Details of courses under BCA Honours

BCA (Honours) (BACHELORS IN COMPUTER APPLICATION)

	Course	***Credits**	
		Theory+ Practical	**Theory + Tutorial**
I.	Core Course (14 Papers)	14X4= 56	14X5=70
	Core Course Practical / Tutorial* (14 Papers)	14X2=28	14X1=14
II.	Elective Course (8 Papers)		
	A.1. Discipline Specific Elective (4 Papers)	4X4=16	4X5=20
	A.2. Discipline Specific Elective Practical/ Tutorial* (4 Papers)	4 X 2=8	4X1=4
	B.1. Generic Elective/ Interdisciplinary (4 Papers)	4X4=16	4X5=20
	B.2. Generic Elective Practical/ Tutorial* (4 Papers)	4 X 2=8	4X1=4
	Optional Dissertation or project work in place of one Discipline Specific Elective paper (6 credits) in 6th Semester.		
III.	Ability Enhancement Courses		
	Ability Enhancement Compulsory (2 Papers of 2 credit each) Environmental Science /English/ MIL Communication	2 X 2=4	2 X 2=4
	Ability Enhancement Elective (Skill Based) (Minimum 2) (2 Papers of 2 credits each)	2 X 2=4	2 X 2=4
	Total credit	**140**	**140**

Institute should evolve a system/policy about ECA/ General Interest/ Hobby/ Sports/NCC/NSS/related courses on its own.

* Wherever there is a practical there will be no tutorial and vice-versa

SEMESTER	COURSE OPTED	COURSE NAME	Credits
I	Core course -I	Programming Fundamentals using C++	4
	Core Course-I Practical/Tutorial	Programming Fundamentals using C++ Lab	2
	Core course-II	Computer System Architecture	4
	Core Course-II Practical/Tutorial	Linux / Unix Programming lab	2
	GenericElective-1	GE-1	4/5
	GenericElective-1Practical/Tutorial		2/1
II	Core course-III	Programming in JAVA	4
	Core Course-III Practical/Tutorial	Programming in JAVA Lab	2
	Core course-IV	Discrete Structures	5
	Core Course-IV Practical/Tutorial	Discrete Structures Tutorial	1
	Generic Elective-2	GE-2	4/5
	Generic Elective-2 Practical/Tutorial		2/1
III	Core course-V	Data Structures	4
	Core Course-V Practical/Tutorial	Data Structures Lab	2
	Core course-VI	Operating Systems	4
	Core Course-VI Practical/Tutorial	Operating Systems Lab	2
	Core course-VII	Computer Networks	4
	Core Course-VII Practical/Tutorial	Computer Networks Lab	2
	Skill Enhancement Course-1	SEC-1	2
	Generic Elective-3	GE-3	4/5
	GenericElective-3Practical/Tutorial		2/1
IV	Core course-VIII	Design and Analysis of Algorithms	4
	Course-VIII Practical/Tutorial	Design and Analysis of Algorithms Lab	2
	Core course-IX	Software Engineering	4
	Core Course-IX Practical/Tutorial	Software Engineering Lab	2
	Core course-X	Database Management Systems	4
	Core Course-X Practical/Tutorial	Database Management Systems Lab	2
	SkillEnhancementCourse-2	SEC-2	2
	GenericElective-4	GE-4	4/5
	GenericElective-4Practical		2/1
V	Core course-XI	Internet Technologies	4
	Core Course-XI Practical/Tutorial	Internet Technologies Lab	2
	Core course-XII	Theory of Computation	5
	Core Course-XII Practical/Tutorial	Theory of Computation Tutorial	1
	DisciplineSpecificElective-1	DSE-1	4
	DisciplineSpecificElective-1 Practical/Tutorial	DSE-1Lab	2
	DisciplineSpecificElective-2	DSE-2	4
	DisciplineSpecificElective-2 Practical/Tutorial	DSE-2Lab	2
VI	Core course-XIII	Artificial Intelligence	4
	Core Course-XIII Practical/Tutorial	Artificial Intelligence Lab	2
	Core course-XIV	Computer Graphics	4
	Core Course-XIV Practical/Tutorial	Computer Graphics Lab	2
	Discipline Specific Elective-3	DSE-3	4
	Discipline Specific Elective-3 Practical/Tutorial	DSE-3Lab	2
	Discipline Specific Project	DSE-4	4
	Discipline Specific Project	DSE-4 Lab	2
Total Credits			140

Core Papers(C): (Credit: 06 each) (1 period / week for tutorials or 4 periods / week of practical)

1. Programming Fundamentals using C/C++ (4 + 4 Lab)
2. Computer System Architecture (4 + 4 Lab)
3. Programming in JAVA (4 + 4 Lab)
4. Discrete Structures (5 + 1 Tutorial)
5. Data Structures (4 + 4 Lab)
6. Operating Systems (4 + 4 Lab)
7. Computer Networks (4 + 4 Lab)
8. Design and Analysis of Algorithms (4 + 4 Lab)
9. Software Engineering (4 + 4 Lab)
10. Database Management Systems (4 + 4 Lab)
11. Internet Technologies (4 + 4 Lab)
12. Theory of Computation (5 + 1 Tutorial)
13. Artificial Intelligence (4 + 4 Lab)
14. Computer Graphics (4 + 4 Lab)

Discipline Specific Elective Papers: (Credit: 06 each) – DSE- 1, DSE- 2, DSE- 3, DSE- 4.

DSE – 1 (any one)
a) Cloud Computing (4) + Lab (4)
b) Microprocessor (4) + Lab (4)
c) Systems Programming (4) + Lab (4)

DSE – 2 (any one)
a) Information Security (4) + Lab (4)
b) Network Programming (4) + Lab (4)
c) Operational Research (4) + Lab (4)

DSE – 3 (any one)
a) Data Mining (4) + Lab (4)
b) Machine Learning (4) + Lab (4)
c) Introduction to Data Science (4) + Lab (4)

DSE – 4 (any one)
a) Computational Linguistics (4) + Lab (4)
b) Digital Image Processing (4) + Lab (4)
c) Dissertation / Project work (4) + Lab (4)

Other Discipline (Four papers of any one discipline) – GE 1 to GE 4

1. Mathematics
6. Commerce
7. Economics

Any one discipline of importance

Skill Enhancement Courses (Credit: 02 each): SEC – 1, SEC – 2 (any two from the list)

1. Android Programming (1) + Lab (2)
2. Programming in Python (1) + Lab (2)
3. PHP Programming (1) + Lab (2)
4. Software Testing (1) + Lab (2)

CORE COURSES (BCA HONOURS)

1. (C-I): Programming Fundamentals using C/C++ Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction to C and C++ (3Lectures)

History of C and C++, Overview of Procedural Programming and Object-Orientation Programming, Using main() function, Compiling and Executing Simple Programs in C++.

2. Data Types, Variables, Constants, Operators and Basic I/O (5Lectures)

Declaring, Defining and Initializing Variables, Scope of Variables, Using Named Constants, Keywords, Data Types, Casting of Data Types, Operators (Arithmetic, Logical and Bitwise), Using Comments in programs, Character I/O (getc, getchar, putc, putcharetc), Formatted and Console I/O (printf(), scanf(), cin, cout), Using Basic Header Files (stdio.h, iostream.h, conio.hetc).

3. Expressions, Conditional Statements and Iterative Statements (5Lectures)

Simple Expressions in C++ (including Unary Operator Expressions, Binary Operator Expressions), Understanding Operators Precedence in Expressions, Conditional Statements (if construct, switch-case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

4. Functions and Arrays (10 Lectures)

Utility of functions, Call by Value, Call by Reference, Functions returning value, Void functions, Inline Functions, Return data type of functions, Functions parameters, Differentiating between Declaration and Definition of Functions, Command Line Arguments/Parameters in Functions, Functions with variable number of Arguments. Creating and Using One Dimensional Arrays (Declaring and Defining an Array, Initializing an Array, Accessing individual elements in an Array, Manipulating array elements using loops), Use Various types of arrays (integer, float and character arrays / Strings) Two- dimensional Arrays (Declaring, Defining and Initializing Two Dimensional Array, Working with Rows and Columns), Introduction to Multi-dimensional arrays

5. Derived Data Types (Structures and Unions) (3Lectures)

Understanding utility of structures and unions, Declaring, initializing and using simple structures and unions, Manipulating individual members of structures and unions, Array of Structures, Individual data members as structures, Passing and returning structures from functions, Structure with union as members, Union with structures as members.

6. Pointers and References in C++ (7Lectures)

Understanding a Pointer Variable, Simple use of Pointers (Declaring and Dereferencing Pointers to simple variables), Pointers to Pointers, Pointers to structures, Problems with Pointers, Passing pointers as function arguments, Returning a pointer from a function, using arrays as pointers, Passing arrays to functions. Pointers vs. References, Declaring and initializing references, using references as function arguments and function return values

7. Memory Allocation in C++ (3Lectures)

Differentiating between static and dynamic memory allocation, use of malloc, calloc and free functions, use of new and delete operators, storage of variables in static and dynamic memory allocation

8. File I/O, Preprocessor Directives (4 Lectures)

Opening and closing a file (use of fstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(), get(), read() and write() functions, Random access in files, Understanding the Preprocessor Directives (#include, #define, #error, #if, #else, #elif, #endif, #ifdef, #ifndef and #undef), Macros

9. Using Classes in C++ (7Lectures)

Principles of Object-Oriented Programming, Defining & Using Classes, Class Constructors, Constructor Overloading, Function overloading in classes, Class Variables &Functions, Objects as parameters, Specifying the Protected and Private Access, Copy Constructors, Overview of Template classes and their use.

10. Overview of Function Overloading and Operator Overloading (5Lectures)

Need of Overloading functions and operators, Overloading functions by number and type of arguments, Looking at an operator as a function call, Overloading Operators (including assignment operators, unary operators)

11. Inheritance, Polymorphism and Exception Handling (8Lectures)

Introduction to Inheritance (Multi-Level Inheritance, Multiple Inheritance), Polymorphism (Virtual Functions, Pure Virtual Functions), Basics Exceptional Handling (using catch and throw, multiple catch statements), Catching all exceptions, Restricting exceptions, Rethrowing exceptions.

Reference Books

1. HerbtzSchildt, "C++: The Complete Reference", Fourth Edition, McGraw Hill.2003
2. E Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill Education, 2008.
3. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice Hall, 2011.

(C-I): LAB

Programming Fundamentals using C/C++ Lab Practical: 60 Lectures

1. WAP to reverse a number.
2. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……
3. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
4. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
5. WAP to print a triangle of stars as follows (take number of lines from user):

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

6. Write a program that swaps two numbers using pointers.
7. Write a program in which a function is passed address of two variables and then alter its contents.

8. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
9. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
10. WAP to display Fibonacci series
(i) using recursion,
(ii) using iteration
11. Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation):
a) Sum
) Difference
c) Product
d) Transpose
12. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).
13. Copy the contents of one text file to another file, after removing all whitespaces.
14. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
15. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

2. (C-II): Computer System Architecture Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Digital Logic Circuits: (8 lectures)
Digital computers – Logic gates – Boolean algebra – Map simplification – Combinational circuits – Flip-flops – Digital Components: Integrated circuits – Decoders – Multiplexers – Registers.

2. Data Representation: (10 lectures)
Data types – Complements – Fixed point representation – Floating point representation. Register Transfer and Microoperations: Register transfer language– Register transfer –Bus and memory transfers – Arithmetic microoperations – Logic Microoperations – Shift Microoperations – Arithmetic logic shift unit.

3. Central Processing Unit: (12 lectures)
Introduction, General Register Organization, Stack Organization, Instruction Format, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer.

4. Computer Arithmetic: (12 lectures)
Introduction, Addition and Subtraction, Multiplication Algorithms,
Division Algorithm, Floating-Point Arithmetic Operation, Decimal Arithmetic Unit.

5. Input-Output Organization: (8 lectures)

Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Direct Memory Access, Input-Output Processor.

. Memory Organization: **(10 lectures)**

Memory Hierarchy, Associative Memory, Cache Memory, Virtual Memory.

Text Book:

1.M.Morris Mano-Computer System Architecture, 3rdEdition, Pearson Education, New Delhi, 2006.

Recommended Books:

Reference Books:

1. W. Stallings- Computer Organization & Architecture, 7thEdition, Pearson Education, New Delhi, 2006

2.N. Carter- Computer Architecture, Schaums Outline Series, TMH, New Delhi, 2006

(C-II): Linux/Unix Lab (Practical): 60 Lectures

Introduction

What is linux/unix Operating systems

a) Difference between linux/unix and other operating systems
b) Features and Architecture
c) Various Distributions available in the market
d) Installation, Booting and shutdown process
e) System processes (an overview)
f) External and internal commands
g) Creation of partitions in OS
h) Processes and its creation phases – Fork, Exec, wait

User Management and the File System

a) Types of Users, Creating users, Granting rights
b) User management commands
c) File quota and various file systems available
d) File System Management and Layout, File permissions
e) Login process, Managing Disk Quotas
f) Links (hard links, symbolic links)

Shell introduction and Shell scripting

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

Text Books:

1. Sumitabha, Das, Unix Concepts And Applications, Tata McGraw-Hill Education, 2006

Reference books:

1. Michael Jang RHCSA/ RHCE Red Hat Linux Certification: Exams (Ex200 & Ex300) (Certification Press), 2011
2. Nemeth Synder& Hein, Linux Administration Handbook, Pearson Education, 2nd Edition ,2010
3. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Unix Network Programming, The sockets Networking API, Vol. 1, 3rd Edition,2014

3. (C-III): Programming in Java Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction to Java (4Lectures)

Java Architecture and Features, Understanding the semantic and syntax differences between C++ and Java, Compiling and Executing a Java Program, Variables, Constants, Keywords Data Types, Operators (Arithmetic, Logical and Bitwise) and Expressions, Comments, Doing Basic Program Output, Decision Making Constructs (conditional statements and loops) and Nesting, Java Methods (Defining, Scope, Passing and Returning Arguments, Type Conversion and Type and Checking, Built-in Java Class Methods),

2. Arrays, Strings and I/O (8Lectures)

Creating & Using Arrays (One Dimension and Multi-dimensional), Referencing Arrays Dynamically, Java Strings: The Java String class, Creating & Using String Objects, Manipulating Strings, String Immutability & Equality, Passing Strings To & From Methods, String Buffer Classes. Simple I/O using System.out and the Scanner class, Byte and Character streams, Reading/Writing from console and files.

3. Object-Oriented Programming Overview (4Lectures)

Principles of Object-Oriented Programming, Defining & Using Classes, Controlling Access to Class Members, Class Constructors, Method Overloading, Class Variables & Methods, Objects as parameters, final classes, Object class, Garbage Collection.

3. Inheritance, Interfaces, Packages, Enumerations, Autoboxing and Metadata (14lectures)

Inheritance: (Single Level and Multilevel, Method Overriding, Dynamic Method Dispatch, Abstract Classes), Interfaces and Packages, Extending interfaces and packages, Package and Class Visibility, Using Standard Java Packages (util, lang, io, net), Wrapper Classes, Autoboxing/Unboxing, Enumerations and Metadata.

4. Exception Handling, Threading, Networking and Database Connectivity (15Lectures)

Exception types, uncaught exceptions, throw, built-in exceptions, Creating your own exceptions; Multi-threading: The Thread class and Runnable interface, creating single and multiple threads, Thread prioritization, synchronization and communication, suspending/resuming threads. Using java.net package, Overview of TCP/IP and Datagram programming. Accessing and manipulating databases using JDBC.

5. Applets and Event Handling (15Lectures)

Java Applets: Introduction to Applets, Writing Java Applets, Working with Graphics, Incorporating Images & Sounds. Event Handling Mechanisms, Listener Interfaces, Adapter and Inner Classes. The

design and Implementation of GUIs using the AWT controls, Swing components of Java Foundation Classes such as labels, buttons, textfields, layout managers, menus, events and listeners; Graphic objects for drawing figures such as lines, rectangles, ovals, using different fonts. Overview of servlets.

Reference Books

1. Ken Arnold, James Gosling, David Homes, "The Java Programming Language", 4th Edition, 2005.
2. E. Balaguruswamy, "Programming with Java", 4th Edition, McGraw Hill.2009.
3. Paul Deitel, Harvey Deitel, "Java: How to Program", 10th Edition, Prentice Hall, 2011.

(C-III) LAB: Programming in Java Lab Practical: 60 Lectures

1. To find the sum of any number of integers entered as command line arguments
2. To find the factorial of a given number
3. To learn use of single dimensional array by defining the array dynamically.
4. Write a program to create a ―distance‖ class with methods where distance is computed in terms of feet and inches, how to create objects of a class and to see the use of this pointer
5. Modify the ―distance‖ class by creating constructor for assigning values (feet and inches) to the distance object. Create another object and assign second object as reference variable to another object reference variable. Further create a third object which is a clone of the first object.
6. Write a program to show that during function overloading, if no matching argument is found, then java will apply automatic type conversions(from lower to higher data type)
7. Write a program to show the use of static functions and to pass variable length arguments in a function.
8. Write a program to demonstrate the concept of boxing and unboxing.
9. Write a program to create your own exception types to handle situation specific to your application (Hint: Define a subclass of Exception which itself is a subclass of Throwable).
10. Write a program to demonstrate priorities among multiple threads.
11. Write a program to demonstrate multithread communication by implementing synchronization among threads (Hint: you can implement a simple producer and consumer problem).
12. Write a program to get the URL/location of code (i.e. java code) and document(i.e. html file).
13. Write a program to demonstrate different mouse handling events like mouse Clicked(), mouse Entered(), mouse Exited(), mouse Pressed, mouse Released() and mouse Dragged().
14. Write a program to demonstrate the use of push buttons.

4. (C-IV): Discrete Structures Theory: 60 Lectures
Total Marks: 100; Theory: 70; Internal Assessment: 30

1.Introduction: (15 Lectures)
Sets - finite and Infinite sets, uncountably Infinite Sets; functions, relations, Properties of Binary Relations, Closure, Partial Ordering Relations; counting - Pigeonhole Principle, Permutation and Combination; Mathematical Induction, Principle of Inclusion and Exclusion.
2.Growth of Functions: (8 Lectures)
Asymptotic Notations, Summation formulas and properties, Bounding Summations, approximation by Integrals
3.Recurrences: (10 Lectures)
Recurrence Relations, generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Recurrence Trees, Master Theorem
4.Graph Theory (15 Lectures)
Basic Terminology, Models and Types, multigraphs and weighted graphs, Graph Representaion, Graph

Isomorphism, Connectivity, Euler and Hamiltonian Paths and Circuits, Planar Graphs, Graph Coloring, Trees, Basic Terminology and properties of Trees, Introduction to Spanning Trees

5.Prepositional Logic (12 Lectures)

Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory

Recommended Books:

1. C.L. Liu , D.P. Mahopatra, Elements of Discrete mathematics, 2nd Edition , Tata McGraw Hill, 1985,
2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition, McGraw Hill 2006

(C-IV) LAB: Discrete Structures Tutorial **Tutorial: 15 lectures**

5. (C-V): Data Structures Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Arrays (5 Lectures)

Single and Multi-dimensional Arrays, Sparse Matrices (Array and Linked Representation)

2. Stacks (5 Lectures)

Implementing single / multiple stack/s in an Array; Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Applications of stack; Limitations of Array representation of stack

3. Linked Lists (10 Lectures)

Singly, Doubly and Circular Lists (Array and Linked representation); Normal and Circular representation of Stack in Lists; Self Organizing Lists; Skip Lists

4. Queues (5 Lectures)

Array and Linked representation of Queue, De-queue, Priority Queues

5. Recursion (5 lectures)

Developing Recursive Definition of Simple Problems and their implementation; Advantages and Limitations of Recursion; Understanding what goes behind Recursion (Internal Stack Implementation)

6. Trees (20 Lectures)

Introduction to Tree as a data structure; Binary Trees (Insertion, Deletion , Recursive and Iterative Traversals on Binary Search Trees); Threaded Binary Trees (Insertion, Deletion, Traversals); Height-Balanced Trees (Various operations on AVL Trees).

7. Searching and Sorting (5 Lectures)

Linear Search, Binary Search, Comparison of Linear and Binary Search, Selection Sort, Insertion Sort, Insertion Sort, Shell Sort, Comparison of Sorting Techniques

8. Hashing (5 Lectures)

Introduction to Hashing, Deleting from Hash Table, Efficiency of Rehash Methods, Hash Table Reordering, Resolving collusion by Open Addressing, Coalesced Hashing, Separate Chaining, Dynamic and Extendible Hashing, Choosing a Hash Function, Perfect Hashing Function.

Reference Books:

1. Adam Drozdek, "Data Structures and algorithm in C++", Third Edition, Cengage Learning, 2012.
. Robert L. Kruse, "Data Structures and Program Design in C++", Pearson, 1999.
2. D.S Malik, Data Structure using C++, Second edition, Cengage Learning, 2010.
3. Mark Allen Weiss, "Data Structures and Algorithms Analysis in Java", Pearson Education, 3rd edition, 2011
4. Aaron M. Tenenbaum, Moshe J. Augenstein, YedidyahLangsam, "Data Structures Using Java, 2003.
5. Robert Lafore, "Data Structures and Algorithms in Java, 2/E", Pearson/ Macmillan Computer Pub, 2003

(C-V) LAB: Data Structures Lab Practical: 60 Lectures

1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. Implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
6. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
7. WAP to reverse the order of the elements in the stack using additional stack.
8. WAP to reverse the order of the elements in the stack using additional Queue.
10. WAP to implement Diagonal Matrix using one-dimensional array.
11. WAP to implement Lower Triangular Matrix using one-dimensional array.

6. (C-VI): Operating Systems Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1.Introduction (10 Lectures)
Basic OS functions, resource abstraction, types of operating systems–multiprogramming systems, batch systems , time sharing systems; operating systems for personal computers & workstations, process control & real time systems.

2. Operating System Organization (6 Lectures)
Processor and user modes, kernels, system calls and system programs.

3.Process Management (20Lectures)
System view of the process and resources, process abstraction, process hierarchy, threads, threading issues, thread libraries; Process Scheduling, non-pre-emptive and pre-emptive scheduling algorithms; concurrent and processes, critical section, semaphores, methods for inter- process communication; deadlocks.

4. Memory Management (10 Lectures)
Physical and virtual address space; memory allocation strategies –fixed and variable partitions, paging, segmentation, virtual memory

5. File and I/O Management (10 Lectures)
Directory structure, file operations, file allocation methods, device management.

6. Protection and Security (4 Lectures)
Policy mechanism, Authentication, Internal access Authorization.

Recommended Books:

Text Book:

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8th Edition, John Wiley Publications 2008.

Reference Books:

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.

(C-VI) LAB: Operating Systems Lab Practical: 60 Lectures
C/ C++ programs

1. WRITE A PROGRAM (using fork() and/or exec() commands) where parent and child execute: a) same program, same code. b) same program, different code. - c) before terminating, the parent waits for the child to finish its task.
2. WRITE A PROGRAM to report behaviour of Linux kernel including kernel version, CPU type and model. (CPU information)
3. WRITE A PROGRAM to report behaviour of Linux kernel including information on configured memory, amount of free and used memory. (memory information)
4. WRITE A PROGRAM to print file details including owner access permissions, file access time, where file name is given as argument.
5. WRITE A PROGRAM to copy files using system calls.
6. Write program to implement FCFS scheduling algorithm.
7. Write program to implement Round Robin scheduling algorithm.
8. Write program to implement SJF scheduling algorithm.
9. Write program to implement non-preemptive priority based scheduling algorithm.
10. Write program to implement preemptive priority based scheduling algorithm.
11. Write program to implement SRJF scheduling algorithm.
12. Write program to calculate sum of n numbers using thread library.
13. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.

7. (C-VII): Computer Networks Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction to Computer Networks (8 Lecture)
DATABASE SYSTEM CONCEPTS & ARCHITECTURE: Network definition; network topologies; network classifications; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite.

2. Data Communication Fundamentals and Techniques (10 Lectures)
Analog and digital signal; data-ratelimits; digital to digital line encoding schemes; pulse code modulation; parallel and serial transmission; digital to analog modulation-; multiplexing techniques- FDM, TDM; transmission media.

3. Networks Switching Techniques and Access mechanisms (10 Lectures)
Circuit switching; packetswitching- connectionless datagram switching, connection-oriented virtual circuit switching; dial-up modems; digital subscriber line; cable TV for data transfer.
4. Data Link Layer Functions and Protocol (10 Lectures)
Error detection and error correction techniques; data-link control- framing and flow control; error recovery protocols- stop and wait ARQ, go-back-n ARQ; Point to Point Protocol on Internet.
5. Multiple Access Protocol and Networks (5 Lectures)
CSMA/CD protocols; Ethernet LANS; connecting LAN and back-bone networks- repeaters, hubs, switches, bridges, router and gateways;
6. Networks Layer Functions and Protocols (6 Lectures)
Routing; routing algorithms; network layer protocol of Internet- IP protocol, Internet control protocols.
7. Transport Layer Functions and Protocols (6 Lectures)
Transport services- error and flow control, Connection establishment and release- three way handshake;
8. Overview of Application layer protocol (5 Lectures)
Overview of DNS protocol; overview of WWW &HTTP protocol.

Recommended Books:
Text Book:
1. B. A. Forouzan: Data Communications and Networking, Fourth edition, THM, 2007.
Reference Books:
2. A. S. Tanenbaum: Computer Networks, Fourth edition, PHI , 2002
3. William Stallings- Data & Communications, 6thEdition, Pearson Education.

(C-VII) LAB: Computer Networks Lab Practical: 60 Lectures

1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm
6. Simulate and implement Dijkstra algorithm for shortest path routing.

8. (C-VIII): Design and Analysis of Algorithms Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction (8 Lectures)
Basic Design and Analysis techniques of Algorithms, Correctness of Algorithm.

2. Algorithm Design Techniques (12 Lectures)
Iterative techniques, Divide and Conquer, Dynamic Programming, Greedy Algorithms.

3.Sorting and Searching Techniques (20 Lectures)
Elementary sorting techniques–Bubble Sort, Insertion Sort, Merge Sort, Advanced Sorting techniques - Heap Sort, Quick Sort, Sorting in Linear Time - Bucket Sort, Radix Sort and Count Sort, Searching Techniques, Medians & Order Statistics, complexity analysis;

4. Lower Bounding Techniques (10 Lectures)
Decision Trees

5. Graphs (10 Lectures)
Graph Algorithms–Breadth First Search, Depth First Search and its Applications, Minimum Spanning Trees.

Recommended Books:
1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, PHI, 3rd Edition 2009
2. Sarabasse& A.V. Gelder Computer Algorithm – Introduction to Design and Analysis, Publisher – Pearson 3rd Edition 1999

(C-VIII) LAB:
Design and Analysis of Algorithms Lab Practical: 60 Lectures

1. i. Implement Insertion Sort (The program should report the number of comparisons)
   ii. Implement Merge Sort (The program should report the number of comparisons)
2. Implement Heap Sort (The program should report the number of comparisons)
3. Implement Randomized Quick sort (The program should report the number of comparisons)
4. Implement Radix Sort
5. Create a Red-Black Tree and perform following operations on it:
   i. Insert a node
   ii. Delete a node
   iii. Search for a number & also report the color of the node containing this number.
6. Write a program to determine the LCS of two given sequences
7. Implement Breadth-First Search in a graph
8. Implement Depth-First Search in a graph
9. Write a program to determine the minimum spanning tree of a graph
   For the algorithms at S.No 1 to 3 test run the algorithm on 100 different inputs of sizes varying from 30 to 1000. Count the number of comparisons and draw the graph. Compare it with a graph of nlogn.

9. (C-IX): Software Engineering Theory : 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1.Introduction (18 Lectures)
The Evolving Role of Software, Software Characteristics, Changing Nature ofSoftware, Software Engineering as a Layered Technology, Software Process Framework, Framework and Umbrella Activities, Process Models, Capability Maturity Model Integration (CMMI).

2.Requirement Analysis (12 Lectures)
Software Requirement Analysis, Initiating Requirement EngineeringProcess, Requirement Analysis and Modeling Techniques, Flow Oriented Modeling, Need for SRS, Characteristics and Components of SRS.

3. Software Project Management (10Lectures)
Estimation in Project Planning Process, Project Scheduling.

4. Risk Management (12 Lectures)
Software Risks, Risk Identification, Risk Projection and Risk Refinement, RMMM Plan.

5. Quality Management (8 Lectures)
Quality Concepts, Software Quality Assurance, Software Reviews, Metrics for Process and Projects.

Recommended Books:
1. R.S. Pressman, Software Engineering: A Practitioner's Approach (7th Edition), McGraw- Hill, 2009.
2. P. Jalote, An Integrated Approach to Software Engineering (2nd Edition), Narosa Publishing House, 2003.
3. R. Mall, Fundamentals of Software Engineering (2nd Edition), Prentice-Hall of India, 2004.

(C-IX) LAB: Software Engineering Lab Practical: 60 Lectures

S. No	Practical Title
1	• 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:

1. **Criminal Record Management:** Implement a criminal record management system for jailers, police officers and CBI officers
2. **DTC Route Information:** Online information about the bus routes and their frequency and fares
3. **Car Pooling:** To maintain a web based intranet application that enables the corporate employees within an organization to avail the facility of carpooling effectively.
4. Patient Appointment and Prescription Management System
5. Organized Retail Shopping Management Software
6. Online Hotel Reservation Service System
7. Examination and Result computation system
8. Automatic Internal Assessment System
9. Parking Allocation System
10. Wholesale Management System

10. (C-X): Database Management Systems Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Database System Concepts & Architecture: (8 lectures)
Data Independence, Schemas, Instances, Database Languages, Database System Environments Data Models, Basic Structure of Oracle System, Storage Organization in Oracle.

2. Data Modeling: (10 lectures)
Use of High –level Conceptual Data Models, ER Diagrams, Subclasses, Superclasses and Inheritance, Specialization & Generalization, Conceptual Object Modeling using UML Class Diagrams, Knowledge Representation Concepts, Exercises.

3. Relational Data Model: (10 lectures)
Relational Constraints, Domain Constraint s, Key Constraints Referential Integrity Constraints, Relational Algebra, Fundamental Operations of Relational Algebra & their Implementation, Interdependence of Operations, Example Queries.

4. ER And EER To Relational Mapping: (8 lectures)
Mapping EER Model Concepts to Relation, Tuple Relational Calculas, Domain Relational Calculas Queries.

5. Database Design: (8 lectures)
Functional Dependencies, Irreducible Sets of Dependencies, Nonloss Decomposition, 1st, 2nd& 3rd NF, Dependency Preservation, Boyce Codd NF, Multivalued Dependency & 4thNF, Join Dependency & 5NF, Domain Key Normal Form, Restriction –Union Normal Form, Denormalization.

6. Query Processing And Optimization: (10 lectures)
SQL- Basic Queries in SQL, Subqueries, Retrieving a Query Plan – Table Space Span & I/O, Index Scan, Equal Unique Index Lookup, Clustered vs. Non Clustered Indexing, Index Only Scan, Methods for Joining Tables –Nested Loop Join Merge Join, Hybrid Join, Multiple table Join, Transforming Nested Queries to Joins, Object Relational SQL, Procedural SQL, Introduction to Embedded SQL.

7. Database Security & Authorization: (6 lectures)
Specifying Privileges, Revoking Privileges, Propagation of Privileges, Statistical Database Security.

Text Books:
1.Fundamental of Database Systems- Elmasri Navathe- Pearson Education Asia
2.Database- Principles, Programming and Performance- Parick O' Neil Elizabeth O' Niel, Harcort Asia PTE Limited

References Books:
1.An Introduction to Database Systems- C.J.Date, Addison Wesley, Pearson Education Press
2.Database System Concepts- Abraham Silberschat, Henry F. Korth, S.Sudarshan, Tata McGraw Hill.

(C-X) LAB: Database Management Systems Lab Practical: 60 Lectures

Create and use the following database schema to answer the given queries.
EMPLOYEE Schema

Field	Type	NULL KEY		DEFAULT
Eno	Char(3)	NO	PRI	NIL
Ename	Varchar(50)	NO		NIL
Job_ type	Varchar(50)	NO		NIL
Manager	Char(3)	Yes	FK	NIL
Hire_date	Date	NO		NIL
Dno	Integer	YES	FK	NIL
Commission	Decimal(10,2)	YES		NIL
Salary	Decimal(7,2)	NO		NIL

DEPARTMENT Schema

Field	Type	NULL KEY		DEFAULT
Dno	Integer No	No	PRI	NULL
Dname	Varchar(50)	Yes		NULL
Location	Varchar(50)	Yes		New Delhi

Query List

1. Query to display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.
2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than $2850.
6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of $1500 and $2850.
8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don‘t have a current Manager.
11. Query to display the Name, Salary and Commission for all the employees who earn commission.
12. Sort the data in descending order of Salary and Commission.
13. Query to display Name of all the employees where the third letter of their name is _A‘.
14. Query to display the names and salaries of all employees who report to King.
15. Query to display the department no, name and job for all employees in the Sales department.

11. (C-XI): Internet Technologies Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Java (5 lectures)
Use of Objects, Array and Array List class
2. JavaScript (15 lectures)
Data types, operators, functions, control structures, events and event handling.

3. JDBC (10 lectures)
JDBC Fundamentals, Establishing Connectivity and working with connection interface, Working with statements, Creating and Executing SQL Statements, Working with Result Set Objects.
4. JSP (20 lectures)
Introduction to Java Server Pages, HTTP and Servlet Basics, The Problem with Servlets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC, Setting Up the JSP Environment, Implicit JSP Objects, Conditional Processing, Displaying Values, Using an expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Database Access.
5. Java Beans (10 lectures)
Java Beans Fundamentals, JAR files, Introspection, Developing a simple Bean, Connecting to DB

Recommended 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. Jim Keogh, The Complete Reference J2EE, TMH, , 2002.
5. O'Reilly, Java Server Pages, Hans Bergsten, Third Edition, 2003.

(C-XI) LAB: Internet Technologies Lab Practical: 60 Lectures

Create event driven program for following:
1. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
2. Print the largest of three numbers.
3. Find the factorial of a number n.
4. Enter a list of positive numbers terminated by Zero. Find the sum and average of these numbers.
5. A person deposits Rs 1000 in a fixed account yielding 5% interest. Compute the amount in the account at the end of each year for n years. Read n numbers. Count the number of negative numbers, positive numbers and zeros in the list.

12. (C-XII): Theory of Computation Theory: 60 Lectures
Total Marks: 100; Theory: 70; Internal Assessment: 30

1. Languages (8 Lectures)
Alphabets, string, language, Basic Operations on language, Concatenation, Kleene Star
2. Finite Automata **(10 lectures)**
Transition Graphs, Deterministics and non-deterministic finite automata, NFA to DFA Conversion,Minimization.
3. Regular Languages (15 Lectures)
Regular Expressions, , Regular languages and their relationship with finite automata, Pumping lemma and closure properties of regular languages.
4. Context free languages (17 Lectures)
Context free grammars, parse trees, ambiguities in grammars and languages, Pushdown automata (Deterministic and Non-deterministic), Pumping Lemma, Properties of context free languages, normal forms.
5. Turing Machines and Models of Computations (10 Lectures)
Only introduction to Turing Machine as a model of computation.

Recommended Books:

1. Hoperoft, Aho, Ullman, Introduction to Automata theory, Language & Computation –3rd Edition, Pearson Education. 2006
2. P. Linz, An Introduction to Formal Language and Automata 4th edition Publication Jones Bartlett, 2006.

(C-XII): Theory of Computation Tutorial: 15 Lectures

13. (C-XIII): Artificial Intelligence Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction (06 Lectures)
Introduction to Artificial Intelligence, Background and Applications, Turing Test and Rational Agent approaches to AI, Introduction to Intelligent Agents, their structure, behavior and environment.

2. Problem Solving and Searching Techniques (18 Lectures)
Problem Characteristics, Production Systems, Control Strategies, Breadth First Search, Depth First Search, Hill climbing and its Variations, Heuristics Search Techniques: Best First Search, A* algorithm, Constraint Satisfaction Problem, Means-End Analysis, Introduction to Game Playing, Min-Max and Alpha-Beta pruning algorithms.

3. Knowledge Representation (16 Lectures)
Introduction to First Order Predicate Logic, Resolution Principle, Unification, Semantic Nets, Conceptual Dependencies, Frames, and Scripts, Production Rules, Conceptual Graphs. Programming in Logic (PROLOG)

4. Dealing with Uncertainty and Inconsistencies (08 Lectures)
Truth Maintenance System, Default Reasoning, Probabilistic Reasoning, Bayesian Probabilistic Inference, Possible World Representations.

5. Understanding Natural Languages (06 Lectures)
Parsing Techniques, Context-Free and Transformational Grammars, Recursive and Augmented Transition Nets.

6. Expert Systems Architectures: (06 Lectures)
Introduction, Rule Based System Architecture, Non-Production System Architecture, Dealing with uncertainty, Knowledge Acquisition and Validation, Knowledge System Building Tools.

Text Book:

1. Dan W. Patterson - Introduction to Artificial Intelligence and Expert Systems, PHI, New Delhi, 2006.

Reference Books:

1. E. Rich & K. Knight - Artificial Intelligence, 2/e, TMH, New Delhi, 2005.
2. P.H. Winston - Artificial Intelligence, 3/e, Pearson Edition, New Delhi, 2006.
3. D.W. Rolston,- Principles of AI & Expert System Development, TMH, New Delhi.

(C-XIII) LAB: Artificial Intelligence Lab Practical: 60 Lectures

1. Write a prolog program to calculate the sum of two numbers.
2. Write a prolog program to find the maximum of two numbers.
3. Write a prolog program to calculate the factorial of a given number.
4. Write a prolog program to calculate the nth Fibonacci number.
5. Write a prolog program, insert_nth(item, n, into_list, result) that asserts that result is the list into_list with item inserted as the n‘th element into every list at all levels.

6. Write a Prolog program to remove the Nth item from a list.
7. Write a Prolog program, remove-nth (Before, After) that asserts the After list is the Before list with the removal of every n'th item from every list at all levels.
8. Write a Prolog program to implement append for two lists.
9. Write a Prolog program to implement palindrome (List).
10. Write a Prolog program to implement max(X,Y,Max) so that Max is the greater of two numbers X and Y.
11. Write a Prolog program to implement maxlist(List,Max) so that Max is the greatest number in the list of numbers List.
12. Write a Prolog program to implement sumlist (List, Sum) so that Sum is the sum of a given list of numbers List.
13. Write a Prolog program to implement two predicates even length (List) and odd length (List) so that they are true if their argument is a list of even or odd length respectively.
14. Write a Prolog program to implement reverse (List, Reversed List) that reverses lists.
15. Write a Prolog program to implement maxlist (List, Max) so that Max is the greatest number in the list of numbers List using cut predicate.
16. Write a Prolog program to implement GCD of two numbers.
17. Write a prolog program that implements Semantic Networks/Frame Structures.

14. (C-XIV): Computer Graphics Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction (5 Lectures)
Basic elements of Computer graphics, Applications of Computer Graphics.

2. Graphics Hardware (8 Lectures)
Architecture of Raster and Random scan display devices, input/output devices.

3. Fundamental Techniques in Graphics (22 Lectures)
Raster scan line, circle and ellipse drawing, thick primitives, Polygon filling, line and polygon clipping algorithms, 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections-Parallel and Perspective), Vanishing points.

4. Geometric Modeling (10 Lectures)
Representing curves & Surfaces.

5. Visible Surface determination (8 Lectures)
Hidden surface elimination.

6. Surface rendering (7 Lectures)
Illumination and shading models. Basic color models and Computer Animation.

Books Recommended:
1. J.D.Foley, A.Van Dan, Feiner, Hughes Computer Graphics Principles & Practice 2nd edition Publication Addison Wesley 1990.
2. D.Hearn, Baker: Computer Graphics, Prentice Hall of India 2008.
3. D.F.Rogers Procedural Elements for Computer Graphics, McGraw Hill 1997.
4. D.F.Rogers, Adams Mathematical Elements for Computer Graphics, McGraw Hill 2nd edition 1989.

(C-XIV) LAB: Computer Graphics Lab Practical: 60 Lectures

1. Write a program to implement Bresenham‘s line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to apply various 2D transformations on a 2D object (use homogenous coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw Hermite/Bezier curve.

Discipline Specific Elective Papers COMPUTER SCIENCE: (Credit: 06 each) (4 papers to be selected) – DSE 1 - 4

DSE-1 (any one from the following)

I. Cloud Computing: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Overview of Computing Paradigm (8 lectures)
Recent trends in Computing: Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing,

2. Introduction to Cloud computing (7 lectures)
Introduction to Cloud Computing, History of Cloud Computing, Cloud service providers, Benefits and limitations of Cloud Computing,

3. Cloud Computing Architecture (20 lectures)
Comparison with traditional computing architecture (client/server), Services provided at various levels, Service Models- Infrastructure as a Service(IaaS), Platform as a Service(PaaS), Software as a Service(SaaS), How Cloud Computing Works, Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud, Case study of NIST architecture.

4. Case Studies (13 lectures)
Case study of Service model using Google App Engine, Microsoft Azure, Amazon EC2 , Eucalyptus.

5. Service Management in Cloud Computing (7 lectures)
Service Level Agreements (SLAs), Billing & Accounting, and Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling.

6. Security (5 lectures)
Infrastructure Security- Network level security, Host level security, Application level security, Data security and Storage- Data privacy and security Issues, Jurisdictional issues raised by Data location, Authentication in cloud computing.

Books Recommended:

Text Books:
1. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

Reference Books:

2. Cloud Computing: Principles and Paradigms, Editors: Rajkumarbuyya, James Broberg, Andrzej M. Goscinski, Wile, 2011
3. Cloud Computing: Principles, Systems and Applications, Editors: Nikos Antonopoulos, Lee Gillam, Springer, 2012
4. Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Ronald L. Krutz, Russell Dean Vines, Wiley-India, 2010
5. Gautam Shroff, Enterprise Cloud Computing Technology Architecture Applications, Adobe Reader Ebooks Available From Ebooks.Com, 2010
6. Toby Velte, Anthony Velte, Robert Elsenpeter, Cloud Computing, A Practical Approach, Mcgraw Hills, 2010.
7. Dimitris N. Chorafas, Cloud Computing Strategies, Crc Press, 2010

Cloud Computing Lab

1. Create virtual machines that access different programs on same platform.
2. Create virtual machines that access different programs on different platforms .
3. Working on tools used in cloud computing online-
 a) Storage
b) Sharing of data
c) manage your calendar, to-do lists,
d) a document editing tool
4. Exploring Google cloud
5. Exploring microsoft cloud
6. Exploring amazon cloud

II. Microprocessor Theory: 60 Lectures

Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

Microprocessor architecture: Internal architecture, system bus architecture, memory and I/O interfaces. Microprocessor programming: Register Organization, instruction formats, assembly language programming.

Interfacing: Memory address decoding, cache memory and cache controllers, I/O interface, keyboard, display, timer, interrupt controller, DMA controller, video controllers, communication interfaces.

Recommended Books:

1. Barry B. Brey: The Intel Microprocessors : Architecture, Programming and Interfacing. Pearson Education, Sixth Edition, 2009.
2. Walter A Triebel, Avtar Singh; The 8088 and 8086 Microprocessors Programming, Interfacing, Software, Hardware, and Applications. PHI, Fourth Edition 2005.

Microprocessor Lab Practical: 60 Lectures

ASSEMBLY LANGUAGE PROGRAMMING
1. Write a program for 32-bit binary division and multiplication
2. Write a program for 32-bit BCD addition and subtraction
3. Write a program for Linear search and binary search.
4. Write a program to add and subtract two arrays
5. Write a program for binary to ascii conversion
6. Write a program for ascii to binary conversion

III. Systems Programming Theory: 60 lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Assemblers & Loaders, Linkers: (10Lecture)
One pass and two pass assembler, design of an assembler, Absolute loader, relocation and linking concepts, relocating loader and Dynamic Linking.
2. Introduction: (10Lecture)
Overview of compilation, Phases of a compiler.
3. Lexical Analysis: (10Lecture)
Role of a Lexical analyzer, Specification and recognition of tokens, Symbol table, lex
4. Parsing: (10Lecture)
Bottom up parsing- LR parser, yacc.
5. Intermediate representations: **(**10Lecture)
Three address code generation, syntax directed translation, translation of types, control statements
6. Storage organization: (5Lecture)
Activation records, stack allocation
7. Code Generation: (5Lecture)
Object code generation

Reference Books
1. Santanu Chattopadhyaya, Systems Programming, PHI, 2011.
2. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers: Principles, Techniques, and Tools, 2nd edition, Prentice Hall, 2006.
3. D. M. Dhamdhere, Systems Programming, Tata McGraw Hill, 2011.
4. Leland Beck, D. Manjula, System Software: An Introduction to System Programming, 3rd edition, Pearson Education, 2008.
5. Grune D, Van Reeuwijk . K, Bal H. E, Jacobs C J H, Langendoen K, Modern Compiler Design, 2nd edition, Springer, 2012

Systems Programming Lab Practical: 60 lectures
1. To implement an assembler for a hypothetical language.
2. To get familiar with lex: write a program to recognize numbers, identifiers.
3. To get familiar with yacc: write a desk calculator.

DSE-2 (any one from the following)

I. Network Programming Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

Transport Layer Protocols: TCP, UDP, SCTP protocol.
Socket Programming: Socket Introduction; TCP Sockets; TCP Client/Server Example ; signal handling; I/O multiplexing using sockets; Socket Options; UDP Sockets; UDP client server example; Address lookup using sockets.
Network Applications: Remote logging; Email; WWW and HTTP.
LAN administration: Linux and TCP/IP networking: Network Management and Debugging.

Recommended Books:

1. W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Unix Network Programming, The sockets Networking API, Vol. 1, 3rd Edition, PHI.2003
2. B. A. Forouzan: Data Communications and Networking, Fourth edition, THM Publishing Company Ltd., 2003
3. Nemeth Synder & Hein, Linux Administration Handbook, Pearson Education, 2nd Edition, 2010
4. R. Stevens, Unix Network Programming, PHI 2nd Edition, 1990

Network Programming Lab Practical: 60 Lectures

Practical exercises based on concepts listed in theory.

II. Information Security Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction Security, Attacks, Computer Criminals, Security Services, Security Mechanisms.

2. Cryptography Substitution ciphers, Transpositions Cipher, Confusion, diffusion, Symmetric, Asymmetric Encryption. DES Modes of DES, Uses of Encryption, Hash function, key exchange, Digital Signatures, Digital Certificates.

3. Program Security Secure programs, Non malicious Program errors, Malicious codes virus, Trap doors, Salami attacks, Covert channels, Control against program

4. Threats. Protection in OS: Memory and Address Protection, Access control, File Protection, User Authentication.

5. Database Security Requirements, Reliability, Integrity, Sensitive data, Inference, Multilevel Security.

6. Security in Networks Threats in Networks, Security Controls, firewalls, Intrusion detection systems, Secure e-mails

7. Administrating Security Planning, Risk Analysis, Organisational Security Policy, Physical Security. Ethical issues in Security: Protecting Programs and data. Information and law.

Recommended Books:

1. C. P. Pfleeger, S. L. Pfleeger; Security in Computing, Prentice Hall of India, 2006
2. W. Stallings; Network Security Essentials: Applications and Standards, 4/E, 2010

Information Security Lab Practical: 60 lectures

1. Demonstrate the use of Network tools: ping, ipconfig, ifconfig, tracert, arp, netstat, whois
2. Use of Password cracking tools: John the Ripper, Ophcrack. Verify the strength of passwords using these tools.
3. Perform encryption and decryption of Caesar cipher. Write a script for performing these operations.
4. Perform encryption and decryption of a Rail fence cipher. Write a script for performing these operations.
5. Use nmap/zenmap to analyse a remote machine.
6. Use Burp proxy to capture and modify the message.
7. Demonstrate sending of a protected word document.
8. Demonstrate sending of a digitally signed document.
9. Demonstrate sending of a protected worksheet.

10. Demonstrate use of steganography tools.
11. Demonstrate use of gpg utility for signing and encrypting purposes.

III. Operational Research Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction to Operational Research (OR): Origin & Development, Different Phases of OR study, Methodology of OR, Scope and Limitations of OR, Applications of OR.
2. Linear Programming: Linearly independent / dependent vectors, Basis, Convex sets, Extreme points. Graphical method. Simplex method, Artificial variable techniques- Two Phase Method; M-Charnes Method, Special cases in LPP.
3. Duality: Definition of the dual problem, Primal-dual relationships, Economic Interpretation of Duality, Dual simplex Method.
4. Sensitivity analysis: Changes in cost and resource vector

Reference Books

1. G. Hadley: Linear Programming. Narosa, 2002 (reprint).
2. A. Ravindran, D. T. Phillips and James J. Solberg: Operations Research-Principles and Practice, John Wiley & Sons, 2005.
3. Hamdy A. Taha: Operations Research-An Introduction, Prentice Hall, 8th Edition, 2008.
4. F.S. Hillier. G.J. Lieberman: Introduction to Operations Research- Concepts and Cases, 9th Edition, Tata McGraw Hill. 2010.

Operational Research Lab Practical: 60 Lectures

1. To solve Linear Programming Problem using Graphical Method with Unbounded solution Infeasible solution Alternative or multiple solutions.
2. Solution of LPP with simplex method.
3. Problem solving using M-Charnes method.
4. Problem solving using Two Phase method.
5. Illustration of following special cases in LPP using Simplex method Unrestricted variables Unbounded solution Infeasible solution Alternative or multiple solution
6. Problems based on Dual simplex method.
7. Problems based on sensitivity analysis.

DSE-3 (any one from the following)

I. Data Mining Theory: 60 lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Data Mining: (8 lectures)
Introduction, Relational Databases, Data Warehouses, Transactional databases, Advanced database Systems and Application, Data Mining Functionalities, Classification of Data Mining Systems, Major Issues in Data Mining.
2. Data Processing: (8 lectures)
Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and concept Hierarchy Generation.
3. Data Mining Primitives, Languages and System Architecture: (8 lectures)
Data Mining Primities, DMQL, Architectures of Data Mining Systems.
4. Concept Description : (10 lectures)
Data Generalization & Summarization – Based Characterization, Analytical Characterization, Mining class Comparisons, Mining Descriptive Statistical Measures in Large Databases.

5. Mining Association Rules in Large Databases: (12 lectures)
Association Rule Mining, Single – Dimensional Boolean Association Rules, Multilevel Association Rules from Transaction Databases, Multi-Dimensional Association Rules from Relational Databases, From Association Mining to Correlation Analysis, Constraint – Based Association Mining.

6. Classification and Prediction: (14 lectures)
Classification & Prediction, Issues Regarding Classification & Prediction, Classification by decision Tree Induction, Bayesian Classification, Classification by Back propagation, Classification based on concepts & Association Rule, Other Classification, Prediction, Classification Accuracy.

Text Books:
1. Jiawei Han & Micheline Kamber - Data Mining Concepts & Techniques Publisher Harcout India. Private Limited.

Reference Books:
1.G.K. Gupta – Introduction to Data Mining with case Studies, PHI, New Delhi – 2006.
2.A. Berson & S.J. Smith – Data Warehousing Data Mining, COLAP, TMH, New Delhi – 2004.
3.H.M. Dunham & S. Sridhar – Data Mining, Pearson Education, New Delhi, 2006.

Data Mining Lab Practical: 60 lectures

Practical exercises based on concepts listed in theory.

II. Machine Learning Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1.Introduction: Concept of Machine Learning, Applications of Machine Learning, Key elements of Machine Learning, Supervised vs. Unsupervised Learning, Statistical Learning: Bayesian Method, The Naive Bayes Classifier

2. Softwares for Machine Learning and Linear Algebra Overview: Plotting of Data, Vectorization, Matrices and Vectors: Addition, Multiplication, Transpose and Inverse using available tool such as MATLAB.

3.Linear Regression: Prediction using Linear Regression, Gradient Descent, Linear Regression with one variable, Linear Regression with multiple variables, Polynomial Regression, Feature Scaling/Selection. Logistic Regression: Classification using Logistic Regression, Logistic Regression vs. Linear Regression, Logistic Regression with one variable and with multiple variables.

4. Regularization: Regularization and its utility: The problem of Overfitting, Application of Regularization in Linear and Logistic Regression, Regularization and Bias/Variance.

5. Neural Networks: Introduction, Model Representation, Gradient Descent vs. Perceptron Training, Stochastic Gradient Descent, Multilayer Perceptrons, Multiclass Representation, Backpropagation Algorithm.

Suggested Books:
1. Ethem Alpaydin, "Introduction to Machine Learning" 2nd Edition, The MIT Press, 2009.
2. Tom M. Mitchell, "Machine Learning", First Edition by Tata McGraw-Hill Education, 2013.
3. Christopher M. Bishop, "Pattern Recognition and Machine Learning" by Springer, 2007.
4. Mevin P. Murphy, "Machine Learning: A Probabilistic Perspective" by The MIT Press, 2012.

Machine Learning Lab Practical: 60 Lectures

For practical Labs for Machine Learning, students may use softwares like MABLAB/Octave or Python. For later exercises, students can create/use their own datasets or utilize datasets from online repositories like UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/).
1. Perform elementary mathematical operations in Octave/MATLAB like addition, multiplication, division and exponentiation.
2. Perform elementary logical operations in Octave/MATLAB (like OR, AND, Checking for Equality, NOT, XOR).

3. Create, initialize and display simple variables and simple strings and use simple formatting for variable.
4. Create/Define single dimension / multi-dimension arrays, and arrays with specific values like array of all ones, all zeros, array with random values within a range, or a diagonal matrix.
5. Use command to compute the size of a matrix, size/length of a particular row/column, load data from a text file, store matrix data to a text file, finding out variables and their features in the current scope.
6. Perform basic operations on matrices (like addition, subtraction, multiplication) and display specific rows or columns of the matrix.
7. Perform other matrix operations like converting matrix data to absolute values, taking the negative of matrix values, additing/removing rows/columns from a matrix, finding the maximum or minimum values in a matrix or in a row/column, and finding the sum of some/all elements in a matrix.
8. Create various type of plots/charts like histograms, plot based on sine/cosine function based on data from a matrix. Further label different axes in a plot and data in a plot.
9. Generate different subplots from a given plot and color plot data.
10. Use conditional statements and different type of loops based on simple example/s.
11. Perform vectorized implementation of simple matrix operation like finding the transpose of a matrix, adding, subtracting or multiplying two matrices.
12. Implement Linear Regression problem. For example, based on a dataset comprising of existing set of prices and area/size of the houses, predict the estimated price of a given house.
13. Based on multiple features/variables perform Linear Regression. For example, based on a number of additional features like number of bedrooms, servant room, number of balconies, number of houses of years a house has been built – predict the price of a house.
14. Implement a classification/ logistic regression problem. For example based on different features of students data, classify, whether a student is suitable for a particular activity. Based on the available dataset, a student can also implement another classification problem like checking whether an email is spam or not.
15. Use some function for regularization of dataset based on problem 14.
16. Use some function for neural networks, like Stochastic Gradient Descent or backpropagation - algorithm to predict the value of a variable based on the dataset of problem 14.

III. Introduction to Data Science Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Data Scientist's Tool Box: Turning data into actionable knowledge, introduction to the tools that will be used in building data analysis software: version control, markdown, git, GitHub, R, and RStudio.

2.R Programming Basics: Overview of R, R data types and objects, reading and writing data, Control structures, functions, scoping rules, dates and times, Loop functions, debugging tools, Simulation, code profiling

3. Getting and Cleaning Data: Obtaining data from the web, from APIs, from databases and from colleagues in various formats. Basics of data cleaning and making data ―tidy.

4. Exploratory Data Analysis: Essential exploratory techniques for summarizing data, applied before formal modeling commences, eliminating or sharpening potential hypotheses about the world that can be addressed by the data, common multivariate statistical techniques used to visualize high-dimensional data.

5. Reproducible Research: Concepts and tools behind reporting modern data analyses in a reproducible manner, To write a document using R markdown, integrate live R code into a literate statistical program, compile R markdown documents using knitr and related tools, and organize a data analysis so that it is reproducible and accessible to others.

Reference Books

1. Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontiline" by Schroff/O'Reilly, 2013.
2. Foster Provost, Tom Fawcett, "Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking" by O'Reilly, 2013.
3. John W. Foreman, "Data Smart: Using data Science to Transform Information into Insight" by John Wiley & Sons, 2013.
4. Ian Ayres, "Super Crunchers: Why Thinking-by-Numbers Is the New Way to Be Smart" Ist Edition by Bantam, 2007.
5. Eric Seigel, "Predictive Analytics: The Power to Predict who Will Click, Buy, Lie, or Die", 1st Edition, by Wiley, 2013.
6. Matthew A. Russel, "Mining the Social Web: Data mining Facebook, Twitter, Linkedln, Goole+, GitHub, and More", Second Edition, by O'Reilly Media, 2013.

Introduction to Data Science Lab Practical: 60 Lectures

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 linear search.
8. Implement binary search.
9. Implement matrices addition, subtraction and Multiplication
10. Fifteen students were enrolled in a course. There ages were:

20 20 20 20 20 21 21 21 22 22 22 22 23 23 23

i. Find the median age of all students under 22 years
ii. Find the median age of all students
iii. Find the mean age of all students
iv. Find the modal age for all students
v. Two more students enter the class.
The age of both students is 23. What is now mean, mode and median ?
11. Following table gives a frequency distribution of systolic blood pressure. Compute all the measures of dispersion.

Midpoint	95.5	105.5	115.5	125.5	135.5	145.5	155.5	165.5	175.5
Number	5	8	22	27	17	9	5	5	2

12. Obtain probability distribution of, where X is number of spots showing when a six-sided symmetric die (i.e. all six faces of the die are equally likely) is rolled. Simulate random samples of sizes 40, 70 and 100 respectively and verify the frequency interpretation of probability.
13. Make visual representations of data using the base, lattice, and ggplot2 plotting systems in R, apply basic principles of data graphics to create rich analytic graphics from available datasets.
14. Use Git / Github software to create Github account. Also, create a repo using Github.

DSE-4 (any one from the following)

I. Computational Linguistics Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

Introduction Computers in linguistics and Natural Language Processing The nature and use of text corpora Introduction to python programming and NLTK, Regular expressions Pattern matching Corpus search and counting Regular languages Finite-state automata Operations and closure properties Pumping Lemma, Finite-state linguistics Transducers Morphological analysis N-grams Language modeling Smoothing Evaluation Part-of-Speech Tagging, Word classes and tagsets Rule-based and stochastic POS tagging Hidden Markov Models Evaluation Word meaning Semantic ambiguity Semantic relations Semantic roles Computational lexical semantics, (Un)supervised word sense disambiguation Classifiers Vector-space semantics

References:
1. Jurafsky, D. and J. H. Martin, Speech and language processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition (2nd Edition). Prentice-Hall, 2008.
2. Charniak, E.: Statistical Language Learning. The MIT Press. 1996.
3. J. Allen: Natural Language Understanding. Benjamin/Cummins.1995.

Computational Linguistics Lab Practical: 60 Lectures

Practical exercises based on concepts listed in theory.

II. Digital Image Processing Theory: 60 Lectures
Total Marks: 150; Theory: 70; Practical: 50; Internal Assessment: 30

1. Introduction (6 Lectures)
Light, Brightness adaption and discrimination, Pixels, coordinate conventions, Imaging Geometry, Perspective Projection, Spatial Domain Filtering, sampling and quantization.

2. Spatial Domain Filtering (7 Lectures)
Intensity transformations, contrast stretching, histogram equalization, Correlation and convolution, Smoothing filters, sharpening filters, gradient and Laplacian.

3. Filtering in the Frequency domain (8 Lectures)
Hotelling Transform, Fourier Transforms and properties, FFT (Decimation in Frequency and Decimation in Time Techniques), Convolution, Correlation, 2-D sampling, Discrete Cosine Transform, Frequency domain filtering.

4. Image Restoration (8 Lectures)
Basic Framework, Interactive Restoration, Image deformation and geometric transformations, image morphing, Restoration techniques, Noise characterization, Noise restoration filters, Adaptive filters, Linear, Position invariant degradations, Estimation of Degradation functions, Restoration from projections.

5. Image Compression (10 Lectures)
Encoder-Decoder model, Types of redundancies, Lossy and Lossless compression, Entropy of an information source, Shannon's 1st Theorem, Huffman Coding, Arithmetic Coding, Golomb Coding, LZW coding, Transform Coding, Sub-image size selection, blocking artifacts, DCT implementation using FFT, Run length coding, FAX compression (CCITT Group-3 and Group-4), Symbol-based coding, JBIG-2,

Bit-plane encoding, Bit-allocation, Zonal Coding, Threshold Coding, JPEG, Lossless predictive coding, Lossy predictive coding, Motion Compensation

6. Wavelet based Image Compression (5 Lectures)
Expansion of functions, Multi-resolution analysis, Scaling functions, MRA refinement equation, Wavelet series expansion, Discrete Wavelet Transform (DWT), Continuous Wavelet Transform, Fast Wavelet Transform, 2-D wavelet Transform, JPEG-2000 encoding, Digital Image Watermarking.

7. Morphological Image Processing (7 Lectures)
Basics, SE, Erosion, Dilation, Opening, Closing, Hit-or-Miss Transform, Boundary Detection, Hole filling, Connected components, convex hull, thinning, thickening, skeletons, pruning, Geodesic Dilation, Erosion, Reconstruction by dilation and erosion.

8. Image Segmentation (9 Lectures)
Boundary detection based techniques, Point, line detection, Edge detection, Edge linking, local processing, regional processing, Hough transform, Thresholding, Iterative thresholding, Otsu's method, Moving averages, Multivariable thresholding, Region-based segmentation, Watershed algorithm, Use of motion in segmentation

Reference Books
1. R C Gonzalez, R E Woods, Digital Image Processing, 3rd Edition, Pearson Education.2008.
2. A K Jain, Fundamentals of Digital image Processing, Prentice Hall of India.1989.
3. K R Castleman, Digital Image Processing, Pearson Education.1996
4. Schalkoff, Digital Image Processing and Computer Vision, John Wiley and Sons.1989.
5. Rafael C. Gonzalez, Richard E. Woods, Steven Eddins,' Digital Image Processing using MATLAB', Pearson Education, Inc., 2004.

Digital Image Processing Lab Practical: 60 Lectures

1. Write program to read and display digital image using MATLAB or SCILAB
a. Become familiar with SCILAB/MATLAB Basic commands
b. Read and display image in SCILAB/MATLAB
c. Resize given image
d. Convert given color image into gray-scale image
e. Convert given color/gray-scale image into black & white image
f. Draw image profile
g. Separate color image in three R G & B planes
h. Create color image using R, G and B three separate planes
i. Flow control and LOOP in SCILAB
j. Write given 2-D data in image file
2. To write and execute image processing programs using point processing method
a. Obtain Negative image
b. Obtain Flip image
c. Thresholding
d. Contrast stretching
3. To write and execute programs for image arithmetic operations
a. Addition of two images
b. Subtract one image from other image
c. Calculate mean value of image
d. Different Brightness by changing mean value
4. To write and execute programs for image logical operations

a. AND operation between two images
b. OR operation between two images
c. Calculate intersection of two images
d. Water Marking using EX-OR operation
e. NOT operation (Negative image)
5. To write a program for histogram calculation and equalization using
a. Standard MATLAB function
b. Program without using standard MATLAB functions
c. C Program
6. To write and execute program for geometric transformation of image
a. Translation
b. Scaling
c. Rotation
d. Shrinking
e. Zooming
7. To understand various image noise models and to write programs for
a. image restoration
b. Remove Salt and Pepper Noise
c. Minimize Gaussian noise
d. Median filter and Weiner filter
8. Write and execute programs to remove noise using spatial filters
a. Understand 1-D and 2-D convolution process
b. Use 3x3 Mask for low pass filter and high pass filter
9. Write and execute programs for image frequency domain filtering
a. Apply FFT on given image
b. Perform low pass and high pass filtering in frequency domain
c. Apply IFFT to reconstruct image
10. Write a program in C and MATLAB/SCILAB for edge detection using different edge detection mask
11. Write and execute program for image morphological operations erosion and dilation.
12. To write and execute program for wavelet transform on given image and perform inverse wavelet transform to reconstruct image.

III. Dissertation / Project work Total Marks = 100.

This option 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 three (03) students. Each group will be assigned a teacher as a supervisor who will handle both their theory as well lab classes. A maximum of Four (04) projects would be assigned to one teacher.

Skill Enhancement Courses (any two from the list given below) (Credit: 02 each) – SEC1 to SEC2
Theory: 01, Labs: 02

1. Android Programming

Total Marks = 100; Theory: 70; Internal: 30.

1. Introduction: (2L)
History of Android, Introduction to Android Operating Systems, Android Development Tools, Android Architecture.
2. Overview of object oriented programming using Java: OOPs Concepts: (4L)
Inheritance, Polymorphism, Interfaces, Abstract class, Threads, Overloading and Overriding, Java Virtual Machine.
3. Development Tools: (5L)
Installing and using Eclipse with ADT plug-in, Installing Virtual machine for Android sandwich/Jelly bean (Emulator), configuring the installed tools, creating an android project
– Hello Word, run on emulator, Deploy it on USB-connected Android device.
4. User Interface Architecture: (2L)
Application context, intents, Activity life cycle, multiple screen size s.
5. User Interface Design: (2L)
Form widgets, Text Fields, Layouts, Button control, toggle buttons, Spinners(Combo boxes),Images, Menu, Dialog.
6. Database: Understanding of SQLite database, connecting with the database. (2L)

Book Recommended:
1. Android application development for java programmers. By James C. Sheusi. Publisher: Cengage Learning, 2013.

ONLINE READING / SUPPORTING MATERIAL:
1. http://www.developer.android.com
2. http://developer.android.com/about/versions/index.html
3. http://developer.android.com/training/basics/firstapp/index.html
4. http://docs.oracle.com/javase/tutorial/index.htm (Available in the form of free downloadable ebooks also).
5. http://developer.android.com/guide/components/activities.html
6. http://developer.android.com/guide/components/fundamentals.html
7. http://developer.android.com/guide/components/intents-filters.html.
8. http://developer.android.com/training/multiscreen/screensizes.html
9. http://developer.android.com/guide/topics/ui/controls.html
10. http://developer.android.com/guide/topics/ui/declaring-layout.html
11. http://developer.android.com/training/basics/data-storage/databases.html

Software Lab Based on Android Programming:
1. Create ―Hello World‖ application. That will display ―Hello World‖ in the middle of the screen in the emulator. Also display ―Hello World‖ in the middle of the screen in the Android Phone.
2. Create an application with login module. (Check username and password).
3. Create spinner with strings taken from resource folder (res >> value folder) and on changing the spinner value, Image will change.
4. Create a menu with 5 options and and selected option should appear in text box.
5. Create a list of all courses in your college and on selecting a particular course teacher-in-charge of that course should appear at the bottom of the screen.

6. Create an application with three option buttons, on selecting a button colour of the screen will change.
7. Create and Login application as above. On successful login, pop up the message.
8. Create an application to Create, Insert, update, Delete and retrieve operation on the database.

2. Programming in Python Planning the Computer Program:

Total Marks = 100; Theory: 70; Internal: 30.

Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation. (2L)
Techniques of Problem Solving: Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming. (2L)
Overview of Programming: Structure of a Python Program, Elements of Python (3L)
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). (4L)
Creating Python Programs: Input and Output Statements, Control statements (Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments. (4L)
Reference Books
1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Python Tutorial/Documentation www.python.or 2015
3. Allen Downey, Jeffrey Elkner, Chris Meyers, How to think like a computer scientist: learning with Python, Freely available online.2012
4. http://docs.python.org/3/tutorial/index.html
5. http://interactivepython.org/courselib/static/pythonds
6. http://www.ibiblio.org/g2swap/byteofpython/read/

Software Lab Based on 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 users 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 paramters 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)
8. WAP to show a ball bouncing between 2 walls. (Optional)

3. PHP Programming (1 +2 Lab)

Total Marks = 100; Theory: 70; Internal: 30.

Introduction to PHP: (3L)
a.PHP introduction, inventions and versions, important tools and software requirements (like Web Server,
b.Database, Editors etc.)
c.PHP with other technologies, scope of PHP
d.Basic Syntax, PHP variables and constants
e.Types of data in PHP, Expressions, scopes of a variable (local, global)
f.PHP Operators: Arithmetic, Assignment, Relational, Logical operators, Bitwise, ternary and MOD operator.

PHP operator Precedence and associativity Handling HTML form with PHP: (2L)
a.Capturing Form Data
b.GET and POST form methods
c.Dealing with multi value fields
d.Redirecting a form after submission

PHP conditional events and Loops: (3L)
a.PHP IF Else conditional statements (Nested IF and Else)
b.Switch case, while, For and Do While Loop
c.Goto, Break, Continue and exit

PHP Functions: (3L)
a.Function, Need of Function, declaration and calling of a function
b.PHP Function with arguments, Default Arguments in Function
c.Function argument with call by value, call by reference

Scope of Function Global and Local String Manipulation and Regular Expression: (3L)
a.Creating and accessing String, Searching & Replacing String
b.Formatting, joining and splitting String, String Related Library functions
c.Use and advantage of regular expression over inbuilt function
d.Use of preg_match(), preg_replace(), preg_split() functions in regular expression

Array: (3L)

a) Anatomy of an Array, Creating index based and Associative array, Accessing array
b) Looping with Index based array, with associative array using each() and foreach()
c) Some useful Library function

Reference Books:

1. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
2. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited, 2008.
3. Robin Nixon, "Learning PHP, MySQL, JavaScript, CSS & HTML5", 3rd Edition Paperback, O'reilly, 2014. 4. Luke Welling, Laura Thompson, PHP and MySQL Web Development", 4th Edition, Addition Paperback, Addison-Wesley Professsional, 2008.
5. David Sklar, Adam Trachtenberg, "PHP Cookbook: Solutions & Examples for PHP Programmers", 2014.

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.

4. Software Testing (1+2 Labs)

Total Marks = 100; Theory: 70; Internal: 30.

Introduction: (4L)
Strategic Approach to Software Testing Test Strategies for Conventional Software, Validation Testing, System Testing, Basic Terminologies, V Shaped Software Lifecycle Model
Functional Testing\ Black-box Testing: (5L)
Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing
Structural Testing\ White-box Testing: (6L)
Basis Path Testing: Program Graph, DD Path graph, Cyclomatic Complexity, Graph Matrices, Control Flow Testing: Statement Coverage, Branch Coverage, Condition Coverage, Path Coverage
Books Recommended:
1. Roger S. Pressman, Software Engineering: A Practitioner‘s Approach, Seventh Edition, Mc Graw Hill Education.2009
2. Yogesh Singh, Software Testing, Cambridge University Press.2011.

Computer Lab Based on Software Testing:

1. Write a program that takes three inputs (a, b &c) that represent the sides of a triangle, and the output is one of the below four:
Not a triangle
Scalene triangle
Isosceles triangle
Equilateral triangle
1.1 Generate test cases using Boundary Value Analysis, Equivalence Class Partitioning and Decision Table Testing.
1.2 Generate test cases using Basis path testing.
1.3 Run code coverage tool.
2. Write a program that determines the nature of roots of a quadratic equation. Output should be one of the following:-

- Not a quadratic equation.
- Complex roots
- Real roots
- Single roots

i. Generate test cases using Boundary Value Analysis, Equivalence Class Partitioning and Decision Table Testing.
ii. Generate test cases using Basis path testing.
iii. Run code coverage tool

3. Write a program that checks whether the number is even or odd. Run code coverage tool and find the amount of code being covered.
4. Write a program that dynamically allocates memory to10 integers using malloc() or calloc() and

- donot free memory leading to memory leaks. Verify the same using Valgrind.
- Now, free memory using free() at the end of the program to avoid memory leaks. Verify the same using Valgrind.

5. Use Load UI load testing tool to test the web application performance.

Array: (3L)

- Anatomy of an Array, Creating index based and Associative array, Accessing array
- Looping with Index based array, with associative array using each () and for each ()
- Some useful Library function

Reference Books:

1. Steven Holzner, "PHP: The Complete Reference Paperback", McGraw Hill Education (India), 2007.
2. Timothy Boronczyk, Martin E. Psinas, "PHP and MYSQL (Create-Modify-Reuse)", Wiley India Private Limited, 2008.
3. Robin Nixon, "Learning PHP, MySQL, JavaScript, CSS & HTML5", 3rd Edition Paperback, O'reilly, 2014.
4. Luke Welling, Laura Thompson, PHP and MySQL Web Development", 4th Edition, Addition Paperback, Addison-Wesley Professional, 2008.
5. David Sklar, Adam Trachtenberg, "PHP Cookbook: Solutions & Examples for PHP Programmers", 2014.

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.