SEMESTER I

Course code: DMA25002

DSC 3A: Computer Science-I

Computer System Organization and Architecture

Credits: Theory – 04 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Deliberate in depth Computer Fundamentals

CO2. Understand in details with features Computer Organization

CO3. Deliberate in details with examples Boolean algebra and logic circuits

CO4. Learn the details of Data Representation and Computer Arithmetic

CO5. Learn in depth Computer Organization and Design

CO6. Learn the details of architecture of CPU

CO7. Deliberate the classification and characteristics of Basic Computer Programming Concepts

CO8. Write down in depth Basic Computer Programming Concepts

CO9. Learn the classification and characteristics of Input -Output organization

Unit - 1 **15Hours**

Computer Fundamentals: Introduction to Computers: Characteristics of Computers, Uses of computers, Types and generations of Computers.

Basic Computer Organization: Units of a computer, CPU, ALU, memory hierarchy, registers, I/O devices.

Introduction: Logic gates, Boolean algebra, combinational circuits, circuit simplification, flip-flops and sequential circuits, decoders, multiplexors, registers, counters and memory units.

Unit - 2 **15 Hours**

Data Representation and basic Computer Arithmetic: Number systems, complements. Fixed and floating point representation, character representation, addition, subtraction, magnitude comparison.

Basic Computer Organization and Design: Computer registers, bus system, instruction set.

Unit - 3 **15 Hours**

Central Processing Unit: Register organization, arithmetic and logical micro-operations, stack organization, micro programmed control.

Timing and control - instruction cycle, memory reference, input-output and interrupt.

Unit - 4 **15 Hours**

Programming the Basic Computer: Instruction formats, addressing modes, instruction codes, machine language, and assembly language, input output programming.
Input-output Organization: Peripheral devices, I/O interface, Modes of data transfer, direct memory access.

Reference:

1. M. Mano, Computer System Architecture, Pearson Education 1992.
2. A. J. Dos Reis, Assembly Language and Computer Architecture using C++ and JAVA, Course Technology, 2004
3. W. Stallings, Computer Organization and Architecture Designing for Performance, 8th Edition, Prentice Hall of India,2009
4. Digital Design, M.M. Mano, Pearson Education Asia, 1979

SEMESTER I

Course code: DMA25102

DSC 3A: Computer Science-I

Computer System Organization and Architecture Lab

Credits: Practical – 02 **60 Hours**

1. Verification of Basic gates(AND,OR,NOT)
2. Verification of Universal gates(NAND,NOR,EX-OR)
3. Verification of NAND gate as a Universal gate
4. Verification of NOR gate as a Universal gate
5. Verification of Demorgon’s theorem
6. Verification of Half adder & Full Adder
7. Verification of Half subtractor& Full Subtractor
8. Verification of Half adder & Half subtractor using NAND gate
9. Conversion of Binary to Gray &gray to Binary Code
10. Simplification of Boolean Expressions
11. Simplification of Boolean Expressions using K-Map
12. Flip-Flops:SR FF(clock, without clock)
13. JK FF
14. Toggle FF
15. Delay FF
16. Multiplexer
17. De-multiplexer
18. Simulate the machine for the following memory-reference instructions with I= 0 and address part = 082. The instruction to be stored at address 022 in RAM. Initialize thememory word at address 082 with the operand B8F2 and AC with A937. Determine thecontents of AC, DR, PC, AR and IR in
Hexadecimal after the execution.

a. ADD
b. AND
c. LDA
d. STA
e. BUN
f. BSA
g. ISZ

19. Simulate the machine for the memory-reference instructions referred in above question with I= 1 and address part = 082. The instruction to be stored at address 026 in RAM. Initialize the memory word at address 082 with the value 298. Initialize the memory word at address 298 with operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the execution.

20. Modify the machine created in Practical 1 according to the following instruction format:

Instruction format

0 2	3	4 15
Opcode	I	Address

a. The instruction format contains a 3-bit opcode, a 1-bit addressing mode and a 12-bitAddress. There are only two addressing modes, I = 0 (direct addressing) and I = 1 (indirect addressing).
b. Create a new register I of 1 bit.
c. Create two new microinstructions as follows:
   i. Check the opcode of instruction to determine type of instruction (Memory Reference/Register Reference/Input-Output) and then jump accordingly.
   ii. Check the I bit to determine the addressing mode and then jump accordingly.

SEMESTER II

Course code: DMB25002

DSC 3B: Computer Science-II

Problem Solving Using Computer

Credits: Theory – 04 **60 Hours**

COURSE OUTCOME:

After successful completion of the course, the student is able to

CO1. Learn the details of Problem Solving Techniques
CO2. Deliberate the characteristics of Programming Languages
CO3. Understand in details with examples - Python Programming Languages
CO4. Specify in depth OOPs, Event Driven and GUI features in Python

Unit - 1 **15 Hours**

Programming Languages: Introductions, Types and its characteristics
Techniques of Problem Solving: Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation, Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming.

Unit - 2 **15 Hours**

Overview of Programming: Structure of a Python Program, Elements of Python
Introduction to Python: Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator, Increment or Decrement operator). Input and Output Statements, Control statements (Looping- while Loop, for Loop, Loop Control)

Unit - 3 **15 Hours**

Creating Python Programs: Conditional Statement- if...else, Difference between break, continue and pass.
Structures: Numbers, Strings, Lists, Tuples, Dictionary, Date & Time, Modules, Defining Functions, Exit function, default arguments.

Unit - 4 **15Hours**

Introduction to Advanced Python: Objects and Classes, Inheritance, Regular Expressions, Event Driven Programming, GUI Programming.

Reference:

1. P. K. Sinha & PritiSinha, “Computer Fundamentals”, BPB Publications, 2007.
2. Dr. Anita Goel, Computer Fundamentals, Pearson Education, 2010.
3. T. Budd, Exploring Python, TMH, 1st Ed, 2011
4. Python Tutorial/Documentation www.python.or 2010
5. Allen Downey, Jeffrey Elkner, Chris Meyers, How to think like a computer scientist: learning with Python, Freely available online.2012

SEMESTER II

Course code: DMB25102

DSC 3B: Computer Science-II

Problem Solving Using Computer

Credits: Practical – 02 **60 Hours**

Software Lab using Python

Section: A (Simple programs)

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user’s choice.
2. WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria

   Grade A: Percentage >=80
   Grade B: Percentage>=70 and <80
   Grade C: Percentage>=60 and <70
   Grade D: Percentage>=40 and <60
   Grade E: Percentage<40
3. Write a menu-driven program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.
4. WAP to display the first n terms of Fibonacci series.
5. WAP to find factorial of the given number
6. WAP to find sum of the following series for n terms: 1 – 2/2! : 3/3! - - - - - n/n!
7. WAP to calculate the sum and product of two compatible matrices.

Section: B (Visual Python)

All the programs should be written using user defined functions, wherever possible.

1. Write a menu-driven program to create mathematical 3D objects
    - I. Curve
    - II. Sphere
    - III. Cone
    - IV. Arrow
    - V. Ring
    - VI. Cylinder.
2. WAP to read n integers and display them as a histogram.
3. WAP to display sine, cosine, polynomial and exponential curves.
4. WAP to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.
5. WAP to calculate the mass m in a chemical reaction. The mass m (in gms) disintegrates according to the formula m=60/(t:2), where t is the time in hours. Sketch a graph for t vs. m, where t>=0.
6. A population of 1000 bacteria is introduced into a nutrient medium. The population p grows as follows:

   P(t) = (15000(1:t))/(15: e)

   Where the time t is measured in hours. WAP to determine the size of the population at given time t and plot a graph for P vs t for the specified time interval.
7. Input initial velocity and acceleration, and plot the following graphs depicting equations of motion:

I. Velocity wrt time (v=u: at)

II. Distance wrt time (s=u*t:0.5*a*t*t)

III. Distance wrt velocity (s=(v*v-u*u)/2*a)

SEMESTER III

Course code: DMC25002

DSC 3C: Computer Science-III

Data Structures and File Processing

Credits: Theory – 04 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Learn the classification, characteristics and understanding of Data structures
CO2. Specify the details of Searching Techniques
CO3. Deliberate in details with examples Basic Concepts of Memory Management Techniques
CO4. Understand in depth File System Operations
CO5. Specify the characteristics of File Organization Methods
CO6. Deliberate in details with examples of Storage Devices

Unit - 1 **15 Hours**

Basic Data Structures: Introduction- Primitive and non-primitive, Abstract data structure, Operations, Data representation, Arrays - Memory representation of one and two dimensional arrays, Stack – Operations, Applications – Recursion, infix to postfix conversion, evaluation of postfix expression, Queues – Operations, Applications, circular queue-Operations, De-queue, priority queue – uses of priority queues,

Unit - 2 **15 Hours**

Pointers and operations, Dynamic memory allocation: Linked list - Singly linked list – Operations, Circular linked list – Operations, Applications of linked list, doubly linked list – memory representation and Operations.

Trees: terminologies, Types, Memory Representation (using Arrays and Linked List) and Operations on binary trees, Introduction to Graphs

Unit - 3 **15 Hours**

Searching and sorting – sequential and binary search, sorting - bubble, insertion, quick sort and merge sort, comparison of different sorting techniques, Memory management: Garbage collection algorithm for equal sized blocks, storage allocation of objects with mixed size, buddy system

Unit - 4 **15 Hours**

Physical Devices: Characteristics of storage devices such as disks and tapes, I/O buffering.

Basic File System & Operations: Introduction to File, Create, open, close, extend/Modification, delete, read-block, write-block, protection mechanisms.

File Organizations: Sequential, indexed sequential, direct, inverted, multi-list, directory systems,

Reference Books:

1. M.T. Goodrich, R. Tamassia and D. Mount, Data Structures and Algorithms in C++, John Wiley and Sons, Inc., 2004.
2. T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein, Introduction to Algorithms, 2nd Ed., Prentice-Hall of India, 2006.
3. Robert L. Kruse and A.J. Ryba, Data Structures and Program Design in C++, Prentice Hall, Inc., NJ, 1998.
4. B. Stroupstrup, The C++ Programming Language, Addison Wesley, 2004
5. D.E. Knuth, Fundamental Algorithms (Vol. I), Addison Wesley, 1997
6. E.Horowiz and S.Sahani, *Fundamentals of Data structures*, Galgotia Book source Pvt. Ltd., 2003.

SEMESTER III

Course code: DMC25102

DSC 3C: Computer Science-III

Data Structures and File Processing Lab

Credits: Practical – 02 **60 Hours**

Part - A

1. Program to find lower triangular and upper triangular matrices for the given matrix.
2. Write an interactive program to insert an element at the given position and delete an element at the specified position in the given array.
3. Program to search an element identifies the number of occurrences with locations in linear array.
4. Program to sort the given M x N matrix row-wise and column-wise using bubble sorting technique.
5. Write an interactive program to search an element in the given linear array using linear and binary searching technique.
6. Write a program to merge two sorted arrays.

Part – B

7. Write an interactive program to implement the following operations on stack.
   a. PUSH b. POP
8. Program to implement Tower of Hanoi problem.
9. Write an interactive program to perform insertion and deletion operations in Linear Queue using arrays.
10. Write an interactive program to perform insertion and deletion operations in Circular Queue using arrays.
11. Write an interactive program to insert a node in a linked list at the front, delete a node from the rear and display.
12. Write an interactive program to implement pre order, post order and in order traversal of a binary tree using linked list.

Note: Lecturer May Change the Programs without deviating Theory Paper

SEMESTER IV

Course code: CMD25002

DSC 3D - Computer Science – IV

Operating Systems

Credits: Theory – 04 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Learn in details with examples system software
CO2. Learn the details of Operating System organization
CO3. Understand the classification and characteristics of Process Management and Scheduling mechanisms
CO4. Understand in depth Memory Management and allocation strategies
CO5. Learn in details with examples basic concepts of shell scripting
CO6. Understand in depth basic Linux environment

Unit - 1 **15 Hours**

Introduction: System Software, Resource Abstraction, OS strategies. Types of operating systems - Multiprogramming, Batch, Time Sharing, Single user and Multiuser, Process Control & Real Time Systems

Operating System Organization: Factors in operating system design, basic OS functions, implementation consideration; process modes, methods of requesting system services – system calls and system programs.

Unit - 2 **15 Hours**

Process Management: System view of the process and resources, initiating the OS, process address space, process abstraction, resource abstraction, process hierarchy, Thread model

Unit - 3 **15 Hours**

Scheduling: Scheduling Mechanisms, Strategy selection, non-pre-emptive and pre-emptive strategies.

Memory Management: Mapping addresses space to memory space. Memory allocation strategies, fixed partition, variable partition, paging, virtual memory

Unit - 4 **15 Hours**

Shell introduction and Shell Scripting: What is shell and various type of shell, Various editors present in Linux, Different modes of operation in vi editor, What is shell script, Writing and executing the shell script, Shell variable (user defined and system variables), System calls, Using system calls, Pipes and Filters, Decision making in Shell Scripts (If else, switch), Loops in shell, Functions, Utility programs (cut, paste, join, tr, uniq utilities), Pattern matching utility (grep)

Reference:

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3rd Edition, Pearson Education 2007.
3. G. Nutt, Operating Systems: A Modern Perspective, 2nd Edition Pearson Education 1997.
4. W. Stallings, Operating Systems, Internals & Design Principles, 5th Edition, Prentice Hall of India. 2008.
5. M. Milenkovic, Operating Systems- Concepts and design, Tata McGraw Hill 1992.

SEMESTER IV

Course code: DMD25102

DSC 3D - Computer Science – IV

Operating Systems Lab

Credits: Practical – 02 **60 Hours**

Software Lab based on Operating Systems

Note: Following exercises can be performed using Linux or UNIX

1. Usage of following commands: ls, pwd, tty, cat, who, who am I, rm, mkdir, rmdir, touch, cd.
2. Usage of following commands: cal, cat(append), cat(concatenate), mv, cp, man, date.
3. Usage of following commands: chmod, grep, tput (clear, highlight), bc.
4. Write a shell script to check if the number entered at the command line is prime or not.
5. Write a shell script to modify “cal” command to display calendars of the specified months.
6. Write a shell script to modify “cal” command to display calendars of the specified range of months.
7. Write a shell script to accept a login name. If not a valid login name display message – “Entered login name is invalid”.
8. Write a shell script to display date in the mm/dd/yy format.
9. Write a shell script to display on the screen sorted output of “who” command along with the total number of users.
10. Write a shell script to display the multiplication table any number,
11. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
12. Write a shell script to find the sum of digits of a given number.
13. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
14. Write a shell script to find the LCD(least common divisor) of two numbers.
15. Write a shell script to perform the tasks of basic calculator.
16. Write a shell script to find the power of a given number.
17. Write a shell script to find the factorial of a given number.
18. Write a shell script to check whether the number is Armstrong or not.
19. Write a shell script to check whether the file have all the permissions or not.
20. Program to show the pyramid of special character “*”.

SEMESTER V

Course code: DME25002

DSE6A: Elective: Computer Science - V
Programming in JAVA

Credits: Theory – 04, 60 Hours

Course Outcome:

After successful completion of the course, the student is able to

CO1. Deliberate in depth java programming fundamental
CO2. Specify in details with examples Basic java OOPs Concepts
CO3. Understand in depth OOPs Concepts
CO4. Understand in depth java Interface and packages
CO5. Deliberate the details of Exception handling in java
CO6. Deliberate the details of Multithreading & I/O operations in java
CO7. Identify the classification and characteristics of File handling in java
CO8. Learn the details of File handling in java
CO9. Learn the characteristics of Applet Programming

Unit - 1 15 Hours

Introduction to Java: Features of Java, JDK Environment, Object Oriented Programming Concept Overview of Programming, Paradigm, Classes, Abstraction, Encapsulation, Inheritance, Polymorphism, Difference between C++ and JAVA,JDK Environment
Java Programming Fundamental: Structure of java program, Data types, Declaration of Variables, scope of variables, Keywords, operators, separators, literals, Machine neutral, JVM, Command line arguments, Decision Making statements, iterative statements, jumping statement, type conversion and casting

Unit - 2 15 Hours

Classes and Objects: Defining a Class, Field Declaration, Methods Declarations, Creating an Object, Accessing class members, Assigning Object Reference Variables, Access specifies, Constructor, Type of Contractors, this keyword instance variable hiding, Garbage Collection, Finalize method, method overloading, overloading Constructor, Understanding Static, Introducing final, Recursion.
Inheritance: Definition, Types of Inheritance, Member Access, Uses of Super, Method Overriding, Dynamic Method Dispatch, Abstract class and Methods, Uses final, Object Class, Nested and Inner classes
Arrays and Strings: Arrays, Creating an array, Types of Arrays, arrays accessing methods/operations String class Methods, String Buffer methods.

Unit - 3 15 Hours

Abstract Class, Interface and Packages: Modifiers and Access Control, Abstract classes and methods, Interfaces, Packages Concept- definition, JAVA API packages, naming conventions, creating packages, Accessing Packages using packages, adding a class to a package, hiding class static import

Interface: Introduction, Defining Interfaces, implementing Interfaces, Nested Interfaces, Extending Interfaces, accessing interface variables

Exception handling: Introduction, Types of Errors, Exception, Syntax of Exception handling, try and catch statement, multiple catch Statements, Nested try Statements, throws, throwing our own exception, finally statement, java built in exception classes

Unit - 4 **15 Hours**

Multithreading: Introduction, Creating Threads, Extending the thread Classes, implementing runnable interface, Declaring the Class, implementing the run () Method, starting new thread, stopping and blocking thread, thread life cycle, thread Priority

Input and Output: Concept of Streams, Stream Classes, Byte Stream classes, Character Stream classes, reading Console input, Writing Console Output, wrapper Classes.

Applet Programming: Introduction, Types of Applet, How applets differ from applications, Applet Life cycle, Creating Applet, Applet tag

Reference Books:

1. Ivan Bayross, Web Enabled Commercial Application Development Using Html, Dhtml,javascript, Perl Cgi , BPB Publications, 2009.
2. Cay Horstmann, BIG Java, Wiley Publication , 3rd Edition., 2009
3. Herbert Schildt , Java 7, The Complete Reference, , 8th Edition, 2009.
4. E Balagurusamy , Programming with JAVA, TMH, 2007

SEMESTER V

Course code: DME25102

DSE6A: Elective: Computer Science - V
Programming in JAVA Lab

Credits: Practical – 02 **60 Hours**

Software Lab based on Java

PART A

1.Write a java program to find whether given number is positive, negative or zero

2.Write a java program to find the largest of three number using ternary operator

3.Write a java program to find out roots of the quadratic Equation

4.Write a java program to check whether given date is valid or not

5.Write a java program to implement ATM Transaction Using Switch statement

6.Write a java program to generate the following pattern

```
         A
      A  B  A
   A  B  C  B  A
A  B  C  D  C  B  A..
```

7.Write a java program to find sum of all digits of a given number until given number become a single digit

8.Write a program to create an array of 10 integers. Accept values from the user in that array. Input another number from the user and find out how many numbers are equal to the number passed, how many are greater and how many are less than the number passed.

9.Write a java program to sort the given element using selection sort

10. Write a java program to find the trace and norm of the given square matrices

PART B

1. Write a java program to Generate Employee Salary slip Using Class and Object

2. Write a java program to check whether entered character is a vowel or consonant using Constructor

3. Write a java program to Demonstrate Method Overloading

4. Write java program to generate Student marks card Using Inheritance

5. Write a java program to calculate bonus for different departments using abstract class

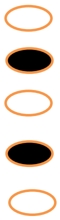
6. Write a java program to Demonstrate Method Overriding

7. Write a java program to that reads two integer numbers for the variables a and b. If any other character except number (0-9) is entered then the error is caught by NumberFormatException object. After that ex.getMessage () prints the information about the error occurring causes

8. Write a java program to Demonstrate multiple Inheritance using Interface

9. Write a java program to Demonstrate multithreading

10. Write a applet program to generate the following pattern



SEMESTER V

Course code: DME25202

DSE6B: Elective: Computer Science - V
Web Technologies

Credits: Theory – 04 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO 1. Learn the details of HTML tags
CO 2. Understand in details with examples Document object Model
CO 3. Understand the details of Basic CSS and implements
CO 4. Understand the details of Basic Concepts of Java Scripts
CO 5. Write down in details with application and Usage of Java scripts
CO 6. Deliberate in depth JQuery& Bootstrap with Examples

Unit - 1 **15 Hours**

Introduction to Web Design: Introduction to HyperText Markup Language (HTML), header, footer, formatting tags, graphical elements, inserting images, lists, hyperlinks, tables. **Frames -** Introduction, frameset. Forms- attributes of forms. Creating web pages

DOM: Basics of DOM, DOM methods, functions Forms collection, table collections Inner HTML.

Unit - 2 **15 Hours**

Cascading Style Sheets: Introduction, Understanding the Basic CSS syntax, Types of style sheets, multiple sheets, and Background properties, Text properties, Font properties, and Border properties, Margin properties padding list & table properties. DIV, SPAN,CSS Layout - The position Property, float and clear, the display-inline-block Property, Overflow

CSS Advanced - Rounded Corners, Border Images, Backgrounds, Colors, Gradients, Shadows, Text Effects, Web Fonts, 2D & 3D Transforms, Transitions, Animations

Unit - 3 **15 Hours**

JavaScript: Introduction, Java script in HTML, Java script statement, comments, Expressions, Data types, operators, Conditional statements, Loop statements, functions, Popup boxes ,Array & Boolean Objects Math & Date Objects String & Number Objects, events and event handling & form document object.

Unit - 4 **15 Hours**

JQuery: Introduction, Syntax, Selectors, Events, Effects, Hide/Show, Fade, Slide, Animate, stop (), Callback, Chaining

JQuery HTML Interface: Get, Set, Add, Remove, CSS Classes, css (), Dimensions

JQuery Traversing: Ancestors, Descendants, Siblings & Filtering

Bootstrap 4: Introduction, Grid Basic, Typography, Colors, Tables, Images, Jumbotron, Alerts, Buttons, Badges, List Groups, Cards, Dropdowns, Collapse, Navs, Navbar, Forms, Inputs, Input Groups, Custom Forms, Carousel, Modal, Tooltip, Popover, Toast, Scrollspy, Utilities, Flex, Icons, Media Objects, Filters

Reference Books:

1. M.Deitel, P.J.Deitel, A.B.Goldberg: Internet & World Wide Web How to program, 3rd Edition, Pearson Education / PHI, 2004.
2. Chris Bates: Web Programming Building Internet Applications, 3rd Edition, Wiley India, 2006.
3. 3. XueBai et al: The Web Warrior Guide to Web Programming, Thomson, 2003.
4. Learning jQuery Fourth Edition-Jonathan Chaffer.
5. Web Development with jQuery -Richard York
6. Bootstrap-Jake Spurlock
7. Mastering Bootstrap4- Benjamin Jakobus, Jason Marah

SEMESTER V

Course code: CME25302

DSE6B: Elective: Computer Science - V
Web Technologies Lab

Credits: Practical – 02 **60 Hours**

1. Program for formatting tags.
2. Creating a Webpage having Hyperlink.
3. Creating Types of Lists (Ordered, Unordered, Definition).
4. Creating a Nested List.
5. Creating a Time Table.
6. Creating a HTML document having vertical frames.
7. Creating Student Application Form.
8. Program to insert audio & video files
9. Creating Internal & External Style Sheets.
10. Program to Margin & Padding.
11. Program to create a Greeting card
12. Program to Image Transparency
13. Program to generate Fibonacci series in JavaScript.
14. Program to display Rainbow Colors in JavaScript.
15. Program to create Pop-Up Boxes.
16. Program to generate multiplication table.
17. Program to find even and odd numbers.
18. Program to add 2 numbers.
19. Program to find factorial of a numbers.
20. Program to generate 2 different patterns.
21. Program to change background color after 5 sec of page load.
22. Display reverse of a given number.
23. Display Time Using Jquery
24. Design Simple Department web Site Using Bootstrap
25. Design Web page using Bootstrap and Jquery

SEMESTER V

Course code: CMF25402

DSE6C: Elective: Computer Science - V
Mobile Applications

Credits: Theory – 04 60 Hours

Course Outcome:

After successful completion of the course, the student is able to

CO1. Deliberate the details of Concepts of Event Driven Programming
CO2. Learn in details with examples issues of Mobile applications
CO3. Specify the details of Mobile applications Development tools and Frameworks
CO4. Deliberate in details with examples common Mobile device UI's
CO5. Write down in depth Data persistence Remote data storage and communication
CO6. Learn in details with examples Code signing

Unit - 1 **15 Hours**

Event Driven Programming: UI event loop, Threading for background tasks, Outlets / actions, delegation, notification, Model View Controller (MVC) design pattern.

Mobile application issues: limited resources (memory, display, network, file system), input / output (multi-touch and gestures), sensors (camera, compass, accelerometer, GPS)

Unit - 2 **15 Hours**

Development tools: Apple IOS tool chain: Objective-C, Xcode IDE, Interface Builder, Device simulator.

Frameworks: Objective-C and Foundation Frameworks, Cocoa Touch, UI Kit, Others: Core Graphics, Core Animation, Core Location and Maps, Basic Interaction.

Unit - 3 **15 Hours**

Common UI's for mobile devices: Navigation Controllers, Tab Bars, Table Views, Modal views, UI Layout.

Data Persistence: Maintaining state between application invocations, File system, Property Lists, SQLite, Core Data.

Unit - 4 **15 Hours**

Remote Data-Storage and Communication: "Back End" / server side of application, RESTful programming, HTTP get, post, put, delete, database design, server side JavaScript / JSON.

Code signing: security, Keychain, Developers and App Store License Agreement

Reference:

1. Rajiv Ramnath, Roger Crawfis, and Paolo Sivilotti, Android SDK 3 for Dummies, Wiley, 2011.
2. Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design, and Development, Prentice Hall, 2004.
3. Brian Fling, Mobile Design and Development, O'Reilly Media, 2009. Maximiliano
4. Firtman, Programming the Mobile Web, O'Reilly Media, 2010.
5. Christian Crumlish and Erin Malone, Designing Social Interfaces, O'Reilly Media, 2009.
6. James E Smith, Ravi Nair, "Virtual Machines", Morgan Kaufmann Publishers, 2006.

SEMESTER V

Course code: CME25502

DSE6C: Elective: Computer Science - V
Mobile Applications Lab

Credits: Practical – 02 **60 Hours**

Software Lab based on Mobile Applications:

1. Installing Android Environment
2. Create Hello World Application
3. Sample Application about Android Resources
4. Sample Application about Layouts
5. Sample Application about Intents
6. Sample Application I about user interfaces
7. Sample Application about Animations
8. Make a Project based on above labs
9. Sample Application about Android Data
10. Sample Application about SQLite I
11. Sample Application about SQLite II
12. Project Presentation

SEMESTER VI

Course code: CMF25002

DSE 6A: Elective: Computer Science – VI

Database Management Systems

Credits: Theory – 04 60 Hours

Course Outcome:

After successful completion of the course, the student is able to

CO1. Understand the characteristics of DBMS with examples
CO2. Deliberate the details of types of database languages with examples
CO3. Learn the details of ER- Diagrams and Relationship
CO4. Understand in depth Basic concepts of Relational Model
CO5. Learn in details with examples MYSQL Commands
CO6. Learn in details with examples in PL-SQL

Unit - 1 15 Hours

Introduction to Database Management Systems: Definition of Data, Information, DBMS, Data base system application, Purpose of database systems, Characteristics of DB – Self describing nature, Insulation between programs, data and data Abstraction (data Independence), support of multiple views of the data, sharing of data and multiples transaction processing, Storage management, Database language – DDL, DML,DCL.

File processing system v/s DBMS, Data models, Levels of Abstraction in a DBMS, Three Schema architecture, Characteristics of database approach,, data models, DBMS architecture and data independence.

Unit - 2 15 Hours

Entity Relationship and Enhanced ER Modeling: Entity types, Entity Sets, Attributes, and Keys, Relationships, Relationship Types, Roles, and Structural Constraints, Weak Entity Types, ER Diagrams, Naming Conventions, SQL99: Schema Definition, constraints, and object modeling

Unit - 3 15 Hours

Relational Data Model: Basic concepts, Relational Constraints and Relational Database Schemas, Update Operations and Dealing with Constraint Violations, Basic Relational Algebra Operations.

Database design: ER and EER to relational mapping, functional dependencies, normal forms-first normal form, second normal forms, third normal form BCNF

Unit - 4 15 Hours

MYSQL (SQL/PL-SQL) :SQL VS. SQL * PLUS: SQL COMMANDS ANDDATA TYPES, OPERATORS AND EXPRESSIONS, INTRODUCTION TO SQL * PLUS.

MANAGING TABLES AND DATA:
• CREATING AND ALTERING TABLES (INCLUDING CONSTRAINTS)
• DATA MANIPULATION COMMAND LIKE INSERT, UPDATE, DELETE
• SELECT STATEMENT WITH WHERE, GROUP BY AND HAVING, ORDER BY, DISTINCT, SPECIAL OPERATOR LIKE IN, ANY, ALL BETWEEN, EXISTS, LIKE
• JOIN, BUILT IN FUNCTIONS OTHER DATABASE OBJECTS
• VIEW • SYNONYMS, INDEX TRANSACTION CONTROL STATEMENTS
• COMMIT, ROLLBACK, SAVEPOINT INTRODUCTION TO PL/SQL
• SQL V/S PL/SQL • PL/SQL BLOCK STRUCTURE
• LANGUAGE CONSTRUCT OF PL/SQL (VARIABLES, BASIC AND COMPOSITE DATA TYPE, CONDITIONS LOOPING ETC.)
• % TYPE AND % ROWTYPE
• USING CURSOR (IMPLICIT, EXPLICIT)

Reference:

1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.
2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.
3. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education,2013.

SEMESTER VI

Course code: CMF25102

DSE 6A: Elective: Computer Science – VI

Database Management Systems Lab

Credits: Practical – 02 60 Hours

Software Lab based on Database Management Systems

The following concepts must be introduced to the students:

Note: MS Access/MySQL may be used.

DDL Commands

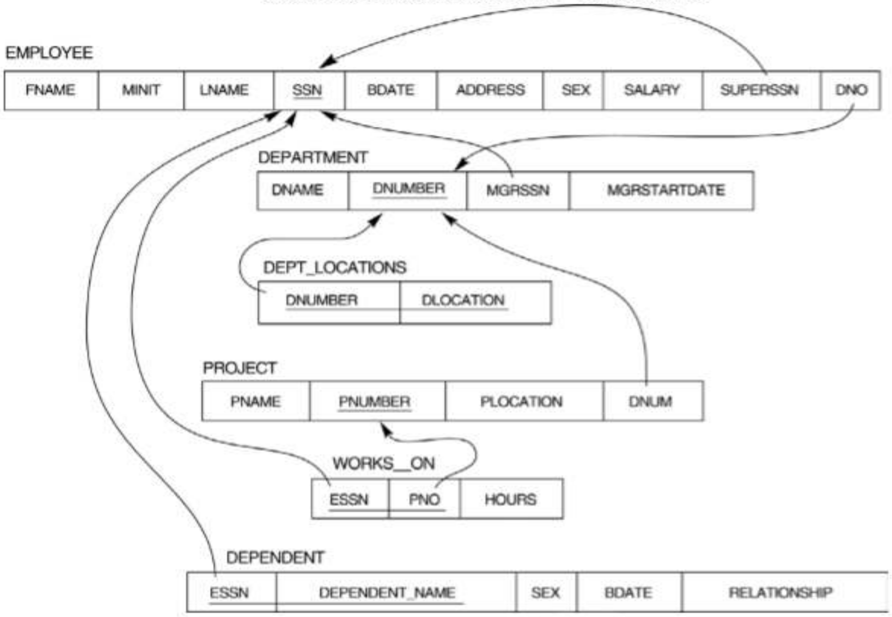
- Create table, alter table, drop table

DML Commands

- Select, update, delete, and insert statements
- Condition specification using Boolean and comparison operators (and, or, not,=,<>,>,<,>=,<=)
- Arithmetic operators and aggregate functions(Count, sum, avg, Min, Max)
- Multiple table queries (join on different and same tables) • Nested select statements

- Set manipulation using (any, in, contains, all, not in, not contains, exists, not exists, union, intersect, minus, etc.)
- Categorization using group by……..having
- Arranging using order by

Relational Database Schema - COMPANY



1. Create tables with relevant foreign key constraints
2. Populate the tables with data
3. Perform the following queries on the database:
   a. Display all the details of all employees working in the company.
   b. Displayssn, lname, fname, address of employees who work in department no 7.
   c. Retrieve the birthdate and address of the employee whose name is ‘Franklin T. Wong'
4. Retrieve the name and salary of every employee
5. Retrieve all distinct salary values
6. Retrieve all employee names whose address is in ‘Bellaire’
7. Retrieve all employees who were born during the 1950s
8. Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)
9. Retrieve the names of all employees who do not have supervisors
10. Retrieve SSN and department name for all employees
11. Retrieve the name and address of all employees who work for the 'Research' department
12. For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birthdate.
13. For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.
14. Retrieve all combinations of Employee Name and Department Name

15. Make a list of all project numbers for projects that involve an employee whose last name is 'Narayan' either as a worker or as a manager of the department that controls the project.
16. Increase the salary of all employees working on the 'ProductX' project by 15%. Retrieve employee name and increased salary of these employees.
17. Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employee first name.
18. Select the names of employees whose salary does not match with salary of any employee in department.
19. Retrieve the name of each employee who has a dependent with the same first name and same sex as the employee.
20. Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford.
21. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Display with proper headings.
22. Find the sum of the salaries and number of employees of all employees of the 'Marketing' department, as well as the maximum salary, the minimum salary, and the average salary in this department
23. Select the names of employees whose salary is greater than the average salary of all employees in department 10.
24. For each department, retrieve the department number, the number of employees in the department, and their average salary.
25. For each project, retrieve the project number, the project name, and the number of employees who work on that project.
26. Change the location and controlling department number for all projects having more than 5 employees to 'Bellaire' and 6 respectively.
27. For each department having more than 10 employees, retrieve the department no, no of employees drawing more than 40,000 as salary.
28. Insert a record in Project table which violates referential integrity constraint with respect to Department number. Now remove the violation by making necessary insertion in the Department table.
29. Delete all dependents of employee whose ssn is '123456789'.
30. Delete an employee from Employee table with ssn = '12345'(make sure that this employee has some dependents, is working on some project, is a manager of some department and is supervising some employees). Check and display the cascading effect on Dependent and Works on table. In Department table MGRSSN should be set to default value and in Employee table SUPERSSN should be set to NULL
31. Perform a query using alter command to drop/add field and a constraint in Employee table.

SEMESTER VI

Course code: CMF25202

DSE6B: Elective: Computer Science - VI
Computer Networks

Credits: Theory – 04 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Understand the Elements of Data Communications and network Systems
CO2. Learn in depth Transmission Media
CO3. Understand in details with examples Network Models
CO4. Understanding the various classifications and characteristics of Protocols
CO5. Learn in depth Error Detection and Corrections Algorithms
CO6. Learn in detail of Network Security

Unit - 1 15 Hours

Basic concepts: Components of data communication, standards and organizations, Network Classification, Network Topologies ; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite.

Unit - 2 15 Hours

Physical Layer: Cabling, Network Interface Card, Transmission Media Devices- Repeater, Hub, Bridge, Switch, Router, Gateway.

Data Link Layer: Framing techniques; Error Control; Flow Control Protocols; shared media protocols - CSMA/CD and CSMA/CA.

Unit - 3 15 Hours

Network Layer: Virtual Circuits and Datagram approach, IP addressing methods – Sub netting; Routing Algorithms (adaptive and non-adaptive)

Transport Layer: Transport services, Transport Layer protocol of TCP and UDP

Unit - 4 15 Hours

Application Layer: Application layer protocols and services – Domain name system, HTTP, WWW, telnet, FTP, SMTP

Network Security: Common Terms, Firewalls, Virtual Private Networks.

Reference:

1. B.A. Forouzan: Data Communication and Networking, 4th Edition, Tata McGraw Hill, 2007.
2. D.E. Comer, Internetworking with TCP/IP, Vol. I, Prentice Hall of India, 1998.
3. W. Stalling, Data & Computer Communication, 8th edition, Prentice Hall of India, 2006.
4. D. Bertsekas, R. Gallager, Data Networks, 2nd edition, Prentice Hall of India, 1992.

SEMESTER VI

Course code: CMF25302

DSE6B: Elective: Computer Science - VI
Computer Networks

Credits: Practicals – 02 **60 Hours**

Software Lab based on Computer Networks:

Implement the concepts of Computer Networks such as:

1. Simulate Checksum Algorithm.
2. Simulate CRC Algorithm
3. Simulate Stop &amp; Wait Protocol.
4. Simulate Go-Back-N Protocol.
5. Simulate Selective Repeat Protocol.

and so on….

SEMESTER VI

Course code: CMF25402

DSE 6C: Elective: Computer Science - VI
Software Engineering

Credits: Theory – 04 60 Hours

Course Outcome:

After successful completion of the course, the student is able to

CO1. Understand in details with examples Concepts of Software process
CO2. Specify the details of Software requirements and analysis
CO3. Learn in depth Design concepts and principles of software engineering
CO4. Understand in depth software Configuration Management and Project Management
CO5. Learn in details with examples Software Testing
CO6. Specify in depth trends in software engineering

Unit - 1 15 Hours

Software Process: Introduction ,S/W Engineering Paradigm , life cycle models (water fall, incremental, spiral, evolutionary, prototyping, object oriented) , System engineering, computer based system, verification, validation, life cycle process, development process, system engineering hierarchy.

Software requirements: Functional and non-functional, user, system, requirement engineering process, feasibility studies, requirements, elicitation, validation and management, software prototyping, prototyping in the software process, rapid prototyping techniques, user interface prototyping, S/W document.

Unit - 2 15 Hours

Analysis and modeling, data, functional and behavioral models, structured analysis and data dictionary.

Design Concepts and Principles: Design process and concepts, modular design, design heuristic, design model and document, Architectural design, software architecture, data design, architectural design, transform and transaction mapping, user interface design, user interface design principles. Real time systems, Real time software design, system design, real time executives, data acquisition system, monitoring and control system.

Unit - 3 15 Hours

Software Configuration Management: The SCM process, Version control, Change control, Configuration audit, SCM standards.

Software Project Management: Measures and measurements, S/W complexity and science measure, size measure, data and logic structure measure, information flow measure. Estimations for Software Projects, Empirical Estimation Models, Project Scheduling.

Unit - 4 **15 Hours**

Testing: Taxonomy of software testing, levels, test activities, types of s/w test, black box testing, and testing boundary conditions, structural testing, test coverage criteria based on data flow, mechanisms, regression testing, testing in the large. S/W testing strategies, strategic approach and issues, unit testing, integration testing, validation testing, system testing and debugging.
Trends in Software Engineering: Reverse Engineering and Re-engineering – wrappers – Case Study of CASE tools.

Reference:

1. Roger S. Pressman, Software engineering- A practitioner's Approach, McGraw-Hill
2. Ian Sommerville, Software engineering, Pearson education Asia, 6th edition, 2000.
3. PankajJalote- An Integrated Approach to Software Engineering, Springer Verlag, 1997.
4. James F Peters and WitoldPedryez, "Software Engineering – An Engineering Approach", John Wiley and Sons, New Delhi, 2000.
5. Ali Behforooz and Frederick J Hudson, "Software Engineering Fundamentals", Oxfor University Press, New Delhi, 1996.
6. Pfleeger, "Software Engineering", Pearson Education India, New Delhi, 1999.
7. Carlo Ghezzi, Mehdi Jazayari and Dino Mandrioli, "Fundamentals of Software Engineering", Prentice Hall of India, New Delhi, 1991.

SEMESTER VI

Course code: CME25502

DSE 6C: Elective: Computer Science - VI
Software Engineering

Credits: Practical – 02 60 Hours

Lab based on Software Engineering

1.Practical Title
- Problem Statement,
- Process Model

2.Requirement Analysis
- Creating a Data Flow
- Data Dictionary,
- Use Cases

3.Project Management
- Computing FP
- Effort
- Schedule, Risk Table, Timeline chart

4.Design Engineering

- Architectural Design
- Data Design, Component Level Design

5.Testing

- Basis Path Testing

Sample Projects

- DTC Route Information: Online information about the bus routes and their frequency and fares
- Car Pooling: To maintain a web based intranet application that enables the corporate employees within an organization to avail the facility of carpooling effectively.
- Patient Appointment and Prescription Management System
- Organized Retail Shopping Management Software
- Parking Allocation System
- Wholesale Management System

SEMESTER III / IV

Course code: DMC##### / DLD#####

SEC 1A: Elective: Computer Application (Practical) Office Automation

Credits: Practical – 02 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Understand the details of fundamentals Of Computer
CO2. Learn in depth Hardware and Software
CO3. Learn the details of Computer Peripherals
CO4. Understand the details of Programming Languages
CO5. Deliberate in details with examples office automation Tools
CO6. Deliberate in depth Operating System and the User Interface
CO7. Understand in details of Internet and its usages

Practical List for WORD:

1. Create a **telephone directory**.
   - The heading should be 16-point Arial Font in bold
   - The rest of the document should use 10-point font size
   - Other headings should use 10-point Courier New Font.
   - The footer should show the page number as well as the date last updated.
2. Design a time-table form for your college.
   - The first line should mention the name of the college in 16-point Arial Font and should be bold.
   - The second line should give the course name/teacher's name and the department in 14-point Arial.
   - Leave a gap of 12-points.
   - The rest of the document should use 10-point Times New Roman font.
   - The footer should contain your specifications as the designer and date of creation.
3. Create the following document: A newsletter with a headline and 2 columns in portrait orientation, including at least one image surrounded by text.

Practical List for EXCEL:

1. Create a student worksheet containing roll numbers, names and total marks. Open a document in Word and insert the excel worksheet using:-

   i) Copy/Paste
   ii) Embedding
   iii) Linking
2. The term wise marks for APS class of 20 students are stored in 3 separate sheets named term1, term2 and term3. Create 4th worksheet that contains student names and their total

and average marks for the entire year. Give proper headings using headers. Make the column headings bold and italic. The 4th worksheet should contain college name as the first line. Make it bold, italic and center it.

3. Consider the following employee worksheet:-

Full Name (First Last)	Grade 1/2/3	Basic Salary	HRA	PF	Gross	Net	(VA)VehicleAllowance

HRA is calculated as follows:

Grade	HRA %(of Basic)
1	40%
2	35%
3	30%

Gross = Basic : HRA : VA
Net = Gross –PF
PF is 8% for all Grades
VA is 15000, 10000 and 7000 for Grades 1, 2 and 3.

i) Find max, min and average salary of employees in respective Grade
ii) Count no. of people where VA>HRA
iii) Find out most frequently occurring grade.
iv) Extract records where employee name starts with "A" has HRA>10000
v) Print Grade wise report of all employees with subtotals of net salary and also grand totals. Use subtotal command.
vi) Extract records where Grade is 1 or 2 and salary is between 10000 and 20000 both inclusive.

4. In a meeting of a marketing department of an organization it has been decided that price of selling an item is fixed at Rs40. It was resolved to increases the sell of more of more items and getting the profit of Rs40,000/.Use Goal Seek of find out how many items you will have to sell to meet your profit figure.

5. Consider the following worksheet for APS 1st year students:-

S.No.	Name	PH	CH	BY	MT	CS	Total Marks	%	Grade
1									
2									

Grade is calculated as follows:-

If % >=90	Grade A
If % >=80 &<90	Grade B
If % >=70 &<80	Grade C
If % >=60 &<70	Grade D

Otherwise students will be declared fail.

i) Calculate Grade using if function

ii) Sort the data according to total marks
iii) Apply filter to display the marks of the students having more than 65% marks.
iv) Draw a pie chart showing % marks scored in each subject by the topper of the class.
v) Draw the doughnut chart of the data as in (iv)
vi) Enter the S.No. of a student and find out the Grade of the student using VLOOKUP.
vii) Extract all records where name
    a) Begins with "A"
    b) Contains "A"
    c) Ends with "A"

Practical List for Power Point:

a. Create five Power point slides. Each slide should support different format. In these slides

Explain areas of applications of IT. Make slide transition time as 10 seconds.

b. Create five Power Point slides to give advantages/disadvantages of computer, application

of computers and logical structure of computer.

c. Create five Power Point slides detailing the process of internal assessment. It should be a self-running demo.

SEMESTER III / IV

Course code: DMC##### / DLD#####

SEC 1B: Elective: Computer Application (Practical)
Elective: XML Programming

Credits: Practical – 02 60 Hours

Course Outcome:

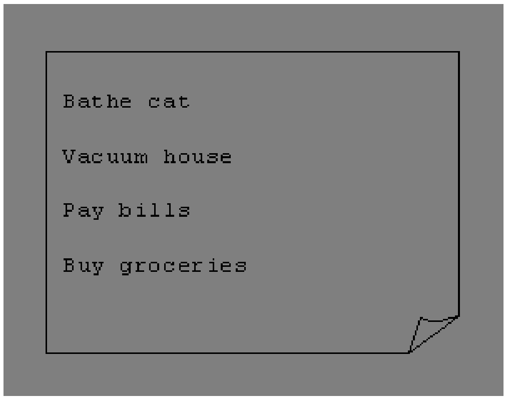
After successful completion of the course, the student is able to

CO1. Understand the details of Elements of XML Programming
CO2. Write down in details with examples DTD
CO3. Deliberate in details with examples XML Schemas
CO4. Deliberate the characteristics of CSS
CO5. Learn in details with application CSS
CO6. Deliberate in details with examples XSL
CO7. Learn the details of XML Security
CO8. Learn in details with examples XML and JAVA
CO9. Learn in details with examples XML and ASP.Net

Software Lab Based on XML:

Exercise #1 – Information Structure

In this exercise, student will practice identifying the structure of an information object.
For the sample document provided below:
Label the information structures you see, including containing structures.
12. Draw a tree representation of the structure.



Bathe cat

Vacuum house

Pay bills

Buy groceries

Exercise 2# Deconstructing an XML Document

In this exercise, student will practice identifying the explicit structure within an XML document. In a sense, this is the reverse of what you did in Exercise #1. For the sample XML markup below, create a document-like representation (or a simple drawing) for the content contained within the XML tags:

```
<book>
<coverInfo>
```

```
<title>The XML Handbook</title>
<author>Charles F. Goldfarb</author>
<author>Paul Prescod</author>
<edition>Second</edition>
<description>The definitive XML resource: applications, products, and technologies. Revised
and expanded—over 600 new pages.
</description>
</coverInfo>
</book>
```

Exercise #3 – Creating XML Markup
In this exercise, create some XML markup based on the tree representation from Exercise #1 above, and the content from the original sample document.

Exercise #4 – Well-Formedness
This exercise checks your understanding of the constraints for well-formedness. Are the following document instances well-formed? Explain any NO answers.

```
<list><title>The first list</title><item>An item</list>
<item>An item</item><item>Another item</item>
<para>Bathing a cat is a <emph>relatively</emph> easy task as long as the cat is
willing.</para>
<bibl><title>How to Bathe a Cat<author></title>Merlin Bauer<author></bibl>
```

Exercise #5 - Well Formedness
This exercise is a bit more challenging than the previous example. Here is a fragment of an XML document instance. Identify all the places where it fails to match the constraints for well-formedness.

```
<PROCEDURE><TITLE>How to Bathe a Cat</TITLE>
<OVERVIEW>This procedure tells you how to bathe a cat. <WARNING></OVERVIEW>Cats
don't like to take baths. You could get hurt doing this. Be sure to obtain all the required
protective gear before you start. </WARNING><EQUIPEMENT><ITEM>Hockey Mask
<ITEM>Padded Full-body Kevlar Armor </ITEM><ITEM>Tub full of warm
water</ITEM><ITEM>Towels </ITEM> <ITEM>First Aid kit</ITEM><ITEM>Cat
Shampoo</ITEM><EQUIPMENT> <INSTRUCTIONS><STEP> Locate the cat, who by now is
hiding under the bed.</STEP> <STEP>Place the cat in the tub of water.</STEP><ITEM>Using
the First Aid kit, repair the damage to your head and arms.</STEP><STEP>Place the cat back in
the tub and hold it down.</STEP><STEP>Wash it really fast, then make an effort to dry it with
the towels.</STEP><STEP>Decide not to do this again. </STEP></INSTRUCTIONS>
```

Note: Cover more exercises based on XML Programming theory concepts.

SEMESTER III/IV

Course code: DMC##### / DLD#####

SEC 1C: Elective: Computer Application (Practical)
Cloud Computing Lab

Credits: Practical – 02 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Understand the details of Cloud Services and File System
CO2. Understand the details of Virtualization in cloud
CO3. Learn the classification and characteristics of Security challenges in Cloud Computing
CO4. Understand the details of Security challenges in Cloud Computing
CO5. Understand the Common standards of Cloud Computing
CO6. Deliberate in details with examples Various Application of Cloud Computing

Software Lab based on Cloud Computing:

1. Create virtual machines that access different programs on same platform.
2. Create virtual machines that access different programs on different platforms.
3. Exploring Google cloud for the following
   a) Storage
   b) Sharing of data
   c) Manage your calendar, to-do lists,
   d) A document editing tool
4. Exploring Open source cloud (Any two)

Reference:

1. Bloor R., Kanfman M., Halper F. Judith Hurwitz "Cloud Computing " Wiley India Edition,2010
2. John Rittinghouse& James Ransome, "Cloud Computing Implementation Management and Strategy", CRC Press, 2010
3. Antohy T Velte ,Cloud Computing : "A Practical Approach", McGraw Hill,2009
4. Michael Miller, Cloud Computing: "Web-Based Applications That Change the Way You Work and Collaborate Online", Que Publishing, August 2008.
5. James E Smith, Ravi Nair, "Virtual Machines", Morgan Kaufmann Publishers, 2006.

Online Reading/Supporting Material

6. Haley Beard, "Cloud Computing Best Practices for Managing and Measuring Processes for On-demand Computing", Applications and Data Centers in the Cloud with SLAs, Emereo Pty Limited, July 2008
7. webpages.iust.ac.ir/hsalimi/.../89.../Cloud%20Common%20standards.pptop ennebula.org,
8. www.cloudbus.org/cloudsim/, http://www.eucalyptus.com/
9. hadoop.apache.org
10. http://hadoop.apache.org/docs/stable/hdfs_design.html
11. http://static.googleusercontent.com/external_content/untrusted_dlcp/research.google.com/en/ /archive/mapreduce-osdi04.pdf

SEMESTER VI

Course code: CMF25602

SEC4A: Elective: Computer Science - VII
Mini Project

Credits: Practical – 02 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Identify in details with examples Problem identification

CO2. Write down in depth System Analysis

CO3. Understand and Develop SRS for selected System Problem

CO4. Understand and Develop System Design for selected System Problem

CO5. Learn in details and Develop a Code and Test the System

CO6. Understand the details of Presentation and Demo of Project Work

- ✓ This option is to be offered only in 6th Semester.
- ✓ The students will be allowed to work on any project based on the concepts studied in
- ✓ Core/elective or skill based elective courses.
- ✓ The group size should be maximum of FOUR (04) students.
- ✓ Each group will be assigned a teacher as a supervisor who will handle both their theory as Well as lab classes.
- ✓ A maximum of Four (04) projects would be assigned to one teacher.

SEMESTER VI

Course code: CMF25702

SEC4B: Elective: Computer Science - VII
PHP Programming

Credits: Practical – 02 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO1. Learn in depth Elements of PHP
CO2. Learn in depth Interaction Methods Between HTML and PHP
CO3. Understand in depth PHP function
CO4. Understand in depth String Manipulation
CO5. Learn the characteristics of Regular Expression
CO6. Learn the details of Developing PHP Web Application

PHP Programming Lab

Software Lab Based on PHP:

1. Create a PHP page using functions for comparing three integers and print the largest number.
2. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.
3. WAP to check whether the given number is prime or not.
4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
5. Write a PHP function that checks if a string is all lower case.
6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
7. WAP to sort an array.
8. Write a PHP script that removes the whitespaces from a string.
   Sample string: 'The quick " " brown fox'
   Expected Output: Thequick""brownfox
9. Write a PHP script that finds out the sum of first n odd numbers.
10. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise error message should be displayed.
11. Write a PHP script that checks if a string contains another string.
12. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.
13. Create a script to construct the following pattern, using nested for loop.
    *

```
* *
* * *
* * * *
* * * * *
```

14. Write a simple PHP program to check that emails are valid.
15. WAP to print first n even numbers.
16. $color = array('white', 'green', 'red")
    Write a PHP script which will display the colors in the following way:
    Output:
    White, green, red,
    • Green
    • Red
    • White
17. Using switch case and dropdown list display a “Hello” message depending on the language selected in drop down list.
18. Write a PHP program to print Fibonacci series using recursion.
19. Write a PHP script to replace the first 'the' of the following string with 'That'.
    Sample: 'the quick brown fox jumps over the lazy dog.'
    Expected Result: That quick brown fox jumps over the lazy dog.

Reference:

1. Rajiv Ramnath, Roger Crawfis, and Paolo Sivilotti, Android SDK 3 for Dummies, Wiley, 2011.
2. Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design, and Development, Prentice Hall, 2004.
3. Brian Fling, Mobile Design and Development, O'Reilly Media, 2009. Maximiliano
4. Firtman, Programming the Mobile Web, O'Reilly Media, 2010.
5. Christian Crumlish and Erin Malone, Designing Social Interfaces, O'Reilly Media, 2009.

SEMESTER VI

Course code: CMF25802

SEC4C: Elective: Computer Science - VII
R Programming

Credits: Credits: Practical – 02 **60 Hours**

Course Outcome:

After successful completion of the course, the student is able to

CO2. Learn in depth Elements of R Programming
CO3. Understand the details of R programming Packages
CO4. Understand in details with examples of Data Management
CO5. Understand in depth Graphical Procedures
CO6. Deliberate in depth Portability Distribution
CO7. Software Lab Based on R Programming:

1. Write a program that prints 'Hello World' to the screen.
2. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n
3. Write a program that prints a multiplication table for numbers up to 12.
4. Write a function that returns the largest element in a list.
5. Write a function that computes the running total of a list.
6. Write a function that tests whether a string is a palindrome.
7. Implement the following sorting algorithms: Selection sort, Insertion sort, Bubble Sort
8. Implement linear search.
9. Implement binary search.
10. Implement matrices addition, subtraction and Multiplication

Question Paper Pattern

Theory (4 Credits)

Time: 3 Hours   Max. Marks: 70

Part – A

I. Answer any Eleven Question out of given Twelve Questions.   11 X 2 = 22

1.
2.
3.
4.
5.
6.
7.
8.
9.
10.
11.
12.

Note: Three Questions each from the Units 1, 2, 3, and 4

Part - B

II. Answer any Two Sub Questions from each main Question.

13. A)   2 X 6 = 12
B)
C)

14. A)   2 X 6 = 12
B)
C)

15. A)   2 X 6 = 12
B)
C)

16. A)   2 X 6 = 12
B)
C)

Note: Each Main questions contains 3 sub questions carries 06 Marks (may have internal Split-ups) and from Units 1, 2, 3 and 4 respectively

Question Paper Pattern

Theory (2 Credits)

Time: 2 Hours   Max. Marks: 35

Part – A

III. Answer any Five Questions.   **05 X 02 = 10**

1.
2.
3.
4.
5.
6.

Part - B

IV. Answer any Five Questions out of given Six Questions.   **5 X 5 = 25**

7.
8.
9.
10.
11.

Note: Each Main questions contains 3 sub questions carries 06 Marks (may have internal Split-ups) and from Units 1 and 2 respectively

Question Paper Pattern

Practical / Project (3 or 4 Credits)

Time: 3 Hours	Max. Marks: 70

Practicals

1. Two Experiments/ Programs 20 Marks Each	2 x 20 = 40 Marks
a. Write-ups 10 Marks	
b. Conducting & Results 10 Marks	
2. Viva – Voice	10 Marks
3. Record	10 Marks

Project

1. Project Presentation	15 Marks
2. Project Demo	20 Marks
3. Viva – Voice	10 Marks
4. Record	25 Marks

Question Paper Pattern

Practical (2 Credit)

Time: 3 Hours	**Max. Marks: 35**

1. a. Experiment/ Programs /Project demo b. Conducting & Results	20 Marks
2. Viva – Voice	10 Marks
3. Record	05 Marks