYSQL db,

### Text and Reference Books

1. Mark Lutz, Programming Python O'REILLY.
2. John M. Zelle Python Programming: Introduction to Computer Science

### Suggested list of Practicals

#### Programming List Control Statements

- A. Write a program to display reverse of the entered number.
- B. Program to print Fibonacci series upto nth number.
- C. Write a program to display multiplication tables from 1 to 5.
- D. Read the String "Hello World" from the user. Make use of *continue* keyword and remove space.

#### Programming List on Functions

1. Program to find the factorial of a number.
2. Write a any simple program to demonstrate the use of keyword and positional argument.
3. Write a program to pass a number and return square and cube of number simultaneously.

4. Write a function `calc_Distance(x1, y1, x2, y2)` to calculate the distance between two points represented by `Point1(x1, y1)` and `Point2 (x2, y2)`. The formula for calculating distance is given below.

a. 
$$\text{Distance} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

### Programs on List, Tuple and Dictionaries

1. Consider the list with 5 different Celsius values. Convert all those Celsius values to Fahrenheit.
2. Write a Program to traverse the tuples from a list.
3. Consider the following example of tuple, `T = (1, 3, 2, 4, 6, 5)` Write a program to store numbers present at odd index into the new tuple.
4. Write a function histogram that takes string as parameter and generates a frequency of characters contained in it.

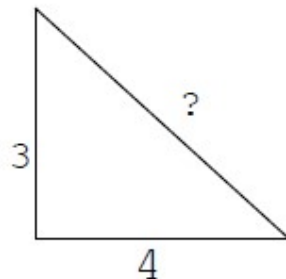
**Input:** `S = "AAPPLE"`

The Program should create a dictionary

`D= {'A': 2, 'E': 1, 'P': 2, 'L': 1}`

### Program on Module(Math Module)

1. Write a program to calculate the hypotenuse of Right angle triangle given below.



3. Programs on String: Write a program to print all the letters from `word1` that also appear in `word2`.
4. Example: `Word1 = USA North America`
  - i. `word2= USA South America`
5. Program on Class and Object Oriented Programming.
6. Write a program to calculate the volume of Box.
  - a. Note: Make use of Constructor, Member variable, and Member Functions.
7. Simple program to demonstrate the concept of Multilevel inheritance.
8. Mapping of Course outcome with Program Outcomes
9. Mapping of Course outcome with Program Outcomes

## HS 3008: Business Intelligence

Teaching Scheme		Evaluation Scheme	
Lectures	02Hrs/Week	Class Test I	15 Marks
Total Credits	02	Class Test II	15 Marks
		Teacher Assessment	10 Marks
		End-Semester Examination	60 Marks

**Course Educational Objectives:** To expose the students to the basics of business intelligence system with an understanding of modeling aspects behind Business Intelligence.

### Course Outcomes Expected:

After completing the course, students will be able to:

- CO1 Realize the need for business intelligence
- CO2 Know the business intelligence life cycle and its techniques
- CO3 Interpret different data analysis tools and techniques
- CO4 Explore modeling and analysis of data
- CO5 Identify business intelligence applications

**UNIT-1 BUSINESS INTELLIGENCE:** Effective and timely decisions – Data, information and knowledge – Role of mathematical models – Business intelligence architectures: Cycle of a business intelligence analysis – Enabling factors in business intelligence projects – Development of a business intelligence system – Ethics and business intelligence.

**UNIT-2 Decision Making, Systems, Modeling:** Introduction and Definitions, Models, Phases of the Decision-Making Process, Decision Making: The Intelligence Phase, Decision Making: The Design Phase, Decision Making: The Implementation Phase

**UNIT-3 Decision Support System:** How Decisions Are Supported, Decision Support System Configurations, Decision Support System Characteristics and Capabilities, Decision Support System Classifications, Components of Decision Support Systems

**UNIT-4 Modeling and Analysis:** Management Support Systems Modeling, Structure of Mathematical Models for Decision Support, Certainty, Uncertainty, and Risk Management Support Systems, Mathematical Programming Optimization, Spreadsheet Model, Linear Programming, Multiple Goals, Sensitivity Analysis, What-If Analysis, and Goal Seeking, Decision Analysis with Decision Tables and Decision Trees

**UNIT-5 BUSINESS INTELLIGENCE APPLICATIONS:** Digital Marketing models: Google analytics, Search Engine marketing, Email marketing, Social media marketing, Sales force management, Business case studies, Logistic and Production models – Supply chain optimization, Optimization models for logistics planning, Revenue management system, Business Case studies.

### Text Books:

1. Efraim Turban, Ramesh Sharda, Dursun Delen, "Decision Support and Business Intelligence Systems", 9th Edition, Pearson 2013.
2. Business Intelligence: Data Mining and Optimization for Decision Making Carlo Vercellis Politecnico di Milano, Italy. A John Wiley and Sons, Ltd., Publication.
3. The Art of Digital Marketing: The Definitive Guide to Creating Strategic, Targeted, and Measurable, Ian Dodson, Wiley publications

**References Books:**

1. Larissa T. Moss, S. Atre, "Business Intelligence Roadmap: The Complete Project Lifecycle of Decision Making", Addison Wesley, 2003.
2. Carlo Vercellis, "Business Intelligence: Data Mining and Optimization for Decision Making", Wiley Publications, 2009.
3. David Loshin Morgan, Kaufman, "Business Intelligence: The Savvy Manager's Guide", Second Edition, 2012.



## IT 3051: Theory of Computation

### Teaching Scheme

Lectures: 3 Hrs/Week

Tutorial: 1 Hr/ Week

Total Credits: 4

### Examination Scheme

Test 1 : 15 Marks

Test 2 : 15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

### Prerequisites:

**Course description:** This course includes introduction to theory of computation. Topics like automata theory, formal languages and grammars, Turing machines are covered in this course.

### Course Objectives:

- To provide introduction to formal languages and grammars
- To recognize deterministic and nondeterministic automata for regular languages
- To get acquainted with the use of context free languages and Push down automata
- To know Turing machines and computing with Turing machine

### Course Outcomes

After completing the course, students will able to:

- CO1 Design finite automata and its equivalent regular expressions
- CO2 Convert DFA's to NFAs and regular expression
- CO3 Formulate pushdown automata and its equivalent context free grammars
- CO4 Use pumping lemmas to prove the type of language
- CO5 Illustrate Turing machines

- Unit 1 Mathematical Preliminaries and Finite Automata:** Sets, Relations and Functions, Principle of Induction, Introduction to Finite Automata, Structural Representations, Automata and Complexity, Central Concepts to Automata Theory, Deterministic finite Automata (DFA), Nondeterministic finite Automata (NFA), Equivalence of DFA and NFA, FA with epsilon transition, Applications of FA, Moore and Mealy machines
- Unit 2 Regular Expressions and Languages:** Regular Expressions, Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Proving languages not to be regular, Closure Properties of Regular Expressions, Decision Properties of Regular Expressions, Equivalence and minimization of Automata
- Unit 3 Context Free Grammar:** Definition, Derivations using grammar, Language of a grammar, Parse Trees, Application of CFG, Ambiguity in Grammars and Languages, Normal Forms for Context Free Grammars, Pumping Lemma for CFL, Closure and Decision Properties of Context Free Languages
- Unit 4 Pushdown Automata:** Definition of Push Down Automata, Languages of Pushdown Automata, Equivalence of PDA's and CFG's, Deterministic PDA
- Unit 5 Turing Machine:** Introduction to Turing Machine, The Turing machine, Programming Techniques of Turing Machines, Extension to Basic Turing Machine, Turing Machines and Computers, Undecidable Problems about Turing Machines

### Text and Reference Books

#### Text Books:

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introduction to Automata Theory, Languages, and Computation" 3rd ed., Pearson Education, ISBN: 81-317-1429-2

**Reference Books:**

1. K.L.P. Mishra, N. Chandrasekaran, "*Theory of Computer Science: Automata, Languages and Computation*" 3<sup>rd</sup> Edition, PHI, ISBN: 978-81-203-2968-3
2. John C Martin, "*Introduction to Languages and the Theory of Computation*", 3rd ed., Tata McGraw Hill, ISBN: 0-07-066048-4

## IT3052: Advanced Database Management Systems

<b>Teaching Scheme</b>	<b>Examination Scheme</b>	
<b>Lectures: 3 Hrs/Week</b>	<b>Test 1</b>	<b>: 15 Marks</b>
<b>Total Credits: 3</b>	<b>Test 2</b>	<b>: 15 Marks</b>
	<b>Teachers Assessment</b>	<b>: 10 Marks</b>
	<b>End Semester Exam</b>	<b>: 60 Marks</b>

### Prerequisites: IT2039 Database Management System

**Course Description:** The course deals with various aspects of database design, data storage and retrieval. It also focuses on issues pertaining to concurrency in transaction management and database security. Current developments in database systems such as Distributed, Parallel, Object-Oriented and No-Sql databases along with their application are discussed. It is intended to cover several case studies relating to database design and implementation for different real-life application scenarios

### Course Educational Objectives:

- 1 Identify, describe, and categorize database objects
- 2 To learn and understand Object Oriented Databases.
- 3 To learn and understand various Database Architectures and Applications
- 4 Interpret and explain the impact of emerging database standards
- 45 Explore non-relational database systems and structures

### Course Outcomes Expected:

- CO1 Describe different database architecture and analyses the use of appropriate architecture in real time environment.
- CO2 Use modern database techniques such as NOSQL. □
- CO3 Define the role of data warehousing and enterprise intelligence in industry and government
- CO4 Design, construct and maintain a database and various database objects using procedural language

### Detailed Syllabus Contents

- UNIT1 Object and Object Relational Databases :** Concepts for Object Databases: Object Identity ,Object structure ,Type Constructors ,Encapsulation of Operations ,Methods ,Persistence, Type and Class Hierarchies Inheritance , Complex Objects ,Object Database Standards, Languages and Design: ODMG Model, ODL , OQL  
Object Relational and Extended – Relational Systems : Object Relational features in SQL/Oracle – Case Studies XML – Structure of XML, Document Schema, Querying and Transformation, API in XML, XML applications.
- UNIT2 Parallel Databases :** Database System Architectures: Centralized and Client-Server Architectures – Server System Architectures – Parallel Systems- Distributed Systems – Parallel Databases: I/O Parallelism – Inter and Intra Query Parallelism – Inter and Intra operation Parallelism

- UNIT3 Distributed Data Base :** Distributed database concepts, distributed DBMS architecture, distributed database design, top-down and bottom design, fragmentation, fragment allocation, distributed query processing, transaction management in distributed database, distributed concurrency control, reliability issues in distributed DBMS.
- UNIT4 NO SQL-**Introduction to NoSQL Database, Types and examples of NoSQL Database- Key value store, document store, graph, Performance, Structured versus unstructured data, Distributed Database Model, CAP theorem and BASE Properties, Comparative study of SQL and NoSQL, NoSQL Data Models, Case Study-unstructured data from social media.
- UNIT5 Introduction to Data Warehousing –** Concepts, Benefits and Problems, Data Warehouse Architecture – Operational Data, load manager, meta data, DW Data flows – inflow, upflow, meta flow, DW tools and technologies – Extraction, cleansing and transformation tools, DW DBMS, admin and management tools, data marts – reasons and issues, Data Warehousing using Oracle. Data Warehousing Design – Designing, Dimensionality modeling, Design methodology, DW design using Oracle. On-line Analytical Processing – OLAP BenchMarks, applications, benefits, tools, categories, extensions to SQL

#### **Text Books**

1. Silberschatz A., Korth H., Sudarshan S., "Database System Concepts", McGraw Hill Publishers, ISBN 0-07-120413-X, 6th edition
2. Pramod J. Sadalage and Martin Fowler, —NoSQL Distilled, Addison Wesley, ISBN10: 0321826620, ISBN-13: 978-0321826626
3. Paulraj Ponniah, “Data Warehousing Fundamentals, Wiley Publications”

#### **References Books:**

1. C J Date, —An Introduction to Database Systems, Addison-Wesley,
2. S.K.Singh, —Database Systems : Concepts, Design and Application, Pearson, Education, ISBN 978-81-317-6092-5
3. Kristina Chodorow, Michael Dirolf, —MangoDB: The Definitive Guide, O'Reilly Publications, ISBN: 978-1-449-34468-9
4. Adam Fowler, —NoSQL For Dummies, John Wiley & Sons, ISBN-1118905628
5. . Connally T, Begg C., "Database Systems", Pearson Education, ISBN 81-7808-861-4

## IT3053: Mobile Computing

### Teaching Scheme

Lectures: 3 Hrs/Week

Total Credits: 3

### Examination Scheme

Test 1

: 15 Marks

Test 2

: 15 Marks

Teachers Assessment

: 10 Marks

End Semester Exam

: 60 Marks

### Prerequisites: Computer Networking

#### Course Objectives:

It introduces basic terminology, fundamental concepts and principles behind design of distributed applications which can run on mobile devices and support user mobility by leveraging wireless communications. Location of mobile devices, judicious use of bandwidth, relocation of computation, are some of the key issues which will constitute the topics for further exploration. The course would be a mix of approximately 50% study of cellular network with wireless communication and 50% study of android operating system and its application development.

#### Course Outcomes:

After completing the course, students will able to:

- CO1 Understand the structure and Design issues of mobile communication systems
- CO2 Design and Study mobile IP, the extension of the Internet protocol (IP) into the mobile domain also to Ad-hoc networks with their requirements for specific routing protocols
- CO3 Implement several approaches of adapting the current transmission control protocol (TCP), which is well known from the Internet, to the special requirements of mobile communication systems
- CO4 Understand basic concept, components behind Android OS also Learn android programming to create applications for smartphones
- CO5 Create smartphone applications with different services, storage, connectivity and publishing mobile applications

#### Detailed Syllabus:

- UNIT 1** Introduction: Challenges in mobile computing, Modern wireless communication system, Cellular Concept & architecture, co-channel interference, frequency reuse, Channel assignment strategy, Handoff strategy, Interference and system capacity improvement. Evolution of mobile system: CDMA, FDMA, TDMA, SDMA, GSM. Wireless transmission, Multiplexing, Modulation, Medium access control
- UNIT 2** **Mobile Network Layer:** Mobile IP :Goals, assumptions and requirements, IP packet delivery, Agent discovery, Registration, Tunnelling and encapsulation, Optimization, Reverse tunnelling ,IPv6 IP micro-mobility support , Dynamic host configuration protocol ,Mobile ad-hoc networks, Routing, Destination sequence distance vector, Dynamic source routing, Alternative metrics, Overview ad-hoc routing protocols
- UNIT 3** **Mobile transport layer:** Traditional TCP ,Congestion control , Slow start , Mobile communications, Fast retransmit/fast recovery , Implications of mobility ,Classical TCP improvements ,Indirect TCP ,Snooping TCP ,Mobile TCP , Fast retransmit/fast recovery ,Transmission/time-out freezing, Selective retransmission , Transaction-oriented TCP , TCP over 2.5/3G wireless networks, Performance enhancing proxies .

- UNIT 4** Introduction , Android OS, History and Version, android-architecture, services, Features of Android, OS kernel, libraries and devices, Dalvik VM, Installing required software's, OS Components, Activity life cycle, intent,, Directory Structure, Android Core Building Blocks, Security Philosophy, Android platform security architecture, OS protected APIs, UI Widgets :Working with Button, Toast , Toggle, switch, image button, checkbox, spinner.
- UNIT 5** User Experience-themes and styles, material design, adaptive layouts, accessibility, localization, debugging the UI ,Storing Data, SQLite database ,Sharing Data, content resolvers and providers, Adding Google Map, Device Connectivity: Working with Camera, Wi-Fi, Bluetooth, paired devices, Animation, Services: Service lifecycle, foreground service, Android - Connecting MYSQL , Publishing Your App.

**Text and Reference Books:**

1. T. Rappaport, "Wireless Communication: Principles and Practice", Pearson Education
2. J. Schiller, "Mobile Communications", 2nd edition, Pearson Education, 2003.
3. Asoke K Talukder and Roopa R. Yavagal; Mobile Computing – Technology, Applications and Service Creation; TMH Pub., New Delhi, 2006
4. Android Programming (Big Nerd Ranch Guide), by Phillips, Stewart, Hardy and Marsicano.
5. Android Programming – Pushing the limits by Hellman
6. Wei-Meng Lee, "Beginning Android Application development", Wiley Publishing, Inc.
7. Reto Meier, "Professional Android 4 Application Development", Wiley Publishing, Inc.

**Web References:**

1. <http://freevideolectures.com/Course/2329/Wireless-Communication>
2. <http://nptel.ac.in/courses/106106147/>
3. <https://www.tutorialspoint.com/android/>

## IT3074: Linux Operating System

### Teaching Scheme

Lectures: 4 Hrs/Week

Total Credits: 4

### Examination Scheme

Test 1 : 15 Marks

Test 2 : 15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

### Prerequisite: Data structures, Operating system

**Course description:** This course provides basic functions and principles of Linux operating System. Main topics include the overview of Linux operating systems, process management, memory management, storage management, essential Linux utilities and commands, shell scripting with system programming.

### Course Objectives:

1. To study internals of Linux Operating System.
2. Understand required data structures for process management.
3. Describe system calls, memory management and interrupt handlers

### Course Outcomes

After completing the course, students will able to:

- CO1 Design solutions for engineering problems using Linux kernel specification
- CO2 Identify process scheduling for computing requirements
- CO3 Execute kernel data structure for engineering practice
- CO4 Analyze software development tools as per appropriate techniques
- CO5 Determine storage techniques to manage projects

### Detailed Syllabus:

#### Unit 1 **Introduction:**

Introduction to the Linux Kernel, Kernel Designs and Versions, Obtaining and Installing the Kernel Source, The Kernel Source Tree, Building and Configuring the Kernel, Installing the New Kernel, GNU C, Inline Functions, Synchronization and Concurrency, Importance of Portability

#### Unit 2 **Process Management:** The Process Structure, Allocating and Storing the Process Descriptor, Process States, The Linux Implementation of Threads, Creating Threads, Kernel Threads, Process Scheduling, Multitasking, Linux's Process Scheduler, The Scheduling Policy in Action, The Linux Scheduling Algorithm, Fair Scheduling, The Linux Scheduling Implementation, Preemption and Context Switching, User and kernel Preemption, Real-Time Scheduling Policies.

#### Unit 3 **System Calls and Kernel Data Structures:** Communicating with the Kernel, APIs, POSIX, and the C Library, System calls, Linked Lists, Manipulating Linked Lists, Queues Maps, Binary Trees, Data Structure to Use, Algorithmic Complexity, Time Complexity

#### Unit 4 **Interrupts and Interrupt Handlers:** Interrupts, Interrupt Handlers, Registering and Writing an Interrupt Handler, A Real-Life Interrupt Handler, Implementing Interrupt Handlers, /proc/interrupts, Interrupt Control, An Introduction to Kernel Synchronization, Causes of Concurrency, Deadlocks, Contention and Scalability, Atomic Operations, Spin Locks, Reader-Writer Spin Locks, Semaphores, Reader-Writer Semaphores, Mutexes.

Unit 5 **Memory Management:** Pages,Zones,kmalloc(),vmalloc(),Slab Layer, Statically Allocating on the Stack, High Memory Mappings, Per-CPU Allocations, The New percpu Interface, Reasons for Using Per-CPU Data, Picking an Allocation Method, The Virtual File system, Common File system Interface, File system Abstraction Layer, Unix File systems, VFS Objects and Their Data Structures, The Superblock Object and Operations, The Inode Object and Operations, Data structures of files and process.

**Text and Reference Books**

1. Linux Operating System” by “Robert Love
2. Linux: The Complete Reference”, Petersen R. L., Tata McGraw Hill, ISBN: 0- 07-05849-7, 2003
3. LINUX with Operating System Concepts” by Richard Fox ISBN: 978-1-4822-3589-0

**IT3076 : Distributed System**

**Teaching Scheme**

**Lectures: 4 Hrs/Week**

**Credits:0**

**Examination Scheme**

**Class Test 1**

**:15 Marks**

**Class Test II**

**:15 Marks**

**Teachers Assessment**

**: 10 Marks**

**End Semester Exam**

**: 60 Marks**



**Prerequisite:** Operating Systems, Computer Networking, Web Technology, Java Programming

**Course Objectives:**

- To get comprehensive knowledge of fundamentals and architecture of distributed systems.
- Understand the principles and desired properties of distributed systems.
- To examine state-of-the-art distributed systems, such as Google File System.
- To make students aware about security issues and protection mechanism for distributed environment and able to design and implement sample distributed systems.

**Course Outcomes:**

- CO1 Demonstrate knowledge of the core architectural aspects of distributed systems and apply the basic theoretical concepts and algorithms of distributed systems in problem solving.
- CO2 Identify the challenges in developing distributed applications.
- CO3 Demonstrate knowledge of details the main underlying components of distributed systems (such as RPC, file systems);
- CO4 Develop and apply knowledge of distributed systems techniques and methodologies.
- CO5 To design and implement distributed applications by considering security measures.

**Detailed Syllabus:**

**Unit 1 Introduction**

Introduction, Examples of distributed systems, Different Trends in distributed systems, Focus on Resource Sharing, Challenges. System Models: Physical models, Architectural Models, Fundamental Models. **Case Study: The World Wide Web**

**Unit 2 Communication**

Inter-process Communication, API for the Internet Protocols, External Data Representation and Marshalling, Multicast Communication, Network Virtualization: Overlay Networks, **Case Study: Java RMI** Indirect Communication: Group Communication, Publish-subscribe Systems, Message Queues, Shared Memory approaches

**Unit 3 Distributed Objects and Components** Introduction, Distributed Objects, Case Study: CORBA. From Objects to Components **Case Studies: Enterprise JavaBeans and Fractal Web Services:** Introduction, Web Services, SERVICE Descriptions and IDL for Web Services, A directory service for use with web services, XML security, Coordination of web services, Applications of Web Services.

**Unit 4 Distributed Storage & Algorithm Distributed File Systems:** Introduction, File Service Architecture, Sun Network File System, and HDFS. **Name Services:** Introduction, Name Services and the Domain Name System, Directory Service Time and Global States: Introduction, Clocks, Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States.

**Unit 5 Security In Distributed Systems** Introduction to Security: overview of Security Techniques, Design Issues, Cryptographic Algorithm, Digital Signature, **Case Study: Kerberos. Designing of Distributed Systems: Google case study**

**Text and Reference Books**

1. George Coulouris, Jean Dollimore, Tim Kindberg, & Gordon Blair, "Distributed Systems – Concept and Design", 5th Edition, Publisher: Pearson.
2. P. K. Sinha, "Distributed Operating Systems Concepts and Design", Publisher: PHI.
3. Reference Books:

4. Sunita Mahajan, Seema Shah, “Distributed Computing”, 2ndEdition, Publisher: Oxford University Press.
5. Advanced concepts in Operating Systems, Mukesh Singhal & N.G.Shivaratri, TMH.
6. Randay Chow, Theodore Johnson, “Distributed Operating System and Algorithm Analysis”, Publisher: Pearson (LPE).

### **OEIT3001: Principles Of Programming Languages**

#### **Teaching Scheme**

**Lectures: 03 Hrs/Week**

**Credits:03**

#### **Examination Scheme**

**Test I : 15 Marks:**

**Test II : 15 Marks**

**Teachers Assessment : 10 Marks**

**End Semester Exam : 60 Marks**

**Prerequisites: CS1001 Basics of Computers & IT**

**Course description:** After completing this course, students will have a broad and fundamental understanding of different programming languages. Students will be able to discuss the basic principles and implement applications using C, Java and C#.

**Course Objectives:**

- To understand the basic building blocks of programming languages.
- To learn and understand various programming paradigms

**Course Outcomes**

After completing the course, students will be able to:

- CO1 Understand the basic principles of various programming languages.
- CO2 Implement programs using Java
- CO3 Design programs using C#
- CO4 Acquire knowledge of Logical Programming
- CO5 Attain knowledge of Functional Programming

**Detailed Syllabus:**

**Unit 1 Introduction**

Role of programming languages, need to study programming languages, characteristics of a good programming languages. Introduction to various programming paradigms: Procedural, object-oriented, logic and functional programming, Parallel Programming, Concurrent Programming

**Procedural Programming:** Design Principles, Control flow: statement-oriented and block-oriented structure programming, Execution steps, desirable and undesirable characteristics of procedural programming. Procedural Programming with C: Program structure, control structures: conditional and iterative, arrays, procedures and functions, pointers, parameter passing, User defined data types

**Unit 2 Design Principles:**

Objects, classes, Messages and methods, Implementation of Object-oriented Programming, Object oriented programming with Java: Program structure, Object and class declarations, constructors, inheritance, polymorphism, access specification, interfaces, packages, exception handling, Java I/O, Java applications and applets, introduction to Java threads and multithreading

**Unit 3 Object Oriented Programming: C#**

Brief introduction to Microsoft .NET - The Microsoft .NET platform, .NET framework, advantages, introduction to C#, type system, classes, method, Properties, Arrays, Interfaces, Delegates and event handlers, Multithreading Patterns, Exception Handling.

**Unit 4 Logic Programming**

Logic programming language model, logical statements, resolution, unification, search structures: backward and forward, Applications of logic programming

**Unit 5 Functional Programming**

Introduction to functional programming, Lambda calculus: Ambiguity, free and bound identifiers, reductions, typed lambda calculus, application of functional programming

**Text Books:**

1. Roosta S., "Foundations of Programming Languages", Thomson Brookes/Cole, ISBN 981-243-141-1
2. Sethi R., "Programming Languages concepts & constructs", 2nd Edition, Pearson Education, ISBN 81 - 7808 - 104 - 0

**References Books:**

1. Scbesta R., "Concepts Of Programming Languages", 4th Edition, Pearson Education, ISBN-81-7808-161-X
2. Balagurusamy E., "Programing with C#", Tata McGraw-Hill, 2002, ISBN 0-07 -047339-0
3. Herbert Schildt "The Complete Reference Java2", 5th Edition, Tata McGraw-Hill 0 - 07-049543-2
4. Programming Language Pragmatics. Michael Scott, Morgan Kaufmann, 2000.
5. Essentials of Programming Languages. Friedman, Wand and Haynes, Prentice-Hall International (PHI), 1998.
6. Principles of Programming Languages. Tennant. PHI, 1981.

Web References:

1. <http://nptel.ac.in/courses/106102067/>
2. [https://www.tutorialspoint.com/computer\\_programming/](https://www.tutorialspoint.com/computer_programming/)
3. <http://freevidelectures.com/Course/2249/Principles-of-Programming-Languages>
4. <http://www.learnerstv.com/Free-Computer-Science-Video-lectures-ltv061-Page1.htm>

## OEIT3002: OBJECT ORIENTED MODELING and DESIGN

### Teaching Scheme

Lectures: 03 Hrs/Week

Credits:03

### Examination Scheme

Test I : 15 Marks:

Test II : 15 Marks

Teachers Assessment : 10 Marks

End Semester Exam : 60 Marks

**Course Description:** After completing this course, students will gain a good understanding of the foundation concepts of object oriented modeling techniques and tools used in object oriented modeling. Students will also gain about Rational Unified Process, Overview of all diagrams in UML, State Machine diagrams, Support for modeling Architecture in UML, Concepts of distributed operating system, etc.

### Course Objectives:

1. To understand object oriented modeling techniques.
2. To develop the ability to use tools used in object oriented modeling.
3. To develop the ability to design object oriented architecture for complex applications.
4. To understand various concepts of distributed operating system
5. To Develop the ability to use tools used in object oriented modeling

### Course Outcomes:

After completing the course, students will able to:

- CO1 Understand the theoretical concepts of OOM.
- CO2 Develop the ability to design object oriented architecture for complex applications.
- CO3 Develop the ability to use tools used in object oriented modeling.
- CO4 Master the art of using UML in object oriented modeling.

### Detailed Syllabus:

- Unit 1** Review of OO concepts. Object diagrams, Class diagrams, Classes and Relationships, Interfaces and ports, Templates, Active Objects, Advanced relationships generalization, inheritance, association, aggregation, dependencies.
- Unit 2** Introduction to UML. UML History, UML New Features. Rational Unified Process emphasizing Inception, Elaboration, Construction, Transition Phases. 4+1 View architecture, Architectural approaches: Use case Centric, Architecture driven, Iterative approach. UML Meta Model. Extensibility mechanisms like stereotypes, tagged values, constraints and profiles. OCL. Overview of all diagrams in UML.
- Unit 3** Composite structure diagrams including composite structures, collaborations. Interaction diagrams. Interaction Overview diagrams including interactions, signals, exceptions, regions, partitions, Sequence diagrams, Communication diagrams.
- Unit 4** State Machine diagrams, States, encapsulation of states, transitions, submachine, state generalization. Timing diagrams, Activity diagrams, Activities, sub activities, signals, exceptions, partitions, and regions.
- Unit 5** Support for modeling Architecture in UML. Package diagrams, Component diagrams, Deployment diagrams. Applications of UML in embedded systems, Web applications, commercial applications. All diagrams are to be assumed for UML 2.0 for each diagram the need, purpose, Concepts, Notation, Forward Engineering, Reverse Engineering & Application must be considered. Concepts of distributed operating system: COM and CORBA , Introduction to Object Oriented Database.

**Text Books:**

1. Grady Booch, James Rumbaugh, Ivar Jacobson “Unified Modeling Language User Guide”, The (2nd Edition) (Addison-Wesley Object Technology Series).
2. Rumbaugh: Object Oriented Modelling and Design, PHI

**Reference Books:**

1. Joseph Schmuller “SAMS Teach yourself UML in 24 Hours”, 3rd edition.
2. Martin Fowler, “UML Distilled: A Brief Guide to the Standard Object Modeling Language”, 3rd Edition (Paperback) ,Addison Wesley.
3. Dan Pilone, Neil Pitman “UML 2.0 in a Nutshell”, (In a Nutshell (O'Reilly)) Paperback)
4. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado “UML 2 Toolkit (Paperback)
5. Jim Arlow, Ila Neustadt “UML 2 and the Unified Process : Practical Object-Oriented Analysis and Design” (2nd Edition) (Addison-Wesley Object Technology Series)
6. Michael Jesse, James A. Schardt “UML 2.0 for dummies “

## IT3054: Lab Advanced Database Management Systems

**Teaching Scheme**  
**Practical: 2Hrs/Week**  
**Credits :01**

**Examination Scheme**  
**Term Work : 25 Marks**  
**Practical Examination :25 Marks**  
**& Viva Voce**

**Prerequisite-IT2041 Database Management Systems**

### **Laboratory Course Outcomes**

As an outcome of completing the Laboratory course, students will able to:

### **Mapping of Course outcome with Program Outcomes**

- CO1 Evaluate models/databases with respect to their accuracy.
- CO2 Discover and measure interesting patterns from different kinds of database
- CO3 Demonstrate capacity to perform a self directed piece of practical work that requires the application of data warehousing techniques
- CO4 Develop hypotheses based on the analysis of the results obtained and test them
- CO5 Design and implement a complete problem solution using current database technology.  
(Oracle 11g)

### **List of Experiments**

<b>Sr. No.</b>	<b>Details</b>
1	<b>Group A: Introduction to Databases (Study assignment )</b> Study and design a database with suitable example using following database systems: <input type="checkbox"/> Relational: SQL / PostgreSQL / MySQL <input type="checkbox"/> Key-value: Riak / Redis <input type="checkbox"/> Columnar: Hbase <input type="checkbox"/> Document: MongoDB / CouchDB <input type="checkbox"/> Graph: Neo4J Compare the different database systems based on points like efficiency, scalability, characteristics and performance. <b>Group B: SQL and PL/SQL</b>
1	Design any database with at least 3 entities and relationships between them. Apply DCL and DDL commands. Draw suitable ER/EER diagram for the system.
2	Design and implement a database and apply at least 10 different DML queries for the following task. For a given input string display only those records which match the given pattern or a phrase in the search string. Make use of wild characters and LIKE operator for the same. Make use of Boolean and arithmetic operators wherever necessary
3	Implement nested sub queries. Perform a test for set membership (in, not in), set comparison (<some, >=some, <all etc.) and set cardinality (unique, not unique
4	Write and execute suitable database triggers .Consider row level and statement level triggers.
	<b>Group C: Large Scale Databases</b>
5	Study of Open Source NOSQL Database: MongoDB (Installation, Basic CRUD operations, Execution)
6	Design and Develop MongoDB Queries using CRUD operations. (Use CRUD operations, SAVE method, logical operators)
7	Implement aggregation and indexing with suitable example using MongoDB.
8	Design and Implement any 5 query using MongoDB
9	Create simple objects and array objects using JSON
10	Encode and Decode JSON Objects using Java/Perl/PHP/Python/Ruby

**Group D:Mini Project :**

- 1 Write a program to implement MogoDB database connectivity with PHP/ python/Java Implement Database navigation operations (add, delete, edit etc. ) using ODBC/JDBC
- 2 Implement MYSQL/Oracle database connectivity with PHP/ python/Java Implement Database navigation operations (add, delete, edit,) using ODBC/JDBC.



## IT 3055 Lab: Mobile Computing

**Teaching Scheme**  
**Practical: 2Hrs/Week**  
**Credits :01**

**Examination Scheme**  
**Term Work : 25 Marks**  
**Practical Examination & Viva Voce : 25 Marks**

### Laboratory Course Outcomes

As an outcome of completing the Laboratory course, students will able to:

- CO1 Design GUI application in Android
- CO2 Construct an applications with database connectivity
- CO3 Create an application which interacts with real time data
- CO4 Design gaming applications
- CO5 Create application using modern tools

### List of Practical

**Sr. No.**

#### **Low Level**

- 1 To study Android Studio and android installation with Eclipse. Create “Hello World” application
- 2 Create login application where you will have to validate EmailID (UserName). Till the username and password is not validated , login button should remain disabled. On successful login , open browser with any URL.
- 3 Design simple GUI application with activity and intents e.g. calculator.
- 4 Create an application to handle address book.
- 5 Create an application to handle images and videos according to size.

#### **Medium Level (Any Four )**

- 6 Create an app that read messages from the mobile and display it on the screen
- 7 Design a client server application to communicate with database server.
- 8 Create an android app for database creation using SQLite Database
- 9 Develop an android application to show nearby police station/ Hotels
- 10 Design an application to locate the user and to show map information

#### **High Level (Any One )**

- 11 Design a pages with Ajax support
- 12 Develop a mini project in Android
- 13 Design a gaming app

#### **Designed by**

Prof. A. N. Ghule  
Prof. P.A.Mohrut  
Prof. K.V.Sonkambale

## IT3075 Lab: Linux Operating System

### Teaching Scheme

Practical: 2 Hrs/Week

Credits:01

### Examination Scheme

Term Work : 25 Marks

Practical Examination : 25 Marks  
& Viva Voce

### Laboratory Course Outcomes

As an outcome of completing the Laboratory course, students will able to:

- CO1 Develop essential operating system skills including how to use, setup, configure, troubleshoot and maintain a current operating system.
- CO2 Identify program structures confidently in Unix/Linux environment.
- CO3 Use shell scripts to automate various tasks.
- CO4 Understand the basics of Linux administration.

### List of Experiments

Sr. No.	Details
1	Installation of Linux OS using different utilities.
2	Shell programs using loops.
3	Shell program using patterns, and expansions.
4	Shell program using substitution and I/O system calls.
5	AWK programming for student database.
6	AWK programming using different loops.
7	Use the vi editor to create and modify file content.
8	Use simple shell features such as pipes and I/O redirection.
9	Implement semaphores, mutex.
10	Implement reader's writer's problem.

### Designed by

1. Prof. S.M.Chavan
2. Prof. U.S.Bodkhe
3. Prof.P.A.Moharut
- 4.Prof.K.Sonkamble

## IT3075 Lab: Distributed Systems

### Teaching Scheme

Practical: 2 Hrs/Week

Credits:01

### Examination Scheme

Term Work : 25 Marks

Practical Examination : 25 Marks  
& Viva Voce

### Laboratory Course Outcomes

As an outcome of completing the Laboratory course, students will able to:

- CO1 Apply Distributed System And Operating System Structures
- CO2 Perform Interprocess communication
- CO3 Implement Distributed Mutual Exclusion and Distributed Deadlock Detection algorithm
- CO4 Use Protection and Security in Distributed file systems.

### List of Experiments

Sr. No.	Details
1	Write the Programs for Remote Procedure call.
2	Write the Programs for Remote Method Invocation.
3	Write the Programs for Thread Programming in JAVA.
4	Write a Program to Increment a Counter in Shared Memory
5	Implement Network File System (NFS)
6	Study of Web Service Programming
7	Study of Grid Services using various Tools
8	Design a Distributed application using socket
9	Design a Distributed Application using Mobile agent that travels in the network and performs a given task on the targeted node
10	Find out the list of users who owns a file having maximum size in the current working directory using MapReduce Program.

**Designed by**  
Prof. K.S.Gandale