

Andhra Pradesh State Council of Higher Education  
**B.Sc. Computer Science/Information Technology (IT) Syllabus Under CBCS**  
w.e.f.2015-2016 (Modified in April 2016)

**Structure of Computer Science/Information Technology (IT) Syllabus**

| Semester                      | Paper  | Subject                                    | Hrs. | Credits | IA | ES  | Total |
|-------------------------------|--|--|------|---------|----|-----|-------|
| <b>FIRST YEAR</b>             |  |  |      |         |    |     |       |
| SEMESTER I                    | I  | Computer Fundamentals and Photoshop        | 4    | 3       | 25 | 75  | 100   |
|                               |  | Photo Shop Lab                             | 2    | 2       | 0  | 50  | 50    |
| SEMESTER II                   | II   | Programming in C                           | 4    | 3       | 25 | 75  | 100   |
|                               |  | Programming in C Lab                       | 2    | 2       | 0  | 50  | 50    |
| <b>SECOND YEAR</b>            |  |  |      |         |    |     |       |
| SEMESTER III                  | III  | Object Oriented Programming Using Java     | 4    | 3       | 25 | 75  | 100   |
|                               |  | Object Oriented Programming Using Java Lab | 2    | 2       | 0  | 50  | 50    |
| SEMESTER IV                   | IV   | Data Structures                            | 4    | 3       | 25 | 75  | 100   |
|                               |  | Data Structures using Java Lab             | 2    | 2       | 0  | 50  | 50    |
| <b>THIRD YEAR</b>             |  |  |      |         |    |     |       |
| SEMESTER V                    | V  | DBMS                                       | 3    | 3       | 25 | 75  | 100   |
|                               |  | DBMS Lab                                   | 2    | 2       | 0  | 50  | 50    |
|                               | VI   | Software Engineering                       | 3    | 3       | 25 | 75  | 100   |
|                               |  | Software Engineering Lab                   | 2    | 2       | 0  | 50  | 50    |
| SEMESTER VI                   | VII<br>(A/B/<br>C)   | <b>Elective-I</b>                          | 3    | 3       | 25 | 75  | 100   |
|                               |  | A. Operating Systems                       |      |         |    |     |       |
|                               |  | B. Computer Networks                       |      |         |    |     |       |
|                               |  | C. Web Technologies                        |      |         |    |     |       |
|                               | <b>Lab for Elective –I</b>                                       | 2  | 2    | 0       | 50 | 50  |       |
|                               | VIII<br>Cluste<br>r-A-<br>1,2,3<br>or<br>Cluste<br>r-B-<br>1,2,3 | <b>Elective-II(cluster A)</b>              | 3    | 3       | 25 | 75  | 100   |
|                               |  | 1. Foundations of Data Science             |      |         |    |     |       |
|                               |  | 2. Big Data Technology                     |      |         |    |     |       |
|                               |  | 3. Computing for Data Analytics            |      |         |    |     |       |
|                               |  | Project Work                               | 2    | 2       | 20 | 30  | 50    |
| <b>Elective-II(cluster B)</b> |  | 3  | 3    | 25      | 75 | 100 |       |
| 1. Distributed Systems        |  |  |      |         |    |     |       |
| 2. Cloud Computing            |  |  |      |         |    |     |       |
| 3. Grid computing             |  |  |      |         |    |     |       |
| Project Work                  | 2  | 2  | 20   | 30      | 50 |     |       |

**Structure of Computer Science/Information Technology (IT) Syllabus**  
**I YEAR 1 SEMESTER**

**Computer Fundamentals & Photoshop**

**Course Outcome**

To explore basic knowledge on computers and Photoshop's beauty from the practical to the painterly artistic and to understand how Photoshop will help you create your own successful images

**UNIT-I:**

Introduction to computers, characteristics and limitations of computer, Block diagram of computer, types of computers, uses of computers, computer generations. Number systems :binary, hexa and octal numbering system

**UNIT-II:**

Input and output devices: Keyboard and mouse, inputting data in other ways, Types of Software: system software, Application software, commercial, open source, domain and free ware software, Memories: primary, secondary and cache memory. Windows basics: desktop, start menu, icons.

**Unit –III**

Introduction to Adobe photoshop, Getting started with photoshop, creating and saving a document in photoshop, page layout and back ground, photoshop program window-title bar,menu bar,option bar,image window,image title bar,status bar,ruler,paletts,tool box,screen modes,saving files,reverting files,closing files.

**Unit –IV**

**Images:** working with images, image size and resolution ,image editing,colour modes and adjustments , Zooming & Panning an Image,, , Rulers, Guides & Grids- Cropping & Straightening an Image,image backgrounds ,making selections.

**Working with tool box:** working with pen tool, save and load selection-working with erasers-working with text and brushes-Colour manipulations: colour modes- Levels – Curves - Seeing Colour accurately - Patch tool – Cropping-Reading your palettes - Dust and scratches- Advanced Retouching- smoothing skin

**Unit-V**

**Layers:** Working with layers- layer styles- opacity-adjustment layers

**Filters:** The filter menu, Working with filters- Editing your photo shoot, presentation –how to create adds ,artstic filter,blur filter,brush store filter,distort filters,noise filters,pixelate filters,light effects,difference clouds,sharpen filters,printing.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**Reference Books:**

1. Fundamentals of Computers by Reema Thareja from Oxford University Press
2. Adobe Photoshop Class Room in a Book by Adobe Creative Team.
3. Photoshop: Beginner's Guide for Photoshop - Digital Photography, Photo Editing, Color Grading & Graphic...19 February 2016 by David Maxwell

**Student Activity:**

1. **Design a poster for technical paper presentation.**
2. **Create a digital scrap book.**

***Photo Shop Lab***

1. Create your Visiting card
2. Create Cover page for any text book
3. Create a Paper add for advertising of any commercial agency
4. Design a Passport photo
5. Create a Pamphlet for any program to be conducted by an organisation
6. Create Broacher for you college
7. Create Titles for any forthcoming film
8. Custom shapes creation
9. Create a Web template for your college
10. Convert color photo to black and white photo
11. Enhance and reduce the given Image size
12. Background changes
13. Design Box package cover
14. Design Texture and patterns
15. Filter effects & Eraser effects

**Structure of Computer Science/Information Technology (IT) Syllabus**  
**I YEAR II SEMESTER**

**Paper-II : PROGRAMMING IN C**

**Course Objectives**

1. Learn how to solve common types of computing problems.
2. Learn data types and control structures of C
3. Learn to map problems to programming features of C.
4. Learn to write good portable C programs.

**Course Outcomes**

Upon successful completion of the course, a student will be able to:

1. Appreciate and understand the working of a digital computer
2. Analyze a given problem and develop an algorithm to solve the problem
3. Improve upon a solution to a problem
4. Use the 'C' language constructs in the right way
5. Design, develop and test programs written in 'C'

**UNIT I**

**Introduction to Algorithms and Programming Languages:** Algorithm – Key features of Algorithms – Some more Algorithms – Flow Charts – Pseudo code – Programming Languages – Generation of Programming Languages – Structured Programming Language- Design and Implementation of Correct, Efficient and Maintainable Programs.

**Introduction to C:** Introduction – Structure of C Program – Writing the first C Program – File used in C Program – Compiling and Executing C Programs – Using Comments – Keywords – Identifiers – Basic Data Types in C – Variables – Constants – I/O Statements in C- Operators in C- Programming Examples – Type Conversion and Type Casting

**UNIT II**

**Decision Control and Looping Statements:** Introduction to Decision Control Statements – Conditional Branching Statements – Iterative Statements – Nested Loops – Break and Continue Statement – Goto Statement

**Functions:** Introduction – using functions – Function declaration/ prototype – Function definition – function call – return statement – Passing parameters – Scope of variables – Storage Classes – Recursive functions – Type of recursion – Towers of Hanoi – Recursion vs Iteration

**UNIT III**

**Arrays:** Introduction – Declaration of Arrays – Accessing elements of the Array – Storing Values in Array – Calculating the length of the Array – Operations on Array – one dimensional array for inter-function communication – Two dimensional Arrays –Operations on Two Dimensional Arrays - Two Dimensional Arrays for inter-function communication – Multidimensional Arrays – Sparse Matrices

**Strings:** Introduction –Suppressive Input – String Taxonomy – String Operations – Miscellaneous String and Character functions

**Structure of Computer Science/Information Technology (IT) Syllabus**

#### **UNIT IV**

**Pointers:** Understanding Computer Memory – Introduction to Pointers – declaring Pointer Variables – Pointer Expressions and Pointer Arithmetic – Null Pointers – Generic Pointers - Passing Arguments to Functions using Pointer – Pointer and Arrays – Passing Array to Function – Difference between Array Name and Pointer – Pointers and Strings – Array of pointers – Pointer and 2D Arrays – Pointer and 3D Arrays – Function Pointers – Array Of Function Pointer – Pointers to Pointers – Memory Allocation in C Programs – Memory Usage – Dynamic Memory Allocation – Drawbacks of Pointers

**Structure, Union, and Enumerated Data Types:** Introduction – Nested Structures – Arrays of Structures – Structures and Functions – Self referential Structures – Union – Arrays of Unions Variables – Unions inside Structures – Enumerated Data Types

#### **UNIT V**

**Files:** Introduction to Files – Using Files in C – Reading Data from Files – Writing Data from Files – Detecting the End-of-file – Error Handling during File Operations – Accepting Command Line Arguments – Functions for Selecting a Record Randomly - Remove() – Renaming a File – Creating a Temporary File

#### **REFERENCE BOOKS**

1. Introduction to C programming by REEMA THAREJA from OXFORD UNIVERSITY PRESS
2. E Balagurusamy: —COMPUTING FUNDAMENTALS & C PROGRAMMING – Tata McGraw-Hill, Second Reprint 2008, ISBN 978-0-07-066909-3.
3. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Edition Publ, 2002.
4. Henry Mullah & Huubert L.Cooper: The Spirit of C An Introduction to modern Programming, Jaico Pub. House,1996.

#### **Student Activity:**

1. Write a program for preparing the attendance particulars of students of your college at the end of semester according to following guidelines
  - a. Above 75 % promoted
  - b. Above 65% condoned
  - c. Below 65% detained
2. Write a program for creating timetable or your class taking work load of faculty into consideration.

**Structure of Computer Science/Information Technology (IT) Syllabus**  
**PROGRAMMING IN C LAB**

1. Find out the given number is perfect number or not using c program.
2. Write a C program to check whether the given number is Armstrong or not.
3. Write a C program to find the sum of individual digits of a positive integer.
4. A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to print the Fibonacci series
5. Write a C program to generate the first n terms of the Fibonacci sequence.
6. Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user.
7. Write a C program to find both the largest and smallest number in a list of integers.
8. Write a C program that uses functions to perform the following:
  - a. Addition of Two Matrices
  - b. Multiplication of Two Matrices
9. Write a program to perform various string operations
10. Write C program that implements searching of given item in a given list
11. Write a C program to sort a given list of integers in ascending order

**Structure of Computer Science/Information Technology (IT) Syllabus**  
**II YEAR III SEMESTER**

**Paper-III : OBJECT ORIENTED PROGRAMMING USING JAVA**

**Course Objectives**

As the business environment becomes more sophisticated, the software development (software engineering is about managing complexity) is becoming increasingly complex. As of the best programming paradigm which helps to eliminate complexity of large projects, Object Oriented Programming (OOP) has become the predominant technique for writing software in the past decade. Many other important software development techniques are based upon the fundamental ideas captured by object-oriented programming.

**Course Outcomes**

At the end of this course student will:

1. Understand the concept and underlying principles of Object-Oriented Programming
2. Understand how object-oriented concepts are incorporated into the Java programming language
3. Develop problem-solving and programming skills using OOP concept
4. Understand the benefits of a well structured program
5. Develop the ability to solve real-world problems through software development in high-level programming language like Java
6. Develop efficient Java applets and applications using OOP concept
7. Become familiar with the fundamentals and acquire programming skills in the Java language.

**UNIT-1**

**FUNDAMENTALS OF OBJECT – ORIENTED PROGRAMMING** :Introduction, Object Oriented paradigm, Basic Concepts of OOP, Benefits of OOP, Applications of OOP, Java features: **OVERVIEW OF JAVA LANGUAGE**: Introduction, Simple Java program structure, Java tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Command line arguments. **CONSTANTS, VARIABLES & DATA TYPES**: Introduction, Constants, Variables, Data Types, Declaration of Variables, Giving Value to Variables, Scope of variables, Symbolic Constants, Type casting, Getting Value of Variables, Standard Default values; **OPERATORS & EXPRESSIONS**.

**UNIT-II**

**DECISION MAKING & BRANCHING**: Introduction, Decision making with if statement, Simple if statement, if. Else statement, Nesting of if. else statements, the else if ladder, the switch statement, the conditional operator. **LOOPING**: Introduction, The While statement, the do-while statement, the for statement, Jumps in loops.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**CLASSES, OBJECTS & METHODS:** Introduction, Defining a class, Adding variables, Adding methods, Creating objects, Accessing class members, Constructors, Method overloading, Static members, Nesting of methods;

**UNIT-III**

**INHERITANCE:** Extending a class, Overloading methods, Final variables and methods, Final classes, Abstract methods and classes;

**ARRAYS, STRINGS AND VECTORS:** Arrays, One-dimensional arrays, Creating an array, Two – dimensional arrays, Strings, Vectors, Wrapper classes;

**INTERFACES: MULTIPLE INHERITANCE:** Introduction, Defining interfaces, Extending interfaces, Implementing interfaces, Assessing interface variables;

**UNIT-IV**

**MULTITHREADED PROGRAMMING:** Introduction, Creating Threads, Extending the Threads, Stopping and Blocking a Thread, Lifecycle of a Thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the ‘Runnable’ Interface.

**MANAGING ERRORS AND EXCEPTIONS:** Types of errors : Compile-time errors, Run-time errors, Exceptions, Exception handling, Multiple Catch Statements, Using finally statement,

**UNIT-V**

**APPLET PROGRAMMING:** local and remote applets, Applets and Applications, Building Applet code, Applet Life cycle: Initialization state, Running state, Idle or stopped state, Dead state, Display state.

**PACKAGES:** Introduction, Java API Packages, Using System Packages, Naming conventions, Creating Packages, Accessing a Package, using a Package.

**MANAGING INPUT/OUTPUT FILES IN JAVA:** Introduction, Concept of Streams, Stream classes, Byte Stream Classes, Input Stream Classes, Output Stream Classes, Character Stream classes: Reader stream classes, Writer Stream classes, Using Streams, Reading and writing files.

**Reference Books:**

1. E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw-Hill Company.



**Structure of Computer Science/Information Technology (IT) Syllabus**

2. John R. Hubbard, Programming with Java, Second Edition, Schaum's outline Series, TATA McGraw-Hill Company.
3. Deitel & Deitel. Java TM: How to Program, PHI (2007)
4. Java Programming: From Problem Analysis to Program Design- D.S Mallik
5. Object Oriented Programming Through Java by P. Radha Krishna, Universities Press (2008)

**Student Activity:**

- 1. Create a front end using JAVA for the student database created**
- 2. Learn the difference between ODBC and JDBC**

**OBJECT ORIENTED PROGRAMMING USING JAVA LAB**

1. Write a program to perform various String Operations
2. Write a program on class and object in java
3. Write a program to illustrate Function Overloading & Function Overriding methods in Java
4. Write a program to illustrate the implementation of abstract class
5. Write a program to implement Exception handling
6. Write a program to create packages in Java
7. Write a program on interface in java
8. Write a program to Create Multiple Threads in Java
9. Write a program to Write Applets to draw the various polygons
10. Write a program which illustrates the implementation of multiple Inheritance using interfaces in Java
11. Write a program to assign priorities to threads in java

**Structure of Computer Science/Information Technology (IT) Syllabus**  
**II YEAR IV SEMESTER**

**Paper-IV : DATA STRUCTURES**

**Course Objectives**

To introduce the fundamental concept of data structures and to emphasize the importance of data structures in developing and implementing efficient algorithms..

**Course Outcomes**

After completing this course satisfactorily, a student will be able to:

1. Describe how arrays, records, linked structures, stacks, queues, trees, and graphs are represented in memory and used by algorithms
2. Describe common applications for arrays, records, linked structures, stacks, queues, trees, and graphs.
3. Write programs that use arrays, records, linked structures, stacks, queues, trees, and graphs
4. Demonstrate different methods for traversing trees
5. Compare alternative implementations of data structures with respect to performance
6. Compare and contrast the benefits of dynamic and static data structures implementations
7. Describe the concept of recursion, give examples of its use, describe how it can be implemented using a stack .
8. Discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing.

**UNIT I**

**Concept of Abstract Data Types (ADTs)-** Data Types, Data Structures, Storage Structures, and File Structures, Primitive and Non-primitive Data Structures, Linear and Non-linear Data Structures.

**Linear Lists** – ADT, Array and Linked representations, Pointers.

**Arrays** – ADT, Mappings, Representations, Sparse Matrices, Sets – ADT, Operations

**Linked Lists:** Single Linked List, Double Linked List, Circular Linked List , applications

**UNIT II**

**Stacks:** Definition, ADT, Array and Linked representations, Implementations and Applications

**Queues:** Definition, ADT, Array and Linked representations, Circular Queues, Dequeues, Priority Queues, Implementations and Applications.

**UNIT III**

**Trees:** Binary Tree, Definition, Properties, ADT, Array and Linked representations, Implementations and Applications. Binary Search Trees (BST) – Definition, ADT, Operations and Implementations, BST Applications. Threaded Binary Trees, Heap trees.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**UNIT IV**

**Graphs** – Graph and its Representation, Graph Traversals, Connected Components, Basic Searching Techniques, Minimal Spanning Trees

**UNIT- V**

**Sorting and Searching:** Selection, Insertion, Bubble, Merge, Quick, Heap sort, Sequential and Binary Searching.

**REFERENCE BOOKS**

1. D S Malik, Data Structures Using C++, Thomson, India Edition 2006.
2. Sahni S, Data Structures, Algorithms and Applications in C++, McGraw-Hill, 2002.
3. SamantaD, Classic Data Structures, Prentice-Hall of India, 2001.
4. Heilman G I,. Data Structures and Algorithms with Object-Oriented Programming, Tata McGraw-Hill. 2002. (Chapters I and 14).
5. Tremblay P, and Sorenson P G, Introduction to Data Structures with Applications, Tata McGraw-Hill,

**Student activity:**

1. **Create a visible stack using C-graphics**
2. **Create a visible Queue using C-graphics**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**DATA STRUCTURES USING JAVA LAB**

1. Write a Program to implement the Linked List operations
2. Write a Program to implement the Stack operations using an array.
3. Write Programs to implement the Queue operations using an array.
4. Write Programs to implement the Stack operations using a singly linked list.
5. Write Programs to implement the Queue operations using a singly linked list.
6. Write a program for arithmetic expression evaluation
7. Write a program to implement Double Ended Queue using a doubly linked list.
8. Write a program to search an item in a given list using Linear Search and Binary Search
9. Write a program for Quick Sort
10. Write a program for Merge Sort
11. Write a program on Binary Search Tree operations(insertion, deletion and traversals)
12. Write a program for Graph traversals

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR V SEMESTER**

**Paper-V: Data Base Management System**

**Course Objective:**

Design & develop database for large volumes & varieties of data with optimized data processing techniques.

**Course Outcomes**

On completing the subject, students will be able to:

1. Design and model of data in database.
2. Store, Retrieve data in database.

**UNIT I**

**Overview of Database Management System:** Introduction, file-based system, Drawbacks of file-Based System ,Data and information, Database, Database management System, Objectives of DBMS, Evaluation of Database management System, Classification of Database Management System, DBMS Approach, advantages of DBMS, Anis/spark Data Model, data models, Components and Interfaces of Database Management System. Database Architecture, Situations where DBMS is not Necessary, DBMS Vendors and Their Products.

**UNIT II**

**Entity-Relationship Model:** Introduction, the building blocks of an entity relationship diagram, classification of entity sets, attribute classification, relationship degree, relationship classification, reducing ER diagram to tables, enhanced entity-relationship model (EER model), generalization and specialization, **IS A** relationship and attribute inheritance, multiple inheritance, constraints on specialization and generalization, aggregation and composition, entity clusters, connection types, advantages of ER modeling.

**UNIT III**

**Relational Model:** Introduction, CODD Rules, relational data model, concept of key, relational integrity, relational algebra, relational algebra operations, advantages of relational algebra, limitations of relational algebra, relational calculus, tuple relational calculus, domain relational Calculus (DRC). QBE

**UNIT IV**

**Structured Query Language:** Introduction, History of SQL Standard, Commands in SQL, Data Types in SQL, Data Definition Language, Selection Operation, Projection Operation, Aggregate functions, Data Manipulation Language, Table Modification Commands, Table Truncation, Imposition of Constraints, Join Operation, Set Operation, View, Sub Query, Embedded SQL,

**Structure of Computer Science/Information Technology (IT) Syllabus**

**UNIT V**

**PL/SQL:** Introduction, Shortcoming in SQL, Structure of PL/SQL, PL/SQL Language Elements, Data Types, Operators Precedence, Control Structure, Steps to Create a PL/SQL, Program, Iterative Control, Cursors, Steps to create a Cursors, Procedure, Function, Packages, Exceptions Handling, Database Triggers, Types of Triggers.

**Reference Books**

1. “Database System Concepts” by Abraham Silberschatz, Henry Korth, and S. Sudarshan, McGrawhill, 2010, 9780073523323
2. “Database Management Systems” by Raghu Ramakrishnan, McGrawhill, 2002,
3. Fundamentals of Relational Database Management Systems by S. Sumathi, S. Esakkirajan, Springer Publications
4. “An Introduction to Database Systems” by Bipin C Desai
5. “Principles of Database Systems” by J. D. Ullman
6. “Fundamentals of Database Systems” by R. Elmasri and S. Navathe

**Student Activity:**

1. **Create your college database for placement purpose.**
2. **Create faculty database of your college with their academic performance scores**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR V SEMESTER**

**DATABASE MANAGEMENT SYSTEMS LAB**

1. Draw ER diagrams for train services in a railway station
2. Draw ER diagram for hospital administration
3. Creation of college database and establish relationships between tables
4. Write a view to extract details from two or more tables
5. Write a stored procedure to process students results
6. Write a program to demonstrate a function
7. Write a program to demonstrate blocks, cursors & database triggers.
8. Write a program to demonstrate Joins
9. Write a program d
10. Write a program to demonstrate of Aggregate functions
11. Creation of Reports based on different queries
12. Usage of file locking table locking, facilities in applications.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR V SEMESTER**  
**Paper VI : Software Engineering**

**Course Objectives**

The Objective of the course is to assist the student in understanding the basic theory of software engineering, and to apply these basic theoretical principles to a group software development project.

**Course outcomes**

1. Ability to gather and specify requirements of the software projects.
2. Ability to analyze software requirements with existing tools
3. Able to differentiate different testing methodologies
4. Able to understand and apply the basic project management practices in real life projects
5. Ability to work in a team as well as independently on software projects

**UNIT I**

**INTRODUCTION:** Software Engineering Process paradigms - Project management - Process and Project Metrics – software estimation - Empirical estimation models - Planning - Risk analysis - Software project scheduling.

**UNIT II**

**REQUIREMENTS ANALYSIS :** Requirement Engineering Processes – Feasibility Study – Problem of Requirements – Software Requirement Analysis – Analysis Concepts and Principles – Analysis Process – Analysis Model

**UNIT III**

**SOFTWARE DESIGN:** Software design - Abstraction - Modularity - Software Architecture - Effective modular design - Cohesion and Coupling - Architectural design and Procedural design - Data flow oriented design.

**UNIT IV**

**USER INTERFACE DESIGN AND REAL TIME SYSTEMS :**User interface design - Human factors - Human computer interaction - Human - Computer Interface design - Interface design - Interface standards.

**UNIT V**

**SOFTWARE QUALITY AND TESTING :**Software Quality Assurance - Quality metrics - Software Reliability - Software testing - Path testing – Control Structures testing - Black Box testing - Integration, Validation and system testing - Reverse Engineering and Re-engineering.

CASE tools –projects management, tools - analysis and design tools – programming tools - integration and testing tool - Case studies.



**Structure of Computer Science/Information Technology (IT) Syllabus**

**REFERENCE BOOKS:**

1. Roger Pressman S., “Software Engineering: A Practitioner's Approach”, 7th Edition, McGraw Hill, 2010.
2. Software Engineering Principles and Practice by Deepak Jain Oxford University Press
2. Sommerville, “Software Engineering”, Eighth Edition, Pearson Education, 2007
3. Pfleeger, “Software Engineering: Theory & Practice”, 3rd Edition, Pearson Education, 2009
4. Carlo Ghazi, Mehdi Jazayari, Dino Mandrioli, “Fundamentals of Software Engineering”, Pearson Education, 2003

**Student Activity:**

1. Visit any financial organization nearby and prepare requirement analysis report
2. Visit any industrial organization and prepare risk chart.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR V SEMESTER**  
**Software Engineering Lab**

1. Studying various phases of Water-Fall Model.
2. Prepare SRS for Banking or On line book store domain problem
3. Using COCOMO model estimate effort for Banking or on line book store domain problem.
4. Calculate effort using FP oriented estimation model
5. Analyze the Risk related to the project and prepare RMMM plan.
6. Develop Time-line chart and project table using PERT or CPM project scheduling methods.
7. Draw E-R diagram, DFD, CFD and STD for the project.
8. Design of the test cases.
9. Prepare FTR. Version control and change control for software configuration item

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**

**Paper-VII: Elective-A**

**Operating Systems**

**Course Objectives**

1. To understand the services provided by and the design of an operating system.
2. To understand the structure and organization of the file system.
3. To understand what a process is and how processes are synchronized and scheduled.
4. To understand different approaches to memory management.
5. Students should be able to use system calls for managing processes, memory and the file system.

**Course Outcomes**

1. Analyze the concepts of processes in operating system and illustration of the scheduling of processor for a given problem instance.
2. Identify the dead lock situation and provide appropriate solution so that protection and security of the operating system is also maintained.
3. Analyze memory management techniques, concepts of virtual memory and disk scheduling.
4. Understand the implementation of file systems and directories along with the interfacing of IO devices with the operating system.

**UNIT - I**

**Operating System Introduction:** Operating Systems Objectives and functions, Computer System Architecture, OS Structure, OS Operations, Evolution of Operating Systems - Simple Batch, Multi programmed, time shared, Parallel, Distributed Systems, Real-Time Systems, Operating System services.

**UNIT - II**

Process and CPU Scheduling - Process concepts - The Process, Process State, Process Control Block, Threads, Process Scheduling - Scheduling Queues, Schedulers, Context Switch, Preemptive Scheduling, Dispatcher, Scheduling Criteria, Scheduling algorithms, Case studies: Linux, Windows.

Process Coordination - Process Synchronization, The Critical section Problem, Synchronization Hardware, Semaphores, and Classic Problems of Synchronization, Monitors, Case Studies: Linux, Windows.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**UNIT - III**

Memory Management and Virtual Memory - Logical & physical Address Space, Swapping, Contiguous Allocation, Paging, Structure of Page Table. Segmentation, Segmentation with Paging, Virtual Memory, Demand Paging, Performance of Demanding Paging, Page Replacement Page Replacement Algorithms, Allocation of Frames.

**UNIT - IV**

File System Interface - The Concept of a File, Access methods, Directory Structure, File System Mounting, File Sharing, Protection, File System Structure,

Mass Storage Structure - Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling.

**UNIT - V**

Deadlocks - System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection and Recovery from Deadlock.

**REFERENCES BOOKS:**

1. Operating System Principles, Abraham Silberchatz, Peter B. Galvin, Greg Gagne 8th Edition, Wiley Student Edition.
2. Principles of Operating Systems by Naresh Chauhan, OXFORD University Press
3. Operating systems - Internals and Design Principles, W. Stallings, 6th Edition, Pearson.
4. Modern Operating Systems, Andrew S Tanenbaum 3rd Edition PHI.
5. Operating Systems A concept - based Approach, 2nd Edition, D. M. Dhamdhare, TMH.
6. Principles of Operating Systems, B. L. Stuart, Cengage learning, India Edition.
7. Operating Systems, A. S. Godbole, 2nd Edition, TMH

**Student Activity:**

- 1. Load any new operating system into your computer.**
- 2. Partition the memory in your system**
- 3. Create a semaphore for process synchronization**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**

**Paper-VII: Elective-B**

**COMPUTER NETWORKS**

**Course Objectives**

1. To provide an introduction to the fundamental concepts on data communication and the design of computer networks.
2. To get familiarized with the basic protocols of computer networks.

**Course Outcomes**

After this course, the student will be able to

1. Identify the different components in a Communication System and their respective roles.
2. Describe the technical issues related to the local Area Networks
3. Identify the common technologies available in establishing LAN infrastructure.

**UNIT – I**

**Introduction:** Uses of Computer Networks, Network Hardware, Network Software, Reference Models, Example Networks.

**The Physical Layer:** The Theoretical Basis for Data Communication, Guided Transmission Media, Wireless transmission, the public switched telephone network

**UNIT – II**

**The Data Link Layer:** Data Link Layer Design Issues, Error Detection and Correction, Sliding Window Protocols.

**The Medium Access Control Sub-layer:** The channel allocation problem, **Multiple Access Protocols, Ethernet**, Data Link Layer Switching.

**UNIT – III**

**The Network Layer:** Network Layer Design Issues, Routing Algorithms, Congestion control algorithms, Quality of Service.

Internet Working, The Network Layer in the Internet

**UNIT – IV:**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**The Transport Layer:** The Transport Service, Elements of Transport Protocols, Congestion Control Algorithms, The Internet Transport Protocols, The Internet Transport Protocols: TCP, Delay Tolerant Networks.

**UNIT – V:**

**The Application Layer:** DNS – The Domain Name System, Electronic Mail, The World Wide Web, Real Time Audio & Video, Content Delivery & Peer-to-Peer.

**Reference Books:**

1. Andrew S. Tanenbaum, “Computer Networks”, Fifth Edition, Pearson Education.
2. Bhushan Trivedi, Computer Networks , Oxford University Press
3. James F.Kurose, Keith W.Ross, “Computer Networking”, Third Edition, Pearson Education
4. Behrouz A Forouzan, “Data Communications and Networking”, Fourth Edition, TMH (2007).
5. Kurose & Ross, “**COMPUTER NETWORKS**” – A Top-down approach featuring the Internet”, Pearson Education – Alberto Leon – Garciak.

**Student Activity:**

1. **Study the functioning of network devices available in your organization .**
2. **Prepare a pictorial chart of LAN connections in your organization**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**

**Paper-VII : Elective-C**

**Web Technologies**

**Course Objective**

- To provide knowledge on web architecture, web services, client side and server side scripting technologies to focus on the development of web-based information systems and web services.
- To provide skills to design interactive and dynamic web sites.

**Course Outcome**

1. To understand the web architecture and web services.
2. To practice latest web technologies and tools by conducting experiments.
3. To design interactive web pages using HTML and Style sheets.
4. To study the framework and building blocks of .NET Integrated Development Environment.
5. To provide solutions by identifying and formulating IT related problems.

**Unit I**

Introduction to XHTML , Cascading Style Sheets (CSS) ,JavaScript: Introduction to Scripting ,Control Statements, Functions ,Arrays ,Objects

**Unit II**

Dynamic HTML: Object Model and Collections , Dynamic HTML: Event Model

**Unit III**

XML Representing Web Data, XSL Related Technologies and Case Study

**Unit IV**

Building Ajax-Enabled Web Applications, Web Servers (IIS and Apache)

Ruby and Ruby on Rails

**Unit V**

Java Server Faces Web Applications, Web Services

**Structure of Computer Science/Information Technology (IT) Syllabus**

**References:**

1. Harvey M. Deitel and Paul J. Deitel, “**Internet & World Wide Web How to Program**”, 4/e, Pearson Education.
2. Uttam Kumar Roy, Web Technologies from Oxford University Press
3. Jason Cranford Teague “**Visual Quick Start Guide CSS, DHTML & AJAX**”, 4e, “Pearson Education.
4. Tom Nerino Doli smith “**JavaScript & AJAX for the web**” Pearson Education 2007.
5. Joshua Elchorn “**Understanding AJAX**” Prentice Hall 2006.
6. Hal Fulton “**The Ruby Way**”, 2e, Pearson Education 2007.
7. David A. Black “**Ruby for rails**” Dreamtech Press 2006.
8. Bill Dudley, Johathan Lehr, Bill Willies, Lery Mattingly “**Mastering Java Server Faces**” Wiley India 2006.

**Student Activities:**

1. Prepare a web site for your college
2. Prepare your personal website



**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**  
**(Cluster 1) Paper-VIII: Elective –A-1**  
**Foundations of Data Science**

**Course Objectives**

Modern scientific, engineering, and business applications are increasingly dependent on data, existing traditional data analysis technologies were not designed for the complexity of the modern world. Data Science has emerged as a new, exciting, and fast-paced discipline that explores novel statistical, algorithmic, and implementation challenges that emerge in processing, storing, and extracting knowledge from Big Data.

**Course Outcomes**

1. Able to apply fundamental algorithmic ideas to process data.
2. Learn to apply hypotheses and data into actionable predictions.
3. Document and transfer the results and effectively communicate the findings using visualization techniques.

**UNIT I**

**INTRODUCTION TO DATA SCIENCE** :Data science process – roles, stages in data science project – working with data from files – working with relational databases – exploring data – managing data – cleaning and sampling for modeling and validation – introduction to NoSQL.

**UNIT II**

**MODELING METHODS** :Choosing and evaluating models – mapping problems to machine learning, evaluating clustering models, validating models – cluster analysis – K-means algorithm, Naïve Bayes – Memorization Methods – Linear and logistic regression – unsupervised methods.

**UNIT III**

**INTRODUCTION TO R Language**:Reading and getting data into R – ordered and unordered factors – arrays and matrices – lists and data frames – reading data from files – probability distributions – statistical models in R - manipulating objects – data distribution.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**UNIT IV**

**MAP REDUCE:** Introduction – distributed file system – algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce – Hadoop - Understanding the Map Reduce architecture - Writing Hadoop Map Reduce Programs - Loading data into HDFS - Executing the Map phase - Shuffling and sorting - Reducing phase execution.

**UNIT V**

**DELIVERING RESULTS :**Documentation and deployment – producing effective presentations– Introduction to graphical analysis – plot() function – displaying multivariate data – matrix plots – multiple plots in one window - exporting graph - using graphics parameters. Case studies.

**Reference Books**

- 1.Nina Zumel, John Mount, “Practical Data Science with R”, Manning Publications, 2014.
- 2.Jure Leskovec, Anand Rajaraman, Jeffrey D.Ullman, “Mining of Massive Datasets”, Cambridge University Press, 2014.
- 3.Mark Gardener, “Beginning R - The Statistical Programming Language”, John Wiley & Sons, Inc., 2012.
- 4.W. N. Venables, D. M. Smith and the R Core Team, “An Introduction to R”, 2013.
- 5.Tony Ojeda, Sean Patrick Murphy, Benjamin Bengfort, Abhijit Dasgupta, “Practical Data Science Cookbook”, Packt Publishing Ltd., 2014.
- 6.Nathan Yau, “Visualize This: The FlowingData Guide to Design, Visualization, and Statistics”, Wiley, 2011.
- 7.Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley, ISBN: 9788126551071, 2015.

**Student Activity:**

1. **Collect data from any real time system and create clusters using any clustering algorithm**
2. **Read the student exam data in R perform statistical analysis on data and print results.**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**  
**(Cluster 1) Paper-VIII : Elective –A-2**

**BIG DATA TECHNOLOGY**

**Course Objective**

The Objective of this course is to provide practical foundation level training that enables immediate and effective participation in big data projects. The course provides grounding in basic and advanced methods to big data technology and tools, including MapReduce and Hadoop and its ecosystem.

**Course Outcome**

1. Learn tips and tricks for Big Data use cases and solutions.
2. Learn to build and maintain reliable, scalable, distributed systems with Apache Hadoop.
3. Able to apply Hadoop ecosystem components.

**UNIT I**

**INTRODUCTION TO BIG DATA:**Introduction – distributed file system – Big Data and its importance, Four V's in bigdata, Drivers for Big data, Big data analytics, Big data applications. Algorithms using map reduce, Matrix-Vector Multiplication by Map Reduce.

**UNIT II**

**INTRODUCTION HADOOP :** Big Data – Apache Hadoop & Hadoop EcoSystem – Moving Data in and out of Hadoop – Understanding inputs and outputs of MapReduce - Data Serialization.

**UNIT- III**

**HADOOP ARCHITECTURE:** Hadoop Architecture, Hadoop Storage: HDFS, Common Hadoop Shell commands , Anatomy of File Write and Read., NameNode, Secondary NameNode, and DataNode, Hadoop MapReduce paradigm, Map and Reduce tasks, Job, Tasktrackers - Cluster Setup – SSH & Hadoop Configuration – HDFS Administering – Monitoring & Maintenance.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**UNIT-IV**

**HADOOP ECOSYSTEM AND YARN :**Hadoop ecosystem components - Schedulers - Fair and Capacity, Hadoop 2.0 New Features- NameNode High Availability, HDFS Federation, MRv2, YARN, Running MRv1 in YARN.

**UNIT-V**

**HIVE AND HIVEQL, HBASE:-**Hive Architecture and Installation, Comparison with Traditional Database, HiveQL - Querying Data - Sorting And Aggregating, Map Reduce Scripts, Joins & Subqueries, HBase concepts- Advanced Usage, Schema Design, Advance Indexing - PIG, Zookeeper - how it helps in monitoring a cluster, HBase uses Zookeeper and how to Build Applications with Zookeeper.

**Reference Books**

1. Boris lublinsky, Kevin t. Smith, Alexey Yakubovich, “Professional Hadoop Solutions”, Wiley, ISBN: 9788126551071, 2015.
2. Chris Eaton, Dirk deroos et al. , “Understanding Big data ”, McGraw Hill, 2012.
3. Tom White, “HADOOP: The definitive Guide” , O Reilly 2012.
4. Vignesh Prajapati, “Big Data Analytics with R and Haoop”, Packet Publishing 2013.
5. Tom Plunkett, Brian Macdonald et al, “Oracle Big Data Handbook”, Oracle Press, 2014.
6. Jy Liebowitz, “Big Data and Business analytics”,CRC press, 2013.

**Student Activity:**

1. Collect real time data and justify how it has become Big Data
2. Reduce the dimensionality of a big data using your own map reducer

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**  
**(Cluster 1 Paper-VIII : Elective –A-3)**

**COMPUTING FOR DATA ANALYTICS**

**Course Objectives**

The objective of this course is to teach fundamental concepts and tools needed to understand the emerging role of business analytics in Organizations.

**Course Outcomes**

1. Learn the Big Data in Technology Perspective.
2. Understanding of the statistical procedures most often used by practicing engineers
3. Understand Forecasting methods and apply for business applications.

**UNIT – I**

**DATA ANALYTICS LIFE CYCLE: Introduction** to Big data Business Analytics - State of the practice in analytics role of data scientists - Key roles for successful analytic project - Main phases of life cycle - Developing core deliverables for stakeholders.

**UNIT – II**

**STATISTICS Sampling Techniques** : Data classification, Tabulation, Frequency and Graphic representation - Measures of central value - Arithmetic mean, Geometric mean, Harmonic mean, Mode, Median, Quartiles, Deciles, Percentile - Measures of variation – Range, IQR, Quartile deviation, Mean deviation, standard deviation, coefficient variance, skewness, Moments & Kurtosis.

**UNIT – III**

**PROBABILITY AND HYPOTHESIS TESTING: Random** variable, distributions, two dimensional R.V, joint probability function, marginal density function. Random vectors - Some special probability distribution - Binomial, Poison, Geometric, uniform, exponential, normal, gamma and Erlang. Multivariate normal distribution - Sampling distribution – Estimation - point, confidence – Test of significance, 1& 2 tailed test, uses of t-distribution, F-distribution,  $\chi^2$  distribution.

**UNIT – IV**

**PREDICTIVE ANALYTICS:** Predictive modeling and Analysis - Regression Analysis, Multicollinearity , Correlation analysis, Rank correlation coefficient, Multiple correlation, Least square, Curve fitting and good ness of fit.

**UNIT – V**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**TIME SERIES FORECASTING AND DESIGN OF EXPERIMENTS** :Forecasting Models for Time series : MA, SES, TS with trend, season - Design of Experiments, one way classification, two way classification, ANOVA, Latin square, Factorial Design.

**Reference Books**

1. Chris Eaton, Dirk Deroos, Tom Deutsch et al., “Understanding Big Data”, McGrawHill, 2012.
2. Alberto Cordoba , “Understanding the Predictive Analytics Lifecycle”, Wiley, 2014.
3. Eric Siegel, Thomas H. Davenport , “Predictive Analytics: The Power to Predict Who Will Click, Buy, Lie, or Die”, Wiley, 2013.
4. James R Evans, “Business Analytics – Methods, Models and Decisions”, Pearson 2013.
5. R. N. Prasad, Seema Acharya, “Fundamentals of Business Analytics”, Wiley, 2015.
6. S M Ross, “Introduction to Probability and Statistics for Engineers and Scientists”, Academic Foundation, 2011.
7. David Hand, Heiki Mannila, Padhria Smyth, “Principles of Data Mining”, PHI 2013.
8. Spyros Makridakis, Steven C Wheelwright, Rob J Hyndman, “Forecasting methods and applications”, Wiley 2013( Reprint).

**Student Activity:**

1. **Collect temperatures of previous months and prepare a logic to estimate the temperature of next one week**
2. **Collect real time data and apply statistical techniques to classify it.**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**  
**(Cluster 2) Paper-VIII : Elective –B-1**

**Distributed Systems**

**Course Objectives**

- To expose the fundamentals of distributed computer systems, assuming the availability of facilities for data transmission.
- To discuss multiple levels of distributed algorithms, distributed file systems, distributed databases, security and protection.

**Course Outcomes**

- Create models for distributed systems.
- Apply different techniques learned in the distributed system.

**UNIT I**

Introduction to Distributed Computing Systems, System Models, and Issues in Designing a Distributed Operating System, Examples of distributed systems.

**UNIT II**

Features of Message Passing System, Synchronization and Buffering, Introduction to RPC and its models, Transparency of RPC, Implementation Mechanism, Stub Generation and RPC Messages, Server Management, Call Semantics, Communication Protocols and Client Server Binding.

**UNIT III**

Introduction, Design and implementation of DSM system, Granularity and Consistency Model, Advantages of DSM, Clock Synchronization, Event Ordering, Mutual exclusion, Deadlock, Election Algorithms.

**UNIT IV**

Task Assignment Approach, Load Balancing Approach, Load Sharing Approach, Process Migration and Threads.

**UNIT V**

File Models, File Accessing Models, File Sharing Semantics, File Caching Schemes, File Replication, Atomic Transactions, Cryptography, Authentication, Access control and Digital Signatures.

**Structure of Computer Science/Information Technology (IT) Syllabus**

**Reference Books**

- 1.Pradeep. K. Sinha: “ Distributed Operating Systems: Concepts and Design ” , PHI, 2007.
- 2 .George Coulouris, Jean Dollimore, Tim Kindberg: “ Distributed Systems” , Concept and Design, 3<sup>rd</sup> Edition, Pearson Education, 2005.

**Student Activity**

1. Implementation of Distributed Mutual Exclusion Algorithm.
2. Create a Distributed Simulation Environment.



**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**  
**(Cluster 2 ) Paper-VIII : Elective –B-2**

**Cloud Computing**

**Course Objectives:**

The student will learn about the cloud environment, building software systems and components that scale to millions of users in modern internet, cloud concepts capabilities across the various cloud service models including IaaS, PaaS, SaaS, and developing cloud based software applications on top of cloud platforms.

**Course Outcomes**

1. Compare the strengths and limitations of cloud computing
2. Identify the architecture, infrastructure and delivery models of cloud computing
3. Apply suitable virtualization concept.
4. Choose the appropriate cloud player , Programming Models and approach.
5. Address the core issues of cloud computing such as security, privacy and interoperability
6. Design Cloud Services and Set a private cloud

**Unit 1**

**Cloud Computing Overview** – Origins of Cloud computing – Cloud components - Essential characteristics – On-demand self-service , Broad network access , Location independent resource pooling , Rapid elasticity , Measured service

**Unit II**

Cloud scenarios – Benefits: scalability , simplicity , vendors ,security. Limitations – Sensitive information - Application development – Security concerns - privacy concern with a third party - security level of third party - security benefits Regularity issues: Government policies

**Unit III**

**Cloud architecture:** Cloud delivery model – SPI framework , SPI evolution , SPI vs. traditional IT Model

**Software as a Service (SaaS):** SaaS service providers – Google App Engine, Salesforce.com and google platform – Benefits – Operational benefits - Economic benefits – Evaluating SaaS

**Platform as a Service ( PaaS ):** PaaS service providers – Right Scale – Salesforce.com – Rackspace – Force.com – Services and Benefits

**Unit IV**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**Infrastructure as a Service ( IaaS):** IaaS service providers – Amazon EC2 , GoGrid – Microsoft soft implementation and support – Amazon EC service level agreement – Recent developments – Benefits

**Cloud deployment model :** Public clouds – Private clouds – Community clouds - Hybrid clouds - Advantages of Cloud computing

**Unit V**

**Virtualization:** Virtualization and cloud computing - Need of virtualization – cost , administration , fast deployment , reduce infrastructure cost - limitations

**Types of hardware virtualization:** Full virtualization - partial virtualization - para virtualization

**Desktop virtualization:** Software virtualization – Memory virtualization - Storage virtualization – Data virtualization – Network virtualization

**Microsoft Implementation:** Microsoft Hyper V – Vmware features and infrastructure – Virtual Box - Thin client

**Reference Books**

1. Cloud computing a practical approach - Anthony T.Velte , Toby J. Velte Robert Elsenpeter TATA McGraw- Hill , New Delhi - 2010
2. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008
3. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.
4. Cloud Computing, A Hands on approach, Arshadeep Bahga, Vijay Madiseti, University Press
5. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christenvecctiola, S Tammarai selvi, TMH

**Student Activity:**

1. Prepare the list of companies providing cloud services category wise.
2. Create a private cloud using local server

**Structure of Computer Science/Information Technology (IT) Syllabus**

**III YEAR VI SEMESTER**  
**(Cluster 2) Paper-VIII : Elective –B-3**

**Grid Computing**

**Course Objectives:**

The student will learn about the Grid environment, building software systems and components that scale to millions of users in modern internet, Grid concepts capabilities across the various Grid services..

**Course Outcomes**

1. Compare the strengths and limitations of Grid computing
2. Identify the architecture, infrastructure and delivery models of Grid computing
3. Apply suitable virtualization concept.
4. Address the core issues of Grid computing such as security, privacy and interoperability

**UNIT I**

**CONCEPTS AND ARCHITECTURE** :Introduction-Parallel and Distributed Computing-Cluster Computing-Grid Computing- Anatomy and Physiology of Grid- Web and Grid Services-Grid Standards - OGSA-WSRF - Trends, Challenges and applications.

**UNIT II**

**GRID MONITORING** :Grid Monitoring Architecture (GMA) - An Overview of Grid Monitoring Systems- R-GMA –Grid ICE – MDS- Service Level Agreements (SLAs) -Other Monitoring Systems- Ganglia, Grid Mon, Hawkeye and Network Weather Service.

**UNIT III**

**GRID SECURITY AND RESOURCE MANAGEMENT**: Grid Security-A Brief Security Primer-PKI-X509 Certificates-Grid Security-Grid Scheduling and Resource Management, Grid way and Grid bus Broker-principles of Local Schedulers- Overview of Condor, SGE, PBS, LSF -Grid Scheduling with QoS.

**UNIT IV**

**DATA MANAGEMENT AND GRID PORTALS** :Data Management-Categories and Origins of Structured Data-Data Management Challenges-Architectural Approaches-Collective Data Management Services-Federation Services-Grid Portals-Generations of Grid Portals.

**UNIT V**

**Structure of Computer Science/Information Technology (IT) Syllabus**

**GRID MIDDLEWARE:** List of globally available Middleware's - Case Studies-Recent version of Globus Toolkit and gLite - Architecture, Components and Features. Features of Next generation grid.

**Reference Books**

1. Ian Foster, Carl Kesselman, The Grid 2: Blueprint for a New Computing Infrastructure, Elsevier Series, 2004.
2. Vladimir Silva, Grid Computing for Developers, Charles River Media, January 2006.
3. Parvin Asadzadeh, Rajkumar Buyya, Chun Ling Kei,Deepa Nayar, and Srikumar Venugopal, Global Grids and Software Toolkits: A Study of Four Grid Middleware Technologies, High Performance Computing : Paradigm and Infrastructure, Laurence Yang and Minyi Guo (editor s), Wiley Press, New Jersey, USA, June 2005.
4. Jarek Nabrzyski, Jennifer M. Schopf, Jan Weglarz , Grid Resource Management: State of the Art and Future Trends , (International Series in Operations Research & Management Science), Springer; First edition, 2003

**Student Activity:**

1. Implement and analyze any one Grid Resource Sharing algorithm.
2. Listout various security issues with Grid

**Structure of Computer Science/Information Technology (IT) Syllabus**

**PROJECT & VIVA-VOCE**

The objective of the project is to motivate them to work in emerging/latest technologies, help the students to develop ability, to apply theoretical and practical tools/techniques to solve real life problems related to industry, academic institutions and research laboratories.

The project is of 2 hours/week for one (semester VI) semester duration and a student is expected to do planning, analyzing, designing, coding, and implementing the project. The initiation of project should be with the project proposal. The synopsis approval will be given by the project guides.

The project proposal should include the following:

- Title
- Objectives
- Input and output
- Details of modules and process logic
- Limitations of the project
- Tools/platforms, Languages to be used
- Scope of future application

The Project work should be either an individual one or a group of not more than three members and submit a project report at the end of the semester. The students shall defend their dissertation in front of experts during viva-voce examinations.