

**Scheme and Syllabus**  
(With effect from 2010-2011)

**Master of Computer Applications**

VISVESVARAYA TECHNOLOGICAL UNIVERSITY, BELGAUM

**SCHEME OF TEACHING AND EXAMINATION  
MASTER OF COMPUTER APPLICATIONS**

**I SEMESTER**

S. No.	Subject Code	Title	Teaching Hrs / Week		Examination			
			Theory	Practical	Duration	Marks		
						IA	Exam	Total
1	10MCA11	Problem Solving using C	04	-	03	50	100	150
2	10MCA12	Discrete Mathematics	04	-	03	50	100	150
3	10MCA13	Fundamentals of Computer Organization	04	-	03	50	100	150
4	10MCA14	Introduction to Unix	04	-	03	50	100	150
5	10MCA15	Professional Communication & Ethics	04	-	03	50	100	150
6	10MCA16	C Programming Laboratory	-	03	03	50	50	100
7	10MCA17	Unix Laboratory	-	03	03	50	50	100
8	10MCA18	IT and Digital Electronics Laboratory	-	03	03	50	50	100
Total			20	09		400	650	1050

**SCHEME OF TEACHING AND EXAMINATION  
MASTER OF COMPUTER APPLICATIONS**

**II SEMESTER**

S. No.	Subject Code	Title	Teaching Hrs / Week		Examination			
			Theory	Practical	Duration	Marks		
						IA	Exam	Total
1	10MCA21	Business Data Processing with COBOL	04	-	03	50	100	150
2	10MCA22	Object Oriented Programming with C++	04	-	03	50	100	150
3	10MCA23	Data Structures using C	04	-	03	50	100	150
4	10MCA24	Management Information Systems	04	-	03	50	100	150
5	10MCA25	Operations Research	04	-	03	50	100	150
6	10MCA26	COBOL Programming Laboratory	-	03	03	50	50	100
7	10MCA27	Data Structures Using C Laboratory	-	03	03	50	50	100
8	10MCA28	Object Oriented Programming with C++ Laboratory	-	03	03	50	50	100
Total			20	09		400	650	1050

**SCHEME OF TEACHING AND EXAMINATION  
MASTER OF COMPUTER APPLICATIONS**

**III SEMESTER**

S. No.	Subject Code	Title	Teaching Hrs / Week		Examination			
			Theory	Practical	Duration	Marks		
						IA	Exam	Total
1	10MCA31	Systems Software	04	-	03	50	100	150
2	10MCA32	Computer Networks	04	-	03	50	100	150
3	10MCA33	Programming with Java	04	-	03	50	100	150
4	10MCA34	Database Management Systems	04	-	03	50	100	150
5	10MCA35	Operating Systems	04	-	03	50	100	150
6	10MCA36	Systems Programming Laboratory	-	03	03	50	50	100
7	10MCA37	Java Programming Laboratory	-	03	03	50	50	100
8	10MCA38	DBMS Laboratory	-	03	03	50	50	100
Total			20	09		400	650	1050

**SCHEME OF TEACHING AND EXAMINATION  
MASTER OF COMPUTER APPLICATIONS**

**IV SEMESTER**

S. No.	Subject Code	Title	Teaching Hrs / Week		Examination			
			Theory	Practical	Duration	Marks		
						IA	Exam	Total
1	10MCA41	Topics in Enterprise Architectures – I	04	-	03	50	100	150
2	10MCA42	Software Engineering	04	-	03	50	100	150
3	10MCA43	Web Programming	04	-	03	50	100	150
4	10MCA44	Design and Analysis of Algorithms	04	-	03	50	100	150
5	10MCA45	Elective I	04	-	03	50	100	150
6	10MCA46	J2EE Laboratory	-	03	03	50	50	100
7	10MCA47	Web programming Laboratory	-	03	03	50	50	100
8	10MCA48	Algorithms Laboratory	-	03	03	50	50	100
Total			20	09		400	650	1050

**Elective I**

10MCA451	Computer Graphics and Visualization
10MCA452	UNIX system Programming
10MCA453	Multimedia Systems
10MCA454	Pattern Recognition
10MCA455	Principles of User Interface Design
10MCA456	Advanced Computer Networks

**SCHEME OF TEACHING AND EXAMINATION  
MASTER OF COMPUTER APPLICATIONS**

**V SEMESTER**

S. No.	Subject Code	Title	Teaching Hrs / Week		Examination			
			Theory	Practical	Duration	Marks		
						IA	Exam	Total
1	10MCA51	Object-Oriented Modeling and Design Patterns	04	-	03	50	100	150
2	10MCA52	System Simulation and Modeling	04	-	03	50	100	150
	10MCA53	Topics in Enterprise Architectures -II	04	-	03	50	100	150
4	10MCA54	Elective II	04	-	03	50	100	150
5	10MCA55	Elective III	04	-	03	50	100	150
6	10MCA56	Software Design Laboratory	-	03	03	50	50	100
7	10MCA57	.Net Laboratory	-	03	03	50	50	100
8	10MCA58	Mini Project	-	03	03	50	50	100
Total			24	06		400	650	1050

**Elective II**

10MCA541	Information Retrieval
10MCA542	Data Warehousing and Data Mining
10MCA543	Supply Chain Management
10MCA544	Network Management
10MCA545	Compiler Design
10MCA546	Software Architectures

**Elective III**

10MCA551	Information & Network Security
10MCA552	Software Testing
10MCA553	Services Oriented Architecture
10MCA554	Wireless Networks and Mobile Computing
10MCA555	Storage Area Networks
10MCA556	Web 2.0 & Rich Internet Applications

**SCHEME OF TEACHING AND EXAMINATION  
MASTER OF COMPUTER APPLICATIONS**

**VI SEMESTER**

S. No.	Subject Code	Title	Teaching Hrs / Week		Examination				
			Theory	Practical	Duration	Marks			
						IA	Dissertation	Viva	Total
1		Project Work	-	03	03	50	125	75	250

**NOTE: Students have to register for one Elective from each of the Three Elective Groups**

# I SEMESTER

## Problem Solving Using C

**Subject Code: 10MCA11**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Marks: 100**  
**Exam Hours: 3**

### **Algorithms and Flowcharts** **2 Hours**

The meaning of algorithms, Flowcharts and their need, Writing algorithms and drawing flowcharts for simple exercises like finding biggest of three numbers, to find roots of given quadratic equation, to find the biggest and smallest of given set of numbers and such other simple examples

### **Constants, Variables and Data Types** **4 Hours**

Character set, C tokens, keywords & identifiers, structure of C program, executing a C program. Constants, variables, data types, declaration of variables, declaration of storage classes, assigning values to variables defining symbolic constants, declaring a variable as constant, declaring a variable as volatile, overflow and underflow of data.

### **Operators and Expressions** **3 Hours**

Arithmetic operators, relational operators, logical operators, assignment operator, increment and decrement operator, conditional operator, bitwise operators, comma operator, special operators, arithmetic expressions, evaluation of expressions, precedence of arithmetic operators, type conversions in expressions, operator precedence and associativity, mathematical functions

### **Managing Input and Output Operations** **4 Hours**

The *scanf()* & *printf()* functions for input and output operations, reading a character, writing a character, (the *getchar()* & *putchar()* functions), the address operator(&), formatted input and output using format specifiers, Writing simple complete C programs.

### **Control Statements** **4 Hours**

Decision making with *if* statement, simple if statement, the *if..else* statement, nesting of *if..else* statements, the *else..if* ladder, the *switch* statement, the *?:* operator, the *goto* statement, the *break* statement, programming examples

### **Loop Control Structures** **3 Hours**

The *while* statement, the *do..while* statement, the *for* statement, nested loops, jumps in loops, the *continue* statement, programming examples

### **Arrays** **3 Hours**

The meaning of an array, one dimensional and two dimensional arrays, declaration and initialization of arrays, reading, writing and manipulation of above types of arrays, multidimensional arrays, dynamic arrays, programming examples.

### **Character Arrays and Strings** **3 Hours**

Declaring and initialing string variables, reading string from terminal, writing string to screen, arithmetic operations on characters, putting strings together, comparison of two strings, string handling functions, table of strings, other features of strings, programming examples.

### **User Defined Functions** **6 Hours**

Need for user defined functions, a multi function program, elements of User defined functions, defining functions, return values and their types, function calls, function declaration, category of functions, no arguments and no return values, arguments but no return values, arguments with return values, no arguments with return value, functions that return multiple values, nesting of functions, recursion, passing arrays to functions, passing string to functions, programming examples.

### **Structures and Unions** **3 Hours**

Defining a structure, declaring structure variables, accessing structure members, structure initialization, copying and comparing structure variables, operations on individual members, array of structures, structures within structures, structures and functions, Unions, size of structures, bit fields, programming examples

**Pointers****5 Hours**

Understanding pointers, accessing the address space of a variable, declaring and initialization pointer variables, accessing a variable through its pointer, chain of pointers, pointer expressions, pointers and arrays, pointer and character strings, array of pointers, pointer as function arguments, functions returning pointers, pointers to functions, pointers and structures, programming examples

**File Management in C****6 Hours**

Defining and opening a file, closing a file, input/output operations on files, error handling during I/O operations, random access files, command line arguments, programming examples.

**Dynamic Memory Allocation****4 Hours**

Dynamic memory allocation, allocating a block of memory: *malloc*, allocating multiple blocks of memory: *calloc*, releasing the used space: *Free*, altering the size of a block: *realloc*, programming examples

**The Preprocessor****2 Hours**

Introduction, macro substitution, files inclusion, compiler control directives, ANSI additions, programming exercises.

**Text Books:**

1. Balagurusamy: Programming in ANSI C, 4<sup>th</sup> Edition, Tata McGraw Hill, 2008.  
(Chapters: 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13.1-13.6 & 14)
2. V Rajaraman: Computer Programming in C, PHI, 2000.  
(Chapters: 1.1, 1.3, 2.1, 2.3, 3.1, 3.2 and 3.3)

**Reference Books:**

1. Behrouz A Forouzan and Richard F Gilberg: Structured Programming Approach C, 2<sup>nd</sup> Edition, Thomson, 2005.
2. M G Venkateshmurthy: Programming Techniques through C, Pearson Education, 2005.
3. Ivor Horton: Beginning C from Novice to Professional, 4<sup>th</sup> Edition, Springer, 2005.
4. Ashok N Kamthane: Programming with ANSI and Turbo C, Pearson Education, 2002.
5. K R Venugopal, S R Prasad: Mastering C, Tata McGraw Hill, 2006.

## Discrete Mathematics

**Subject Code: 10MCA12**

**Hours/Week: 4**

**Total Hours: 52**

**I.A. Marks: 50**

**Exam Marks: 100**

**Exam Hours: 3**

### **Set Theory**

**6 Hours**

Sets and Subsets, Set Operations and the Laws of Set Theory, Counting and Venn Diagrams, A First Word on Probability, Countable and Uncountable Sets

### **Fundamentals of Logic**

**13 Hours**

Basic Connectives and Truth Tables, Logic Equivalence – The Laws of Logic, Logical Implication – Rules of Inference; The Use of Quantifiers, Quantifiers, Definitions and the Proofs of Theorems

### **Properties of the Integers**

**7 Hours**

Mathematical Induction, The Well Ordering Principle – Mathematical Induction, Recursive Definitions

### **Relations and Functions**

**14 Hours**

Cartesian Products and Relations, Functions – Plain and One-to-One, Onto Functions – Stirling Numbers of the Second Kind, Special Functions, The Pigeon-hole Principle, Function Composition and Inverse Functions; Properties of Relations, Computer Recognition – Zero-One Matrices and Directed Graphs, Partial Orders – Hasse Diagrams, Equivalence Relations and Partitions

### **Groups, Coding Theory and Rings**

**6 Hours**

Definitions, Examples, and Elementary Properties, Homomorphisms, Isomorphisms, and Cyclic Groups, Cosets, and Lagrange's Theorem

Elements of Coding Theory, The Hamming Metric, The Parity Check, and Generator Matrices

### **Group Codes, Rings and Modular Arithmetic**

**6 Hours**

Decoding with Coset Leaders, Hamming Matrices

The Ring Structure – Definition and Examples, Ring Properties and Substructures, The Integers Modulo  $n$

### **Text Book:**

1. Ralph P. Grimaldi: Discrete and Combinatorial Mathematics, 5<sup>th</sup> Edition, Pearson Education, 2004. (Chapter 3.1, 3.2, 3.3, 3.4, Appendix 3, Chapter 2, Chapter 4.1, 4.2, Chapter 5.1 to 5.6, Chapter 7.1 to 7.4, Chapter 16.1, 16.2, 16.3, 16.5 to 16.9, and Chapter 14.1, 14.2, 14.3).

### **Reference Books:**

1. Kenneth H. Rosen: Discrete Mathematics and its Applications, 6<sup>th</sup> Edition, McGraw Hill, 2007.
2. Jayant Ganguly: A Treatise on Discrete Mathematical Structures, Sanguine-Pearson, 2010.
3. D.S. Malik and M.K. Sen: Discrete Mathematical Structures: Theory and Applications, Thomson, 2004.
4. Thomas Koshy: Discrete Mathematics with Applications, Elsevier, 2005, Reprint 2008.

## Fundamentals of Computer Organization

**Subject Code: 10MCA13**

**Hours/Week: 4**

**Total Hours: 52**

**I.A. Marks: 50**

**Exam Marks: 100**

**Exam Hours: 3**

### **Number Systems and Logic Gates**

**7 Hours**

Counting in Decimal and Binary, Place Value, Binary to Decimal Conversion, Decimal to Binary Conversion, Hexadecimal Numbers, Octal Numbers, Bits, Bytes, Nibbles, and Word Size, The AND Gate, The OR gate, The inverter and Buffer, The NAND gate, the NOR Gate, The exclusive OR gate, The Exclusive NOR Gates, The NAND Gate as an universal Gate, Gates with More than two inputs, Using Inverters to convert gates.

### **Combining Logic Gates and Arithmetic Circuits**

**9 Hours**

Constructing Circuits from Boolean Expression, Drawing a circuit from a Maxterm Boolean Expression, Truth Tables and Boolean Expressions, Sample Problem Simplifying Boolean Expression, Karnaugh Maps, Karnaugh Maps with three variables, Karnaugh Maps with four variables, more Karnaugh Maps, using Demorgan's Theorem, Binary Addition, Half Adders, Full Adders, Three Bit Adders, Binary Subtraction, Parallel Subtractors, IC Adders, Binary Multiplication, Binary Multipliers, 2s Complement Notation, Addition and Subtraction, 2s Complement adders/subtractor

### **Basic Structure of Computer**

**6 Hours**

Computer Types, Functional Units, Basic Operational Concepts, Bus structures, Performance, Memory Location and Addresses

### **Machine Instruction and Programmes**

**8 Hours**

Memory Operations, Instructions & Instruction Sequencing, Addressing Modes, Assemble Language, Basic Input/Output Operations

### **Input/Output Organization**

**8 Hours**

Accessing I/O Devices, Interrupts, Direct Memory Accesses, Buses 240, Interface Circuits

### **Memory Systems**

**8 Hours**

Some Basics concepts, Semiconductors RAM Memories, Read-Only Memories, Cache Memories, Virtual Memories

### **Arithmetic Unit**

**6 Hours**

Addition & subtraction of Signed Numbers, Design of Fast adders, Multiplication of Positive Numbers, Signed-Operand Multiplication, Fast Multiplication, Integer division  
Floating-Point Numbers & Operations

### **Text Books:**

1. Tokheim: Digital Electronics Principles and Applications, McGraw Hill, 6<sup>th</sup> Edition, 2004.  
(Chapters: 2.1 to 2.4, 2.6 to 2.8, 3.1 to 3.10, 4.1 to 4.9, 4.16, 10.1 to 10.11)
2. Carl Hamacher, Z Varnesic and S Zaky: Computer Organization, 5<sup>th</sup> Edition, McGraw Hill, 2002.  
(Chapters: 1.1 to 1.4, 1.6, 2.2 to 2.7, 4.1, 4.2, 4.4 to 4.6, 5.1 to 5.5, 5.7, 6.1 to 6.7)

### **Reference Books:**

1. M Morris Mano: Digital Logic and Computer Design, 10<sup>th</sup> Edition, Pearson, 2008.
2. Morris Mano, and Charles R Kime: Logic and Computer Design Fundamentals, 2<sup>nd</sup> Edition, Pearson Education, 2001.

## Introduction to UNIX

**Subject Code: 10MCA14**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Marks: 100**  
**Exam Hours: 3**

### 10 Hours

**General Purpose Utilities:** banner, cal, date, calendar, who, printf, tty, stty, uname, passwd, lock, echo, tput, bc, script, spell and ispell.

**Navigating the File System:** The file, what's in a filename? the parent-child relationship, pwd, the Home directory, Absolute pathnames, using absolute pathnames for a command, cd, mkdir, rmdir, Relative pathnames, The UNIX file system

**Handling Ordinary Files:** cat, cp, rm, mv, more, lp, file, wc, od, split, cmp, comm., diff.

### 6 Hours

**The vi editor:** The Three Modes, Input Mode, Saving, The Repeat Factor, Command Mode, Deletion, Navigation, Pattern Search, Joining lines, Repeating the Last command, undoing the Last Editing Instructions, Search and Replace.

**Basic File Attributes:** ls -l, the -d option, File Permissions, chmod.

**Security and File Permission:** users and groups, security level, changing permission, user masks, changing ownership and group

### 6 Hours

**Shell Programming:** Shell Scripts, read, Command Line Arguments, Exit Status of a Command, The Logical Operators && and ||, exit, if, and case conditions, expr, sleep and wait, while, until, for, \$, @, redirection, The here document, set, trap, Sample Validation and Data Entry Scripts.

**The Process:** process basics, PS, internal and external commands, running jobs in background, nice, at and batch, cron, time commands

**Customizing the Environment:** System Variables, profile, sty, PWD, Aliases, Command History, On-Line Command Editing

### 8 Hours

**More file attributes:** hard link, symbolic link, umask, find

**Simple filters:** PR, head, tail, cut, paste, sort, uniq, tr commands

### 7 Hours

**Filters using Regular Expression and the grep Family:** grep, Regular Expression, egrep, fgrep, a sed instruction, Line Addressing, Inserting and Changing Text, Context addressing, writing selected lines to a file, The -f option, Substitution, Properties of Regular Expressions

### 7 Hours

**Essential Shell Programming:** shell script, read, exit, the if conditional, using Tests and [] to evaluate expression, the case conditional expr, while: looping, for: looping with a list, set and shift, trap, debugging shell scripts with Set - X.

**Essential System Administration:** root, administrator's privileges, startup & Shutdown, managing disk space, cpio, tar

### 6 Hours

**awk-Advanced Filters:** Simple awk Filtering, Splitting a Line in to Fields, printf, the Logical and Relational Operators, Number Processing, Variables, The -f option, The BEGIN and END Positional Parameters, get line, Built-in variables, Arrays, Functions, Interface with the Shell, Control Flow.

**Advanced Shell Programming:** The sh command, export, cd, the Command, expr, Conditional Parameter Substitution, Merging Streams, Shell Functions, eval, Exec Statement.

### 2 Hours

**Advanced vi:** Operators, the ex mode, named buffered , Numbered Buffers, Entering Control Characters, Searching for a Character, Marking Text Customizing, vi

### Text Books:

1. Sumitabha Das: UNIX Concepts and Applications, 4<sup>th</sup> Edition, Tata McGraw Hill, 2006.  
 (Chapters: 1.2, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 21, 20, 21)



2. Behrouz A Forouzan and Richard F Gilberg: UNIX and Shell Programming, Thomson Course Technology, 2005.  
(Chapter: 4)

**Reference Books:**

1. Kenneth Roson et al: UNIX - The Complete Reference, Osborne/McGraw Hill, 1999.
2. Steve Montsugu: Using UNIX, 2<sup>nd</sup> Editon, Prentice Hall India, 1999.
3. M G Venkateshmurthy: UNIX and Shell Programming, Pearson Education, 2005.

## Professional Communication and Ethics

**Subject Code: 10MCA15**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Marks: 100**  
**Exam Hours: 3**

### **Professional / Technical Communication**

#### **Basics of Technical Communication** **5 Hours**

Introduction, Process of Communication, Language as a Tool, Levels of Communication, Levels of Communication, Communication Networks, Importance of Technical Communication

#### **Barriers to Communication** **3 Hours**

Definition of Noise, Classification of Barriers

#### **Technology in Communication** **4 Hours**

Impact of Technology, Software for Creating Messages, Software for Writing Documents, Software for Presenting Documents, Transmitting Documents, Effective use of Available Technology

#### **Active Listening** **3 Hours**

Introduction, Types of Listening, Traits of good Listener, Active versus passive listening, implications of effective listening

#### **Effective Presentation Strategies** **5 Hours**

Introduction, Defining purpose, Analyzing Audience and Locale, Organizing Contents, preparing outline, Visual Aids, Understanding Nuances of Delivery, Kinesics, Proxemics, Paralinguistic's, Chronemics, Sample speech

#### **Group Communication** **5 Hours**

Introduction, Group Discussion, Organizational Group discussion, Group discussion as part of selection process Meetings, conferences.

#### **Research paper, Dissertation and Thesis** **4 Hours**

Introduction, Research paper, Dissertation, Thesis

### **Ethics:**

#### **An Overview of Ethics** **3 Hours**

What are Ethics? Ethics in the Business World, Ethics in Information Technology (IT)

#### **Ethics for IT Professionals and IT users** **4 Hours**

IT professionals, The Ethical behavior of IT professionals, IT Users

#### **Privacy** **3 Hours**

Privacy Protection and the Law, Key Privacy and Anonymity Issues

#### **Software Development** **3 Hours**

Strategies to Engineer Quality s/w, Key Issues

#### **Employer/Employee Issues** **3 Hours**

Use of Nontraditional workers, Whistle Blowing

#### **Case Studies and Discussion** **7 Hours**

### **Text Books:**

1. Meenakshi Raman and Sangeeta Sharma: Technical Communication - Principles and Practices, Oxford University Press, 2004.  
(Chapters: 1, 2, 3, 4, 5, 7, 16)
2. George Reynolds: Ethics in Information Technology, 2<sup>nd</sup> Edition, Thomson Course Technology, 2007.  
(Chapters: 1, 2, 4, 7, 8)

### **Reference Books:**

1. M.Ashraf Rizivi: Effective Technical Communication, Tata McGraw Hill, 2005.

2. Mike W Martin and Ronald Schinzinger: Ethics in Engineering, 3<sup>rd</sup> Edition, Tata McGraw Hill, 2007.

## C Programming Laboratory

**Subject Code: 10MCA16**

**Hours/Week: 3**

**Total Hours: 42**

**I.A. Marks: 50**

**Exam Marks: 50**

**Exam Hours: 3**

### Part A

1. a. Write a program to find the area of a triangle (Given the three sides).  
b. Write a program to find the area of a circle (Given the radius).
2. Write a program to find the Simple interest, given the principle, time and rate of interest with appropriate validations.
3. Write a program to find out whether a given year is a leap year or not.
4. Write a program to find the roots of a quadratic equation with appropriate error messages.
5. Write a program to display the following files of current directory.  
i) .EXE files    ii) .BAT files    iii) .OBJ files    iv) .BAK files.  
By using system DOS command.
6. Write a program to find GCD and LCM of given two numbers.
7. Write a program to find the value of Sin (x) using the series.  
$$\text{Sin}(x) = x - x^3/3! + x5/5! - x7/7! + \dots\dots\dots$$
8. Write a program to print all prime numbers between m and n.
9. Write a program to reverse a number and check whether it is palindrome or Not.
10. Write a program to generate and print first n Fibonacci numbers using function.
11. Write a program to find a factorial of a given number using recursive function.
12. Write a program to convert UPPERCASE alphabets to LOWERCASE alphabets in a given string and vice-versa.
13. Write a program to read two strings and concatenate them (without using library functions).
14. Write a program to read a sentence and count the number of vowels and constants.

### PART - B

1. Write a program to read N integers (zero, + ve and -ve) into an array and find sum of positive numbers, sum of negative numbers and average of all input numbers.
2. Write a program to input N real numbers and to find the mean, variance and standard deviation, where,

$$\text{Mean} = \Sigma xi / N$$

$$\text{Variance} = \frac{\Sigma(x_i - \text{mean})^2}{N}$$

$$\text{Deviation} = \sqrt{\text{variance}} \quad \text{and } 0 \leq i < n$$

3. Write a program to input N numbers (integers or real) and store them in an array. Conduct a Linear search for a given key number and report success or failure in the form of a suitable message.

\*\*\*

4. Write a program to sort N numbers in ascending or descending order using bubble sort.
5. Write a program to accept N numbers sorted in ascending order and search for a given number using binary search. Report success or failure in the form of suitable messages.
6. Write a program to read two matrices A and B of size M x N and perform product of two given matrices.
7. Write a program to list the names of students who have scored more than 60% of total marks in three subjects using structure variables.
8. Write a program to compute the sum of two complex numbers – passing a structure to a function.
9. Define a book structure having title of the book, ISBN, author, price and month and year of publication as its members. Use a substructure to store the month and year of Publication information. Develop a program to accept a date (in the form of month and year) and list out all the book titles (along with price and ISBN) published during that date.
10. Define a student structure having the name, USN (university seat number), marks in five subjects, total and percentage of marks as its members. Marks of all the subjects are to be stored in an array. Develop a program to list the names of all the students who have failed.
11. Write a program to read N integers and store them in an array, find the sum of all these elements using pointer. Output the given array and the computed sum with suitable heading.
12. Write a program to read and write to a file.
13. Write a program to Create and count number of characters in a file.
14. Write a program to handle files with mixed data type.

**Note: Students are required to execute one question from Part A and one from Part B**

## UNIX Programming Laboratory

**Subject Code: 10MCA17**  
**Hours/Week: 3**  
**Total Hours: 42**

**I.A. Marks: 50**  
**Exam Marks: 50**  
**Exam Hours: 3**

1. a. Write a non-recursive shell script which accepts any number of arguments and prints them in the reverse order (For example, if the script is named rags, then executing args A B C should produce C B A on the standard output).  
 b. Write a shell script that accepts two file names as arguments, checks if the permissions for these files are identical and if the permissions are identical, output common permissions and otherwise output each file name followed by its permissions.
2. a. Write a shell script that takes a valid directory name as an argument and recursively descend all the subdirectories, finds the maximum length of any file in that hierarchy and writes this maximum value to the standard output.  
 b. Write a shell script that accepts a path name and creates all the components in that path name as directories. For example, if the script is named mpc, then the command mpc a/b/c/d should create directories a, a/b, a/b/c, a/b/c/d.
3. a. Write a shell script which accepts valid log-in names as arguments and prints their corresponding home directories, if no arguments are specified, print a suitable error message.  
 b. Write shell script to implement terminal locking (similar to the lock command). It should prompt the user for a password. After accepting the password entered by the user, it must prompt again for the matching password as confirmation and if match occurs, it must lock the keyword until a matching password is entered again by the user, Note that the script must be written to disregard BREAK, control-D. No time limit need be implemented for the lock duration.
4. a. Create a script file called file-properties that reads a file name entered and outputs its properties.  
 b. Write a shell script that accept one or more filenames as argument and convert all of them to uppercase, provided they exist in current directory.
5. a. Write a shell script that displays all the links to a file specified as the first argument to the script. The second argument, which is optional, can be used to specify in which the search is to begin. If this second argument is not present, the search is to begin in current working directory. In either case, the starting directory as well as all its subdirectories at all levels must be searched. The script need not include any error checking.  
 b. Write a shell script that accepts as filename as argument and display its creation time if file exist and if it does not send output error message.
6. a. Write a shell script to display the calendar for current month with current date replaced by \* or \*\* depending on whether the date has one digit or two digits.  
 b. Write a shell script to find smallest of three numbers that are read from keyboard.
7. a. Write a shell script using expr command to read in a string and display a suitable message if it does not have at least 10 characters.  
 b. Write a shell script to compute the sum of number passed to it as argument on command line and display the result.
8. a. Write a shell script that compute gross salary of an employee, accordingly to rule given below.  
     If basic salary is < 15000 then HRA=10% of basic & DA=90% of basic.  
     If basic salary is >=15000 then HRA=500 of basic & DA=98% of basic.  
 b. Write a shell script that delete all lines containing a specific word in one or more file supplied as argument to it.
9. a. Write a shell script that gets executed displays the message either “Good Morning” or “Good Afternoon” or “Good Evening” depending upon time at which the user logs in.  
 b. Write a shell script that accept a list of filenames as its argument, count and report occurrence of each word that is present in the first argument file on other argument files.
10. a. Write a shell script that determine the period for which a specified user is working on system.

\*\*\*

- b. Write a shell script that reports the logging in of a specified user within one minute after he/she log in. The script automatically terminate if specified user does not log in during a specified period of time.
11. a. Write a shell script that accepts two integers as its argument and compute the value of first number raised to the power of second number.  
b. Write a shell script that accept the file name, starting and ending line number as an argument and display all the lines between the given line number.
12. a. Write a shell script that folds long lines into 40 columns. Thus any line that exceeds 40 characters must be broken after 40<sup>th</sup>, a “\” is to be appended as the indication of folding and the processing is to be continued with the residue. The input is to be supplied through a text file created by the user.  
b. Write an awk script that accepts date argument in the form of mm-dd-yy and displays it in the form if day, month, and year. The script should check the validity of the argument and in the case of error, display a suitable message.
13. a. Write an awk script to delete duplicated line from a text file. The order of the original lines must remain unchanged.  
b. Write an awk script to find out total number of books sold in each discipline as well as total book sold using associate array down table as given below.
- |                       |     |
|-----------------------|-----|
| i. Electrical         | 34  |
| ii. Mechanical        | 67  |
| iii. Electrical       | 80  |
| iv. Computer Science  | 43  |
| v. Mechanical         | 65  |
| vi. Civil             | 198 |
| vii. Computer Science | 64  |
14. Write an awk script to compute gross salary of an employee accordingly to rule given below.  
If basic salary is < 10000 then HRA=15% of basic & DA=45% of basic.  
If basic salary is >=10000 then HRA=20% of basic & DA=50% of basic.

**Note: In the examination each student picks one question from a lot of *all* the 14 questions.**

## IT and Digital Electronics Laboratory

**Subject Code: 10MCA18**  
**Hours/Week: 3**  
**Total Hours: 42**

**I.A. Marks: 50**  
**Exam Marks: 50**  
**Exam Hours: 3**

### PART - A

#### Exercises on Office Tools

**21 Hours**

**Note: Exercises on the following topics to be conducted in the laboratory using any of the very widely used current software packages**

#### **Word Processing**

Preparation of business letters, project proposals, etc. Experimenting with all manipulation facilities like bold facing, Italicizing, Alignment cut and paste, spell checking including headers and footers, etc., Use of Mail-merge facility.

#### **Computer Presentation**

Learning commands, Preparation of slides, inserting texts, graphs, etc., color changing, automatic presentation of slides, changing time settings, object linking and embedding, etc

#### **Database Management System**

Creation of Database -Tables, fields, keys, Basic queries - insert, delete, update, design of data entry forms

### PART - B

#### Exercises on Digital Electronics

**21 Hours**

1. Realization of X-OR and X-NOR operations using basic gates.
2. Design of Half Adder and Full adder using 2-input NAND gates.
3. Verify addition of two binary numbers using 4-bit adder chip.
4. Design and implement Excess-3 to BCD code converter using 4-bit adder chip.
5. Construct and Implement the Boolean expression using AND,OR and NOT gates  

$$(A+B).(C+D).(A+C) = Y$$
6. Construct and Implement the Boolean expression using only NAND gates  

$$A B C D + A B' C D + A B C D + A B C D + A B C' D + A B C D = Y$$

**Note: Students are required to execute one question from Part A and one from Part B**



## II SEMESTER

### Business Data Processing with COBOL

**Subject Code: 10MCA21**

**Hours/Week: 4**

**Total Hours: 52**

**IA Marks: 50**

**Exam Marks: 100**

**Exam Hours: 3**

#### **Introduction**

**3 Hours**

Coding Format for COBOL Programs, Structure of a COBOL Program, Character Set, COBOL words, Data Names and Identifiers, Literal, Figurative Constants, Continuation of lines and notations

#### **Identification, Environment and Data Division**

**5 Hours**

General Formats, Configuration Section, Input-Output Section, Level Structure, Data Description entries, Picture Clause, Value Clause, File Section, Working-storage Section, Editing characters of different data and examples, Special-names Paragraph, Classes and Categories of Data

#### **Procedure Division and Basic Verbs**

**6 Hours**

Structure of the Procedure Division, Data Movement Verb and other options of Move Statements, Arithmetic Verbs, Sequence Control Verbs, Input and Output Verbs, Conditional Verb: simple IF, categories of COBOL statements.

#### **Different Types of Clauses and other Verbs**

**4 Hours**

Usage Clause, Synchronized Clause, Justified Clause, Redefines Clause and Renames Clause, Qualification of Data Names, Sign Clause, Elementary and Group Moves, Corresponding Option: Move Corresponding, Add Corresponding and Subtract Corresponding, Rounded Option, On Size Error Option, Compute Verb

#### **Computation and Decision Making**

**8 Hours**

Interactive Processing using Screen Sections, Intrinsic Functions, Conditions: Relational Condition, Class Condition, Condition Name Condition, Negated Simple Condition, Compound Condition, Sign Condition, If Statements, Alter Statement, Perform Statements, Exit Statement

#### **Array Processing and Table Handling**

**6 Hours**

Occurs Clause and Subscripting, assigning values to table elements, Multi Dimensional Tables, Perform and Table Handling, Indexed Tables and Indexing, Set Verb, Search Verb, Occurs Depending Clause, Index data item

#### **Sequential File Processing, Sorting and Merging of Files**

**10 Hours**

File Characteristics, File Control Entries, File Description, Statements for Sequential Files, Sequential File with Variable Length Records, I-O Control Paragraph, Simple Sort Verb, File updation, Variation of updation, simple Merge Verb, Input and Output Procedure in Sort Statement, Merge Verb with Output Procedure

#### **Indexed and Relative File Processing, Report Writer, and Subroutines**

**10 Hours**

File Control Paragraph for Relative Files, Procedure Division Statements for Relative Files, Indexed Sequential Files, General Format of a Report, File Section, Report Clause, Outline of Report Section, Report Section - Report Description Entry, Report Group Description, Procedure Division Statements, Sample Program, Structure of a COBOL Subroutine, Calling of a Subroutine, State of Subroutine and Cancel Statement, Advantages and Disadvantages of COBOL Subroutines

#### **Text Books:**

1. M K Roy and D Ghosh Dastidar: COBOL Programming Including MS COBOL and COBOL-85, 2<sup>nd</sup> Edition, Tata McGraw Hill, 2001.  
(Chapters: 3.1 to 3.10, 4.1, 4.2, 5.1 to 5.7, 6.1 to 6.7, 8.1 to 8.7, 9.1 to 9.5, 10.1 to 10.6, 11.1 to 11.10, 13.1 to 13.4, 13.6, 14.1 to 14.7, 16.1, 16.2, 18.1 to 18.7, 19.1 to 19.6)
2. Nancy Stern and Robert A Stern: Structured COBOL Programming, 11<sup>th</sup> Edition, Willey Publications, 2006.  
(Chapters: 1, 6, and 7)

#### **Reference Books:**

1. Shelly Cashman Foreman: Structural COBOL Programming, 2<sup>nd</sup> Edition, Thomson Course Technology.

2. David M. Collopy: Introduction to COBOL, A Guide to Modular Structured Programming, Pearson Education, 2000.

## Object Oriented Programming with C++

**Subject Code: 10MCA21**

**Hours/Week: 4**

**Total Hours: 52**

**IA Marks: 50**

**Exam Marks: 100**

**Exam Hours: 3**

### **Introduction**

**6 Hours**

Overview of C++, Sample C++ program, Different data types, operators, expressions, and statements, arrays and strings, pointers & user-defined types

Function Components, argument passing, inline functions, function overloading, recursive functions

### **Classes & Objects**

**14 Hours**

Class Specification, Class Objects, Scope resolution operator, Access members, Defining member functions, Data hiding, Constructors, Destructors, Parameterized constructors, Static data members, Functions;

Friend functions, Passing objects as arguments, Returning objects, Arrays of objects, Dynamic objects, Pointers to objects, Copy constructors, Generic functions and classes, Applications

Operator overloading using friend functions such as +, -, pre-increment, post-increment, [ ] etc., overloading <<, >>.

### **Inheritance**

**12 Hours**

Base Class, Inheritance and protected members, Protected base class inheritance, Inheriting multiple base classes; Constructors, Destructors and Inheritance, Passing parameters to base class constructors, Granting access, Virtual base classes

### **Virtual functions, Polymorphism**

**7 Hours**

Virtual function, Calling a Virtual function through a base class reference, Virtual attribute is inherited, Virtual functions are hierarchical, Pure virtual functions, Abstract classes, Using virtual functions, Early and late binding.

### **I/O System Basics, File I/O**

**6 Hours**

C++ stream classes, Formatted I/O, I/O manipulators, fstream and the File classes, File operations

### **Exception Handling, STL**

**7 Hours**

Exception handling fundamentals, Exception handling options

STL: An overview, containers, vectors, lists, maps.

### **Text Books:**

1. Herbert Schildt: The Complete Reference C++, 4<sup>th</sup> Edition, Tata McGraw Hill, 2003.

### **Reference Books:**

1. Stanley B.Lippmann, Josee Lajore: C++ Primer, 4<sup>th</sup> Edition, Addison Wesley, 2005.
2. Paul J Deitel, Harvey M Deitel: C++ for Programmers, Pearson Education, 2009.
3. K R Venugopal, Rajkumar Buyya, T Ravi Shankar: Mastering C++, Tata McGraw Hill, 1999.

## Data Structures using C

**Subject Code: 10MCA23**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A Marks: 50**  
**Exam Marks: 100**  
**Exam Hours: 3**

**BASIC CONCEPTS** **8 Hours**  
 Pointers and Dynamic Memory Allocation, Algorithm Specification, Data Abstraction, Performance Analysis, Performance Measurement

**ARRAYS and STRUCTURES** **6 Hours**  
 Arrays, Dynamically Allocated Arrays, Structures and Unions, Polynomials, Sparse Matrices, Representation of Multidimensional Arrays

**STACKS AND QUEUES** **6 Hours**  
 Stacks, Stacks Using Dynamic Arrays, Queues, Circular Queues Using Dynamic Arrays, Evaluation of Expressions, Multiple Stacks and Queues.

**LINKED LISTS** **6 Hours**  
 Singly Linked lists and Chains, Representing Chains in C, Linked Stacks and Queues, Polynomials, Additional List operations, Sparse Matrices, Doubly Linked Lists

**TREES, Graphs** **12 Hours**  
 Introduction, Binary Trees, Binary Tree Traversals, Threaded Binary Trees, Heaps. Binary Search Trees, Selection Trees, Forests, Representation of Disjoint Sets, Counting Binary Trees, The Graph Abstract Data Type.

**PRIORITY QUEUES** **6 Hours**  
 Single- and Double-Ended Priority Queues, Leftist Trees, Binomial Heaps, Fibonacci Heaps, Pairing Heaps.

**EFFICIENT BINARY SEARCH TREES** **8 Hours**  
 Optimal Binary Search Trees, AVL Trees, Red-Black Trees, Splay Trees.

**Text Book:**

1. Horowitz, Sahni, Anderson-Freed: Fundamentals of Data Structures in C, 2<sup>nd</sup> Edition, University Press, 2007.  
 (Chapters 1, 2.1 to 2.6, 3, 4, 5.1 to 5.3, 5.5 to 5.11, 6.1, 9.1 to 9.5, 10)

**Reference Books:**

1. Debasis Samanta: Classic Data Structures, 2<sup>nd</sup> Edition, PHI, 2009.
2. Richard F. Gilberg and Behrouz A. Forouzan: Data Structures A Pseudocode Approach with C, Thomson, 2005.
3. Robert Kruse & Bruce Leung: Data Structures & Program Design in C, Pearson Education, 2007.

## Management Information System

**Subject Code: 10MCA24**

**Hours/Week: 4**

**Total Hours: 52**

**I.A. Marks: 50**

**Exam Marks: 100**

**Exam Hours: 3**

### **Systems Engineering**

**4 Hours**

System concepts, system control, types of systems, handling system complexity, Classes of systems, General model of MIS, Need for system analysis, System analysis for existing system & new requirement, system development model, MIS & system analysis

### **Information and Knowledge**

**4 Hours**

Information concepts, classification of information, methods of data and information collection, value of information, information: A quality product, General model of a human as information processor, Knowledge,

### **Introduction of MIS**

**4 Hours**

MIS: Concept, Definition, Role of the MIS, Impact of MIS, MIS and the user, Management as a control system, MIS support to the management, Management effectiveness and MIS, Organization as system. MIS: organization effectiveness

### **Strategic Management of Business**

**3 Hours**

Concept of corporate planning, Essentiality of strategic planning, Development of the business strategies, Type of strategies, short-range planning, tools of planning, MIS: strategic business planning

### **Development of MIS**

**4 Hours**

Development of long range plans of the MIS, Ascertaining the class of information, Determining the information requirement, Development and implementation of the MIS, Management of information quality in the MIS, Organization for development of MIS, MIS development process model

### **Developing Business/IT Strategies/IT Solutions**

**5 Hours**

Planning fundamentals (real world cases), Organizational planning, planning for competitive advantage,(SWOT Analysis), Business models and planning. Business/IT planning, identifying business/IT strategies, Implementation Challenges, Change management., Developing business systems, (real world case), SDLC, prototyping, System development process, implementing business system

### **Business Process Re-Engineering**

**2 Hours**

Introduction, Business process, process model of the organization, value stream model of the organization, what delay the business process, relevance of information technology, MIS and BPR

### **Technology of Information System**

**4 Hours**

Introduction, Data processing, Transaction processing, Application processing, information system processing, TQM of information systems, Human factors & user interface, Strategic nature of IT decision, MIS choice of information technology

### **Decision Making and DSS**

**3 Hours**

Decision making concepts; decision making process, decision-making by analytical modeling, Behavioral concepts in decision making, organizational decision-making, Decision structure, DSS components, Management reporting alternatives.

### **Data resource Management**

**3 Hours**

Managing data sources, Foundation concepts of data, types of databases, traditional file processing, DBMS approach, Database structure, Database development

### **Electronic Business systems**

**4 Hours**

Enterprise business system – Introduction, cross-functional enterprise applications, real world case, Functional business system, - Introduction, marketing systems, sales force automation, CIM, HRM, online accounting system, Customer relationship management, ERP, Supply chain management (real world cases for the above)

### **Enterprise Business Systems**

**6 Hours**

Electronic commerce fundamentals, e-Commerce applications and Issues, (real world cases)

**Client Sever Architecture and E-business Technology****6 Hours**

Client server architecture, implementation strategies, Introduction to E-business, model of E-business, internet and World Wide Web, Intranet/Extranet, Electronic, Impact of Web on Strategic management, Web enabled business management, MIS in Web environment.

**Text Books:**

1. Waman S Jhawadekar: Management Information System, 3<sup>rd</sup> Edition, Tata McGraw Hill, .  
(Chapters: 1, 3, 5, 6, 7,8,10, 11, 16.1 to 16.7, 16.10 to 16.12, 18.7, 18.8, 20)
2. James A O'Brien and George M Marakas: Management Information System, 7<sup>th</sup> Edition, Tata McGraw Hill, 2006  
(Chapters: 1, 5, 7, 8, 9, 10, 11)

**Reference Books:**

1. Ralph M Stair and George W Reynolds: Principles of Information Systems, 7<sup>th</sup> Edition, Thomson,2010 .
2. Steven Alter: Information Systems - The Foundation of E-Business, 4<sup>th</sup> Edition, Pearson Education, 2001
3. Mahadeo Jaiswal and Monika Mital: Management Information System, , Oxford University Press.
4. Effy Oz: Management Information Systems, 5<sup>th</sup> Edition, Thomson Course Technology,2006.

## Operations Research

**Subject Code: 10MCA25**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Marks: 100**  
**Exam Hours: 3**

### **Introduction and Overview of the OR Modeling Approach**

**3 Hours**

The origin of OR, the nature of OR, the impact of OR, defining the problem and gathering data, Formulating a mathematical model, deriving solutions from the model, testing the model, preparing to apply the model, implementation .

### **Introduction to Linear Programming**

**6 Hours**

Formulation of linear programming problem (LPP), examples, Graphical solution, the LP Model, Special cases of Graphical method, assumptions of Linear Programming (LP), additional example

### **Solving LPP - the Simplex Method**

**12 Hours**

The essence of the simplex method, setting up the simplex method, algebra of the simplex method, the simplex method in tabular form, special cases in the simplex method, tie breaking in the simplex method, adopting to other model forms (Two Phase method, Big-M method), post optimality analysis.

### **Theory of the Simplex Method**

**4 Hours**

Foundation of the simplex method, the revised simplex method, a fundamental insight

### **Duality Theory and Sensitivity Analysis**

**9 Hours**

The essence of duality theory, economic interpretation of duality, primal dual relationship, adapting to other primal forms, the role of duality in sensitive analysis, the dual simplex method

### **Transportation and Assignment Problems**

**6 Hours**

The transportation problem, a stream line simplex method for the transportation problem, the assignment problem, a special algorithm for the assignment problem

### **Metaheuristics**

**6 Hours**

The nature of Metaheuristics, Tabu Search, Simulated Annealing, Generating Algorithms

### **Game Theory**

**6 Hours**

The formulation of two persons, zero sum games, solving simple games- a prototype example, games with mixed strategies, graphical solution procedure, solving by linear programming, extensions

### **Text Books:**

1. Frederick S.Hillier & Gerald J.Lieberman: Introduction to Operations Research, 8<sup>th</sup> Edition, Tata McGraw Hill, 2006.  
(Chapters 1.1, 1.2, 1.3, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 3.2, 3.3, 3.4, 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, 5.1, 5.2, 5.3, 6.1, 6.2, 6.3,6.4, 6.5, 6.6, 6.7, 7.1, 8.1, 8.2, 8.3, 8.4, 13.1, 13.2, 13.3, 13.4, 14.1, 14.2, 14.3, 14.4, 14.5, 14.6)

### **Reference Books:**

1. Wayne L. Winston: Operations Research Applications and Algorithms, 4<sup>th</sup> Edition, Thomson Course Technology, 2003.
2. Hamdy A Taha: Operations Research - An Introduction, 8<sup>th</sup> Edition, Pearson Education, 2007.
3. Theory and Problems of Operations Research, Richard Bronson and Govindasami Naadimuthu, Schaum's Outline, Tata McGraw Hill, 2<sup>nd</sup> Edition, 1997.

## COBOL Programming Lab

**Subject Code: 10MCA26**

**Hours/Week: 3**

**Total Hours: 42**

**I.A Marks: 50**

**Exam Marks: 50**

**Exam Hours: 3**

1. Write a program for the following:
  - a. Evaluate the following equation using COMPUTE verb.  
 $Z=(A - B) * C$
  - b. To Find the sum of individual digits in the given 'n' digit decimal number.
2. Write a program for the following:
  - a. Generating odd and even numbers between the given limits by using PERFORM statement.
  - b. Sorting numeric numbers.
3. Write a program to search a record in the file without using SEARCH verb.
4. By using table handling concepts, write a program for
  - a. Sorting alphanumeric data.
  - b. Searching a numeric number using SEARCH verb.
5. By using table handling concepts, write a program for matrix addition and matrix subtraction.
6. By using table handling concepts, write a program for matrix multiplication.
7. Write a program for the following:
  - a. To sort records in the file using SORT verb.
  - b. To merge files using MERGE verb.
8. Write a program to read employee file as sequential file and calculate Gross Pay and Net Pay for employees in the organization.
9. Write a program to store student's details with results in a file with sequential file handling techniques.
10. Using the index file organization, write a program to store employee details.
11. Using the index file organization, write a program to update the employee salary (increase DA of 3%).
12. Write a program for the following:
  - a. To delete a record in the index file.
  - b. To search a record in the index file.
13. Write a program to calculate Net Pay and Gross Pay by reading relative file containing fields Emp\_no, Emp\_name, Basic\_pay. Store the output in relative file.
14. Write a program to find simple interest and compound interest using subroutine concept.
15. Write a program to generate a sales report.

**Note: In the examination each student picks one question from a lot of *all* the 15 questions.**



## Data Structures using C Laboratory

**Subject Code: 10MCA27**

**Hours/Week: 3**

**Total Hours: 42**

**I.A Marks: 50**

**Exam Marks: 50**

**Exam Hours: 3**

1. Write a C Program to construct a stack of integers and to perform the following operations on it:
  - a. Push
  - b. Pop
  - c. DisplayThe program should print appropriate messages for stack overflow, stack underflow, and stack empty.
2. Write a C Program to convert and print a given valid parenthesized infix arithmetic expression to postfix expression. The expression consists of single character operands and the binary operators + (plus), - (minus), \* (multiply) and / (divide).
3. Write a C Program to evaluate a valid suffix/postfix expression using stack. Assume that the suffix/postfix expression is read as a single line consisting of non-negative single digit operands and binary arithmetic operators. The arithmetic operators are + (add), - (subtract), \* (multiply) and / (divide).
4. Write a C program using recursive function for the following:
  - a. To calculate GCD and LCM of 2 integer numbers.
  - b. To solve Towers of Hanoi problem.
  - c. To search an element in a list using binary search
5. Write a C Program to simulate the working of a queue of integers using an array. Provide the following operations:
  - a. Insert
  - b. Delete
  - c. Display
6. Write a C Program to simulate the working of a circular queue of integers using an array. Provide the following operations:
  - a. Insert
  - b. Delete
  - c. Display
7. Write a program to design a priority queue which is maintained as a set of queues (assume a maximum of 3 queues). The elements are inserted based upon the given priority. The deletion of an element is to be done starting from the 1<sup>st</sup> queue, if it is not empty. If it is empty, the elements from the 2<sup>nd</sup> queue will be deleted and so on.
8. Write a C Program using dynamic variables and pointers, to construct a singly linked list consisting of the following information in each node: student id (integer), student name (character string) and semester (integer). The operations to be supported are:
  - a. The insertion operation
    - i. At the front of a list
    - ii. At the back of the list
    - iii. At any position in the list
  - b. Deleting a node based on student id. If the specified node is not present in the list an error message should be displayed. Both the options should be demonstrated.
  - c. Searching a node based on student id and update the information content. If the specified node is not present in the list an error message should be displayed. Both situations should be displayed.
  - d. Displaying all the nodes in the list.
9. Write a C Program using dynamic variables and pointers, to construct an ordered (ascending) singly linked list based on the rank of the student, where each node consists of the following information : student id (integer), student name (character string) and rank (integer).

\*\*\*

10. Write a C Program using dynamic variables and pointers to construct a stack of integers using singly linked list and to perform the following operations:
  - a. Push
  - b. Pop
  - c. DisplayThe program should print appropriate messages for stack overflow and stack empty.
11. Write a C Program to support the following operations on a doubly linked list where each node consists of integers:
  - a. Create a doubly linked list by adding each node at the front.
  - b. Insert a new node to the left of the node whose key value is read as an input.
  - c. Delete the node of a given data, if it is found, otherwise display appropriate message.
  - d. Display the contents of the list.(Note: Only either (a, b and d) or (a, c and d) may be asked in the examination)
12. Write a C Program
  - a. To construct a binary search tree of integers.
  - b. To traverse the tree using all the methods i.e., inorder, preorder and postorder.
  - c. To display the elements in the tree.
13. Write a C Programs for searching an element on a given list of integers using the
  - a. Binary Search.
  - b. Linear search
14. Write a C program to sort a list of N integers using the quick sort algorithm.
15. Write a C program to traverse the nodes in a graph using Breadth First Search.

**Note: In the examination each student picks one question from a lot of *all* the 15 questions.**

## Object Oriented Programming with C++ Laboratory

**Subject Code: 10MCA27**

**Hours/Week: 3**

**Total Hours : 42**

**I.A Marks: 50**

**Exam Marks: 50**

**Exam Hours: 3**

1. Given that an EMPLOYEE class contains the following members:
  - a. Data Members: Employee\_Number, Employee\_Name, Basic, DA, IT, Net\_Sal
  - b. Member Functions: to read data, to calculate Net\_Sal and to print data members
2. Write a C++ program to read data on N employees and compute the Net\_Sal of each employee (DA = 52% of Basic and Income Tax = 30% of the gross salary)
3. Define a STUDENT class with USN, Name, and Marks in 3 tests of a subject. Declare an array of 10 STUDENT objects. Using appropriate functions, find the average of the two better marks for each student. Print the USN, Name and the average marks of all the students.
4. Write a C++ program to create a class called COMPLEX and implement the following overloading functions ADD that return a complex number:
  - a. ADD(a, s2) – where 'a' is an integer (real part) and s2 is a complex number
  - b. ADD(s1, s2) – where s1 and s2 are complex numbers
2. Write a C++ program to create a class called LIST (linked list) with member functions to insert an element at the front as well as to delete an element from the front of the list. Demonstrate all the functions after creating a list object.
3. Write a C++ program to create a template function for Quicksort and demonstrate sorting of integers and doubles.
4. Write a C++ program to create a class called STACK using an array of integers. Implement the following operations by overloading the operators '+' and '-':
  - a.  $s1 = s1 + \text{element}$ ; where s1 is an object of the class STACK and element is an integer to be pushed on the top of the stack
  - b.  $s1 = s1 -$ ; where s1 is an object of the class STACK. '-' operator pops the element.
 Handle the STACK empty and full conditions. Also display the contents of the stack after each operation, by overloading the << operator.
5. Write a C++ program to create a class called DATE. Accept two valid dates in the form dd/mm/yy. Implement the following operations by overloading the operators '+' and '-'. After every operation display the results by overloading the operator <<.
  - a.  $\text{no\_of\_days} = d1 - d2$ ; where d1 and d2 are DATE objects, and no\_of\_days is an integer
  - b.  $d2 = d1 + \text{no\_of\_days}$ ; where d1 is a DATE object and no\_of\_days is an integer
6. Create a class called MATRIX using two-dimensional array of integers. Implement the following operations by overloading the operator ++ which checks the compatibility of two matrices to be added and subtracted. Perform the addition and subtraction by overloading the + and - operators respectively. Display the results by overloading the operator <<.
 

If (m1==m2) then m3 = m1+m2 and m4 = m1-m2 else display error.
7. Write a C++ program to create a class called OCTAL which has the characteristics of an octal number. Implement the following operations by writing an appropriate constructor and an overloaded operator +.
  - a. OCTAL h = x; where x is an integer.
  - b.  $\text{int } y = h + k$ ; where h is an OCTAL object and k is an integer
 Display the OCTAL result by overloading the operator <<. Also display the values of h and y.
8. Write a C++ program to create a class called QUEUE with member functions to add an element and to delete an element from the queue. Using the member functions, implement a queue of integers and double. Demonstrate the operations by displaying the contents of the queue after every operation.
9. Write a C++ program to create a class called DLIST (doubly Linked List) with member functions to insert a node at a specified position and delete a node from a specified position of the list. Demonstrate the operations by displaying the content of the list after every operation.

10. Write a C++ program to create a class called STUDENT with data members USN, Name and Age. Using inheritance, create the classes UGSTUDENT and PGSTUDENT having fields as Semester, Fees and Stipend. Enter the data for at least 5 students. Find the semester-wise average age for all UG and PG students separately.
11. Write a C++ program to create a class called STRING and implement the following operations. Display the results after every operation by overloading the operator <<.
  - a. STRING s1 = "VTU"
  - b. STRING s2 = "BELGAUM"
  - c. STRING s3 = s1 + s2 (Use copy constructor)
12. Write a C++ program to create a class called BIN\_TREE (Binary Tree) with member functions to perform in-order, preorder and post-order traversals. Create a BIN\_TREE object and demonstrate the traversals.
13. Write a C++ program to create a class called EXPRESSION. Using appropriate member functions convert a given valid Infix expression into postfix form. Display the infix and postfix expressions.

**Note: In the examination each student picks one question from a lot of *all* the 13 questions.**

## III SEMESTER

### Systems Software

**Subject Code: 10MCA31**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

#### **Machine Architecture**

**6 Hours**

Introduction, System Software and Machine Architecture, Simplified Instructional Computer (SIC) - SIC Machine Architecture, SIC/XE Machine Architecture, SIC Programming Examples.

#### **Assemblers**

**12 Hours**

Basic Assembler Function - A Simple SIC Assembler, Assembler Algorithm and Data Structures, Machine Dependent Assembler Features - Instruction Formats & Addressing Modes, Program Relocation.

Machine Independent Assembler Features – Literals, Symbol-Definition Statements, Expression, Program Blocks, Control Sections and Programming Linking, Assembler Design Operations - One-Pass Assembler, Multi-Pass Assembler, Implementation Examples - MASM Assembler.

#### **Loaders and Linkers**

**8 Hours**

Basic Loader Functions - Design of an Absolute Loader, A Simple Bootstrap Loader, Machine-Dependent Loader Features – Relocation, Program Linking, Algorithm and Data Structures for a Linking Loader; Machine-Independent Loader Features - Automatic Library Search, Loader Options, Loader Design Options - Linkage Editor, Dynamic Linkage, Bootstrap Loaders, Implementation Examples - MS-DOS Linker.

#### **Editors And Debugging Systems**

**6 Hours**

Text Editors - Overview of Editing Process, User Interface, Editor Structure, Interactive Debugging Systems - Debugging Functions and Capabilities, Relationship With Other Parts Of The System, User-Interface Criteria

#### **Macro Processor**

**8 Hours**

Basic Macro Processor Functions - Macro Definitions and Expansion, Macro Processor Algorithm and Data Structures, Machine-Independent Macro Processor Features - Concatenation of Macro Parameters, Generation of Unique Labels, Conditional Macro Expansion, Keyword Macro Parameters, Macro Processor Design Options - Recursive Macro Expansion, General-Purpose Macro Processors, Macro Processing Within Language Translators, Implementation Examples - MASM Macro Processor, ANSI C Macro Processor.

#### **Lex and Yacc**

**12 Hours**

Lex and Yacc - The Simplest Lex Program, Recognizing Words With LEX, Symbol Tables, Grammars, Parser-Lexer Communication, The Parts of Speech Lexer, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Parsing a Command Line.

Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, Symbol Values and Actions, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity, Variables and Typed Tokens.

#### **Text Books:**

1. Leland.L.Beck: System Software, 3<sup>rd</sup> Edition, Addison-Wesley, 1997.  
(Chapters 1.1 to 1.3, 2 (except 2.5.2 and 2.5.3), 3 (except 3.5.2 and 3.5.3), 4 (except 4.4.3))
2. John.R.Levine, Tony Mason and Doug Brown: Lex and Yacc, O'Reilly, SPD, 1998.  
(Chapters 1, 2 (Page 2-42), 3 (Page 51-65))

#### **Reference Books:**

1. D.M.Dhamdhare: System Programming and Operating Systems, 2<sup>nd</sup> Edition, Tata McGraw - Hill, 1999.

## Computer Networks

**Subject Code: 10MCA32**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

<b>Foundation</b> Building a Network; Applications; Requirements; Network Architecture; Implementing Network software; Performance	<b>8 Hours</b>
<b>Direct Link Networks</b> Physically connecting hosts; Hardware building blocks; Encoding; Framing; Error detection; Reliable transmission; Ethernet (802.3); Ring; (802.5, FDDI, 802.17); Wireless (802.15.1, 802.11, 802.16, Cell Phone Technologies).	<b>12 Hours</b>
<b>Packet Switching</b> Switching and forwarding; Bridges and LAN Switches	<b>7 Hours</b>
<b>Internetworking</b> Simple internetworking (IP); Routing; Global Internet	<b>12 Hours</b>
<b>End –to-End Protocols; Resource Allocation Issues</b> Simple demultiplexer (UDP); Reliable byte stream (TCP); Issues in resource allocation	<b>7 Hours</b>
<b>Applications</b> Application-Layer overview; Domain Name System; Remote Login protocols; Electronic mail; File transfer and FTP; World Wide Web and HTTP; Overview of VoIP telephony; VoIP signaling protocols.	<b>6 Hours</b>

### Text Books:

1. Larry L. Peterson and Bruce S. David: Computer Networks – A Systems Approach, 4<sup>th</sup> Edition, Elsevier, 2010.  
(Chapters 1, 2, 3.1, 3.2, 4.1, 4.2, 4.3, 5.1, 5.2, 6.1)
2. Nader F. Mir: Computer and Communication Networks, Pearson Education, 2007.  
(Chapters 9.1 to 9.6, 18.1, 18.2)

### References:

1. Behrouz A. Forouzan: Data Communications and Networking, 4<sup>th</sup> Edition, Tata McGraw-Hill, 2006.
2. William Stallings: Data and Computer Communication, 8<sup>th</sup> Edition, Pearson Education, 2007.
3. Alberto Leon-Garcia and Indra Widjaja: Communication Networks -Fundamental Concepts and Key architectures, 2<sup>nd</sup> Edition Tata McGraw-Hill, 2004.

## Programming with Java

**Subject Code: 10MCA33**  
**Hours/Week: 4**  
**Total Hours: 52**

**IA Marks: 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **An Overview of Java**

**4 Hours**

Object-Oriented Programming, A First Simple Program, A Second Short Program, Using Blocks of Code Lexical Issues, The Java Class Libraries. Data Types, Variables, and Arrays: Java Is a Strongly Typed Language, The Primitive Types, A Closer Look at Literals, Variables, Type Conversion and casting, Automatic Type promotion in Expression, Arrays, A few words about Strings. Operators: Arithmetic Operators, The Bitwise Operators, Relational Operators Boolean Logical Operators, The Assignment Operator, The ? Operator, Operator Precedence, Using Parentheses. Control Statements: Java's Selection, Iteration Statements, Jump Statements.

### **Introducing Classes:**

**7 Hours**

Class Fundamentals, Declaring Objects, Assigning Object Reference Variables Introducing Methods, Constructors, The this Keyword, Garbage, The finalize() Method A Stack Class.

A Closer Look at Methods and Classes: Overloading Methods, Overloading Constructors, Using Objects as Parameters, A Closer Look at Argument Passing, Returning Objects, Recursion, Introducing Access Control, Understanding static, Introducing final, Arrays Revisited, Introducing Nested and Inner Classes, Exploring the String Class, Using Command-Line Arguments, Varargs

### **Inheritance:**

**7 Hours**

Inheritance Basics, Using super, Using super Creating a Multilevel Hierarchy, When Constructors are Called, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance, The Object Class. Packages and Interfaces: Packages, Access Protection, An Access Example Importing Packages, Interfaces. Exception Handling: Exception-Handling Fundamentals, Exception Types, Uncaught Exceptions Using try and catch, Multiple catch Clauses, Nested try Statements, throw, throws, finally, Java's Built-in Exceptions, Creating Your Own Exception Subclasses, Chained Exceptions, Using Exceptions.

### **Multithreaded Programming:**

**6 Hours**

The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization, Interthread Communication, Suspending, Resuming, and Stopping Threads, Using Multithreading.

Input/Output: Exploring java.io: The Java I/O Classes and Interfaces, File, The Closable and Flushable Interfaces, The Stream Classes, The Byte Streams, The Character Streams, The Console Class, Using Stream I/O, Serialization, Stream Benefits.

### **Enumeration, Autoboxing:**

**6 Hours**

Enumeration, Type Wrappers, Autoboxing. Generics: What are Generics?, A Simple Generics Example, A Generics Class with two Type Parameters, The General Form of a Generic Class. String Handling: The String Constructors, String Length, Special String Operations, Character Extraction, String Comparison, Searching Strings, Modifying a String, Data Conversion Using valueOf(), Changing the Case of Characters Within a String, Additional String Methods, StringBuffer, String Builder.

### **Exploring java.lang:**

**7 Hours**

Primitive Type Wrappers, System, Object, Class, Class Loader, Math, Thread, ThreadGroup, Runnable, Throwable. The Collections Framework: Collections Overview, The Collection Interfaces, The List Interface, The Queue Interface, The Collection Classes, The ArrayList Class, The LinkedList Class, The HashSet, The TreeSet Class.

### **Networking:**

**5 Hours**

Networking Basics, The Networking Classes and Interfaces InetAddress, TCP/IP Client Sockets, URL, URLConnection, HTTP URL Connection, TCP/IP Server Sockets, Cookies, Inet4Address and Inet6Address, The URI Class, RMI.

### **The Applet Class:**

**7 Hours**

\*\*\*

The Applet Class, Event Handling: Two Event Handling Mechanisms, The Delegation Event Model, Event Classes, Sources of Events, Event Listener Interfaces, Using the Delegation Event Model, Adapter Classes, Inner Classes.

Swing: Introducing Swing, Exploring Swing.

**Text Books:**

1. Herbert Schildt: The Complete Reference JAVA, 7<sup>th</sup> Edition, Tata McGraw Hill, 2006.  
(Chapters: 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 17, 19, 20, 21, 22, 27, 29, 30)

**Reference Books:**

1. James P Cohoon, Jack W Davidson : Programming in JAVA 5.0, Tata McGraw Hill, 2007.
2. Cay S Horstmann, Gary Cornell: Core Java 2 - Volume 1, 7th Edition, Pearson Education, 2005.
3. Cay S Horstmann, Gary Cornell: Core Java 2 - Volume 2, 8<sup>th</sup> Edition, Pearson Education, 2008.
4. Y. Daniel Liang: Introduction to JAVA Programming, 6<sup>th</sup> Edition, Pearson Education, 2007.



## Database Management Systems

**Subject Code: 10MCA34**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **Introduction**

**7 Hours**

Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures; Classification of Database Management systems.

### **Entity-Relationship Model**

**7 Hours**

Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two.

### **Relational Model and Relational Algebra**

**8 Hours**

Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations : JOIN and DIVISION; Additional Relational Operations; Examples of Queries in Relational Algebra; Relational Database Design Using ER- to-Relational Mapping.

### **SQL**

**16 Hours**

SQL Data Definition and Data Types; Specifying basic constraints in SQL; Schema change statements in SQL; Basic queries in SQL; More complex SQL Queries. Insert, Delete and Update statements in SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL / PSM.

### **Database Design**

**8 Hours**

Informal Design Guidelines for Relation Schemas; Functional Dependencies; Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form

### **Transaction Management**

**6 Hours**

The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery.

### **Text Books:**

1. Elmasri and Navathe: Fundamentals of Database Systems, 5<sup>th</sup> Edition, Pearson Education, 2007.  
(Chapters 1, 2, 3 except 3.8, 5, 6.1 to 6.5, 7.1, 8, 9.1, 9.2 except SQLJ, 9.4, 10)
2. Raghu Ramakrishnan and Johannes Gehrke: Database Management Systems, 3<sup>rd</sup> Edition, McGraw-Hill, 2003.  
(Chapters 16, 17.1, 17.2, 18)

### **Reference Books:**

1. Silberschatz, Korth and Sudharshan: Data base System Concepts, 5<sup>th</sup> Edition, Mc-GrawHill, 2006.
2. C.J. Date, A. Kannan, S. Swamynatham: A Introduction to Database Systems, 8<sup>th</sup> Edition, Pearson education, 2006.

## Operating Systems

**Subject Code: 10MCA35**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction to Operating Systems, System structures**

**6 Hours**

What operating systems do; Computer System organization; Computer System architecture; Operating System structure; Operating System operations; Process management; Memory management; Storage management; Protection and security; Distributed system; Special-purpose systems; Computing environments, Operating System Services; User - Operating System interface; System calls; Types of system calls; System programs; Operating System design and implementation; Operating System structure; Virtual machines; Operating System generation; System boot.

### **Process Management**

**7 Hours**

Process concept; Process scheduling; Operations on processes; Inter-process communication. Multi-Threaded Programming: Overview; Multithreading models; Thread Libraries; Threading issues. Process Scheduling: Basic concepts; Scheduling criteria; Scheduling algorithms; Multiple-Processor scheduling; Thread scheduling.

### **Process Synchronization**

**7 Hours**

Synchronization: The Critical section problem; Peterson's solution; Synchronization hardware; Semaphores; Classical problems of synchronization; Monitors.

### **Deadlocks**

**6 Hours**

Deadlocks: System model; Deadlock characterization; Methods for handling deadlocks; Deadlock prevention; Deadlock avoidance; Deadlock detection and recovery from deadlock.

### **Memory Management**

**7 Hours**

Memory Management Strategies: Background; Swapping; Contiguous memory allocation; Paging; Structure of page table; Segmentation. Virtual Memory Management: Background; Demand paging; Copy-on-write; Page replacement; Allocation of frames; Thrashing.

### **File System, Implementation of File System**

**7 Hours**

File System: File concept; Access methods; Directory structure; File system mounting; File sharing; Protection. Implementing File System: File system structure; File system implementation; Directory implementation; Allocation methods; Free space management

### **Secondary Storage Structures, Protection**

**6 Hours**

Mass storage structures; Disk structure; Disk attachment; Disk scheduling; Disk management; Swap space management. Protection: Goals of protection, Principles of protection, Domain of protection, Access matrix, Implementation of access matrix, Access control, Revocation of access rights, Capability-Based systems.

### **Case Study: The Linux Operating System**

**6 Hours**

Linux history; Design principles; Kernel modules; Process management; Scheduling; Memory management; File systems, Input and output; Inter-process communication.

### **Text Books:**

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 7<sup>th</sup> edition, Wiley India, 2006.
2. (Chapters: 1, 2, 3.1 to 3.4, 4.1 to 4.4, 5.1 to 5.5, 6.1 to 6.7, 7, 8.1 to 8.6, 9.1 to 9.6, 10, 11.1 to 11.5, 12.1 to 12.6, 17.1 to 17.8, 21.1 to 21.9)

### **Reference Books:**

1. D.M Dhamdhare: Operating systems - A concept based Approach, 2<sup>nd</sup> Edition, Tata McGraw- Hill, 2002.
2. P.C.P. Bhatt: Operating Systems, 2<sup>nd</sup> Edition, PHI, 2006.
3. Harvey M Deital: Operating systems, 3<sup>rd</sup> Edition, Pearson Education, 1990.

## Systems Programming Laboratory

**Subject Code: 10MCA36**  
**Hours/Week : 03**  
**Total Hours : 42**

**I.A. Marks: 50**  
**Exam Hours : 03**  
**Exam Marks : 50**

### PART- A

**Execute of the following programs using LEX:**

1. Program to count the number of vowels and consonants in a given string.
2. Program to count the number of characters, words, spaces and lines in a given input file.
3. Program to count number of
  - a. Positive and negative integers
  - b. Positive and negative fractions
4. Program to count the numbers of comment lines in a given C program. Also eliminate them and copy that program into separate file.
5. Program to count the number of 'scanf' and 'printf' statements in a C program. Replace them with 'readf' and 'writef' statements respectively.
6. Program to recognize a valid arithmetic expression and identify the identifiers and operators present. Print them separately.
7. Program to recognize and count the number of identifiers in a given input file.

### PART- B

**Execute of the following programs using YACC:**

1. Program to test the validity of a simple expression involving operators +, -, \* and /.
2. Program to recognize nested IF control statements and display the number of levels of nesting.
3. Program to recognize a valid arithmetic expression that uses operators +, -, \* and /.
4. Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.
5. Program to evaluate an arithmetic expression involving operators +, -, \* and /.
6. Program to recognize strings 'aab', 'abbb', 'ab' and 'a' using the grammar:  
( $a^n b^n, n \geq 0$ )
7. Program to recognize the grammar ( $a^n b, n \geq 10$ ).

#### **Instructions:**

In the examination, a combination of one LEX and one YACC problem has to be asked based on lots. Both will have equal weightages.

## Java Programming Laboratory

**Subject Code: 10MCA37**

**Hours/Week: 3**

**Total Hours: 42**

**I.A Marks: 50**

**Exam Marks: 50**

**Exam Hours: 3**

1.
  - a. Write a JAVA Program to demonstrate Constructor Overloading and Method overloading.
  - b. Write a JAVA Program to implement Inner class and demonstrate its Access Protections.
2.
  - a. Write a JAVA Program to demonstrate Inheritance.
  - b. Write a JAVA Program to demonstrate Exception Handling (Using Nested try catch and finally).
3. Write a JAVA program which has
  - i. A Class called Account that creates account with 500Rs minimum balance, a deposit() method to deposit amount, a withdraw() method to withdraw amount and also throws LessBalanceException if an account holder tries to withdraw money which makes the balance become less than 500Rs.
  - ii. A Class called LessBalanceException which returns the statement that says withdraw amount (\_\_\_Rs) is not valid.
  - iii. A Class which creates 2 accounts, both account deposit money and one account tries to withdraw more money which generates a LessBalanceException take appropriate action for the same.
4. Write a JAVA program using Synchronized Threads, which demonstrates Producer Consumer concept.
5. Write a JAVA program which has
  - i. A Interface class for Stack Operations
  - ii. A Class that implements the Stack Interface and creates a fixed length Stack.
  - iii. A Class that implements the Stack Interface and creates a Dynamic length Stack.
  - iv. A Class that uses both the above Stacks through Interface reference and does the Stack operations that demonstrates the runtime binding.
6. Write a JAVA program which has
  - i. 2 classes which initializes a String in its constructor
  - ii. A Generic class with 2 type Parameters
  - iii. Create a Generic Class reference for the above 2 Class and try to print the message inside the constructor (Use to string method).
7. Write JAVA programs which demonstrates utilities of LinkedList Class
8. Write a JAVA Program which uses FileInputStream / FileOutPutStream Classes.
9. Write a JAVA Program which writes a object to a file (use transient variable also).
10. Write a JAVA program which uses Datagram Socket for Client Server Communication.
11. Write JAVA Applet programs which handles MouseEvent
12. Write JAVA Applet programs which handles KeyboardEvent
13. Write a JAVA program which implements RMI.
14. Write a Swing Application which uses
  - i. JTabbed Pane
  - ii. Each Tab should use JPanel, which includes any one component given below in each Panel
  - iii. ComboBox / List / Tree / Radiobutton

\*\*\*

**Note:**

**All the above Components should Listen to any one of their respective events and print appropriate message.**

## DBMS Laboratory

**Subject Code: 10MCA38**

**Hours/Week : 03**

**Total Hours : 42**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 50**

1. Consider the following relations:
  - Student (*snum*: integer, *sname*: string, *major*: string, *level*: string, *age*: integer)
  - Class (*name*: string, *meets at*: string, *room*: string, *d*: integer)
  - Enrolled (*snum*: integer, *cname*: string)
  - Faculty (*fid*: integer, *fname*: string, *deptid*: integer)

The meaning of these relations is straightforward; for example, Enrolled has one record per student-class pair such that the student is enrolled in the class. Level is a two character code with 4 different values (example: Junior: JR etc)

Write the following queries in SQL. No duplicates should be printed in any of the answers.

  - i. Find the names of all Juniors (level = JR) who are enrolled in a class taught by Prof. Harshith
  - ii. Find the names of all classes that either meet in room R128 or have five or more Students enrolled.
  - iii. Find the names of all students who are enrolled in two classes that meet at the same time.
  - iv. Find the names of faculty members who teach in every room in which some class is taught.
  - v. Find the names of faculty members for whom the combined enrollment of the courses that they teach is less than five.
  
2. The following relations keep track of airline flight information:
  - Flights (*no*: integer, *from*: string, *to*: string, *distance*: integer, *Departs*: time, *arrives*: time, *price*: real)
  - Aircraft (*aid*: integer, *aname*: string, *cruisingrange*: integer)
  - Certified (*eid*: integer, *aid*: integer)
  - Employees (*eid*: integer, *ename*: string, *salary*: integer)

Note that the Employees relation describes pilots and other kinds of employees as well; Every pilot is certified for some aircraft, and only pilots are certified to fly.

Write each of the following queries in SQL.

  - i. Find the names of aircraft such that all pilots certified to operate them have salaries more than Rs.80,000.
  - ii. For each pilot who is certified for more than three aircrafts, find the *eid* and the maximum *cruisingrange* of the aircraft for which she or he is certified.
  - iii. Find the names of pilots whose *salary* is less than the price of the cheapest route from Bengaluru to Frankfurt.
  - iv. For all aircraft with *cruisingrange* over 1000 Kms, find the name of the aircraft and the average salary of all pilots certified for this aircraft.
  - v. Find the names of pilots certified for some Boeing aircraft.
  - vi. Find the *aids* of all aircraft that can be used on routes from Bengaluru to New Delhi.
  
3. Consider the following database of student enrollment in courses & books adopted for each course.
  - STUDENT (regno: string, name: string, major: string, bdate:date)
  - COURSE (course #:int, cname:string, dept:string)
  - ENROLL ( regno:string, course#:int, sem:int, marks:int)
  - BOOK \_ ADOPTION (course#:int, sem:int, book-ISBN:int)
  - TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)
  - i. Create the above tables by properly specifying the primary keys and the foreign keys.
  - ii. Enter at least five tuples for each relation.
  - iii. Demonstrate how you add a new text book to the database and make this book be adopted by some department.
  - iv. Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
  - v. List any department that has *all* its adopted books published by a specific publisher.
  - vi. Generate suitable reports.
  - vii. Create suitable front end for querying and displaying the results.

4. The following tables are maintained by a book dealer.
- AUTHOR (author-id:int, name:string, city:string, country:string)  
 PUBLISHER (publisher-id:int, name:string, city:string, country:string)  
 CATALOG (book-id:int, title:string, author-id:int, publisher-id:int, category-id:int, year:int, price:int)  
 CATEGORY (category-id:int, description:string)  
 ORDER-DETAILS (order-no:int, book-id:int, quantity:int)
- i. Create the above tables by properly specifying the primary keys and the foreign keys.
  - ii. Enter at least five tuples for each relation.
  - iii. Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
  - iv. Find the author of the book which has maximum sales.
  - v. Demonstrate how you increase the price of books published by a specific publisher by 10%.
  - vi. Generate suitable reports.
  - vii. Create suitable front end for querying and displaying the results.
5. Consider the following database for a banking enterprise
- BRANCH(branch-name:string, branch-city:string, assets:real)  
 ACCOUNT(accno:int, branch-name:string, balance:real)  
 DEPOSITOR(customer-name:string, accno:int)  
 CUSTOMER(customer-name:string, customer-street:string, customer-city:string)  
 LOAN(loan-number:int, branch-name:string, amount:real)  
 BORROWER(customer-name:string, loan-number:int)
- i. Create the above tables by properly specifying the primary keys and the foreign keys
  - ii. Enter at least five tuples for each relation
  - iii. Find all the customers who have at least two accounts at the *Main* branch.
  - iv. Find all the customers who have an account at *all* the branches located in a specific city.
  - v. Demonstrate how you delete all account tuples at every branch located in a specific city.
  - vi. Generate suitable reports.
  - vii. Create suitable front end for querying and displaying the results.

**Instructions:**

1. The exercises are to be solved in an RDBMS environment like Oracle or DB2.
2. Suitable tuples have to be entered so that queries are executed correctly.
3. Front end may be created using either VB or VAJ or any other similar tool.
4. The student need not create the front end in the examination. The results of the queries may be displayed directly.
5. Relevant queries other than the ones listed along with the exercises may also be asked in the examination.
6. Questions must be asked based on lots.

## IV SEMESTER

## Topics in Enterprise Architecture - I

**Subject Code: 10MCA41**  
**Hours/Week: 4**  
**Total Hours: 52**

**IA Marks: 50**  
**Exam Marks: 100**  
**Exam Hours: 3**

**JDBC Object** **5 Hours**  
 The Concept of JDBC, JDBC Driver Types, JDBC Packages, A Brief Overview of JDBC Process, Database Connection, Associating the JDBC/ODBC Bridge with the Database, Statement Objects, ResultSet, Transaction Processing, Metadata, Data Types, Exceptions.

**JDBC and Embedded SQL** **5 Hours**  
 Model Programs, Tables, Indexing, Inserting Data into Tables, Selecting Data from a Table, Metadata, Updating Tables, Deleting Data from a Table, Joining Tables, Calculating Data, Grouping and Ordering Data, Subqueries, VIEW.

**Servlets** **6 Hours**  
 Introduction, Advantages of Servlets over CGI, Installing Servlets, The Servlets Life Cycle, Servlet API, Handling HTTP GET Request, Handling HTTP POST Request, Servlet Context.

**Servlets - Sessions** **5 Hours**  
 Cookies, Session Tracking, Filter API, Multi-tier Applications Using Database Connectivity

**Java Server Pages (JSP)** **8 Hours**  
 Introduction, Advantages of JSP, Developing First JSP, JSP Scripting Elements- (Directives, Declaratives, Scriptlets, Expressions, Implicit Variables), Page Directives, JSTL, Standard Action, Custom Tags.

**Java Beans** **7 Hours**  
 What is a Java Bean? Advantages of Java Beans, The Java Beans API – Introspector, property Descriptor, Event Descriptor, Method Descriptor, A Bean Example, JSP with Java Beans

**Enterprise Java Beans** **8 Hours**  
 Deployment Descriptors; Session Java Bean, Entity Java Bean; Message-Driven Bean; The JAR File.

**J2ME** **8 Hours**

**Text Books:**

1. Web Technologies: Html, Javascript, Php, Java, Jsp, ASP. Net, Xml and Ajax, Black Book by Kogent Learning Solutions Inc., 2009. (Chapters: 9, 10, 11, 12, 13 (UNIT 1-6))
2. [http://java.sun.com/j2ee/tutorial/1\\_3-fcs/doc/EJBConcepts.html](http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/EJBConcepts.html), <http://www.roseindia.net/ejb/>, <http://www.roseindia.net/strutstutorial/introduction.shtml> (UNIT 7 & 8)

**Reference Books:**

1. Jim Keogh: The complete Reference J2EE (J2EE1.4), Tata McGraw Hill, 2002.
2. James McGovern, Rahim Adatia, Yakov Fain, Jason Gordon, Ethan Henry, Walter Hurst, Ashish Jain, Mark Little, Vaidyanathan Nagarajan, Harshad Oak, Lee Anne Phillips: Java 2 Enterprise Edition 1.4 (J2EE 1.4) Bible, Wiley, 2003.
3. Bond, Law, Longshaw, Haywood, Roxburgh: Teach Yourself J2EE (J2EE 1.4), 2<sup>nd</sup> Edition, Pearson Education, 2004.



## Software Engineering

**Subject Code: 10MCA42**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Overview**

**6 Hours**

Introduction: FAQ's about software engineering, Professional and ethical responsibility.

Socio-Technical systems: Emergent system properties; Systems engineering; Organizations, people and computer systems; Legacy systems.

### **Software Processes**

**4 Hours**

Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering.

### **Requirements**

**8 Hours**

Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; Interface specification; The software requirements document.

Requirements Engineering Processes: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management.

### **System models, Project Management**

**8 Hours**

System Models: Context models; Behavioral models; Data models; Object models; Structured methods.

Project Management: Management activities; Project planning; Project scheduling; Risk management

### **Software Design**

**7 Hours**

Architectural Design: Architectural design decisions; System organization; Modular decomposition styles; Control styles.

Object-Oriented design: Objects and Object Classes; An Object-Oriented design process; Design evolution.

### **Development**

**6 Hours**

Rapid Software Development: Agile methods; Extreme programming; Rapid application development.

Software Evolution: Program evolution dynamics; Software maintenance; Evolution processes; Legacy system evolution.

### **Verification and Validation**

**7 Hours**

Verification and Validation: Planning; Software inspections; Automated static analysis; Verification and formal methods.

Software testing: System testing; Component testing; Test case design; Test automation.

### **Management**

**6 Hours**

Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model.

Software Cost Estimation: Productivity; Estimation techniques; Algorithmic cost modeling, Project duration and staffing.

### **Text Books:**

1. Ian Sommerville: Software Engineering, 8<sup>th</sup> Edition, Pearson Education, 2007.  
(Chapters:- 1, 2, 3, 4, 5, 6, 7, 8, 11, 14, 17, 21, 22, 23, 25, 26)

### **Reference Books:**

1. Roger.S.Pressman: Software Engineering-A Practitioners approach, 7<sup>th</sup> Edition, McGraw Hill, 2007.
2. Pankaj Jalote: An Integrated Approach to Software Engineering, 3<sup>rd</sup> Edition, Narosa Publishing House, 2005.

## Web Programming

**Subject Code: 10MCA43**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **Fundamentals of Web, XHTML, CSS**

**13 Hours**

Internet, WWW, Web Browsers and Web Servers, URLs, MIME, HTTP, Security, The Web Programmers Toolbox.

XHTML: Basic syntax, Standard structure, Basic text markup, Images, Hypertext Links. Lists, Tables, Forms, Frames.

CSS: Introduction, Levels of style sheets, Style specification formats, Selector forms, Property value forms, Font properties, List properties, Color, Alignment of text, The box model, Background images, The <span> and <div> tags, Conflict resolution.

### **Javascript**

**6 Hours**

Overview of Javascript, Object orientation and Javascript, Syntactic characteristics, Primitives, operations, and expressions, Screen output and keyboard input, Control statements, Object creation and modification, Arrays, Functions, Constructors, Pattern matching using regular expressions, Errors in scripts, Examples.

### **Javascript and HTML Documents, Dynamic Documents with Javascript**

**7 Hours**

The Javascript execution environment, The Document Object Model, Element access in Javascript, Events and event handling, Handling events from the Body elements, Button elements, Text box and Password elements, The DOM 2 event model, The navigator object, DOM tree traversal and modification.

Introduction to dynamic documents, Positioning elements, Moving elements, Element visibility, Changing colors and fonts, Dynamic content, Stacking elements, Locating the mouse cursor, Reacting to a mouse click, Slow movement of elements, Dragging and dropping elements.

### **XML**

**6 Hours**

Introduction, Syntax, Document structure, Document type definitions, Namespaces, XML schemas, Displaying raw XML documents, Displaying XML documents with CSS, XSLT style sheets, XML processors, Web services.

### **Perl, CGI Programming**

**7 Hours**

Origins and uses of Perl, Scalars and their operations, Assignment statements and simple input and output, Control statements, Fundamentals of arrays, Hashes, References, Functions, Pattern matching, File input and output; Examples.

The Common Gateway Interface; CGI linkage; Query string format; CGI.pm module; A survey example; Cookies.

Database access with Perl and MySQL

### **PHP**

**6 Hours**

Origins and uses of PHP, Overview of PHP, General syntactic characteristics, Primitives, operations and expressions, Output, Control statements, Arrays, Functions, Pattern matching, Form handling, Files, Cookies, Session tracking, Database access with PHP and MySQL.

### **Ruby, Rails**

**7 Hours**

Origins and uses of Ruby, Scalar types and their operations, Simple input and output, Control statements, Arrays, Hashes, Methods, Classes, Code blocks and iterators, Pattern matching.

Overview of Rails, Document requests, Processing forms, Rails applications with Databases, Layouts.

### **Text Books:**

1. Robert W. Sebesta: Programming the World Wide Web, 4th Edition, Pearson education, 2008.  
(Listed topics only from Chapters 1 to 9, 11 to 15)

### **Reference Books:**

1. M. Deitel, P.J. Deitel, A. B. Goldberg: Internet & World Wide Web How to Program, 3<sup>rd</sup> Edition, Pearson education, 2004.
2. Chris Bates: Web Programming Building Internet Applications, 3<sup>rd</sup> Edition, Wiley India, 2006.
3. Xue Bai et al: The web Warrior Guide to Web Programming, Thomson, 2003.

## Design and Analysis of Algorithms

**Subject Code: 10MCA44**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **Introduction**

**7 Hours**

Notion of Algorithm, Review of Asymptotic Notations, Mathematical Analysis of Non-Recursive and Recursive Algorithms

Brute Force Approaches: Introduction, Selection Sort and Bubble Sort, Sequential Search and Brute Force String Matching.

### **Divide and Conquer**

**6 Hours**

Divide and Conquer: General Method, Defective Chess Board, Binary Search, Merge Sort, Quick Sort and its performance.

### **The Greedy Method**

**7 Hours**

The General Method, Knapsack Problem, Job Sequencing with Deadlines, Minimum-Cost Spanning Trees: Prim's Algorithm, Kruskal's Algorithm; Single Source Shortest Paths.

### **Dynamic Programming**

**6 Hours**

The General Method, Warshall's Algorithm, Floyd's Algorithm for the All-Pairs Shortest Paths Problem, Single-Source Shortest Paths: General Weights, 0/1 Knapsack, The Traveling Salesperson problem.

### **Decrease-and-Conquer Approaches, Space-Time Tradeoffs**

**7 Hours**

Decrease-and-Conquer Approaches: Introduction, Insertion Sort, Depth First Search and Breadth First Search, Topological Sorting,  
 Space-Time Tradeoffs: Introduction, Sorting by Counting, Input Enhancement in String Matching.

### **Limitations of Algorithmic Power and Coping with them**

**7 Hours**

Lower-Bound Arguments, Decision Trees, P, NP, and NP-Complete Problems, Challenges of Numerical Algorithms.

### **Coping with Limitations of Algorithmic Power**

**6 Hours**

Backtracking: n - Queens problem, Hamiltonian Circuit Problem, Subset – Sum Problem.

Branch-and-Bound: Assignment Problem, Knapsack Problem, Traveling Salesperson Problem.

Approximation Algorithms for NP-Hard Problems – Traveling Salesperson Problem, Knapsack Problem

### **PRAM Algorithms**

**6 Hours**

Introduction, Computational Model, Parallel Algorithms for Prefix Computation, List Ranking, and Graph Problems

### **Text Books:**

1. Anany Levitin: Introduction to The Design & Analysis of Algorithms, 2<sup>nd</sup> Edition, Pearson Education, 2007.  
 (Listed topics only from the Chapters 1, 2, 3, 5, 7, 8, 10, 11)
2. Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran: Fundamentals of Computer Algorithms, 2<sup>nd</sup> Edition, University press, 2007.  
 (Listed topics only from the Chapters 3, 4, 5, 13)

### **Reference Books:**

1. Introduction to Algorithms, Thomas H. Cormen, Charles E. Leiserson, Ronal L. Rivest, Clifford Stein, 2<sup>nd</sup> Edition, PHI, 2006.
2. Introduction to the Design and Analysis of Algorithms A Strategic Approach, R.C.T. Lee, S.S. Tseng, R.C. Chang & Y.T.Tsai, Tata McGraw Hill, 2005.

## Computer Graphics and Visualization

**Subject Code: 10MCA451**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 05**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**7 Hours**

Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging Systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable Pipelines; Performance Characteristics  
Graphics Programming: The Sierpinski gasket; Programming Two Dimensional Applications.

### **The OpenGL**

**6 Hours**

The OpenGL API; Primitives and attributes; Color; Viewing; Control functions; The Gasket program; Polygons and recursion; The three-dimensional gasket; Plotting Implicit Functions

### **Input and Interaction**

**7 Hours**

Interaction; Input devices; Clients and Servers; Display Lists; Display Lists and Modeling; Programming Event Driven Input; Menus; Picking; A simple CAD program; Building Interactive Models; Animating Interactive Programs; Design of Interactive Programs; Logic Operations

### **Geometric Objects and Transformations**

**11 Hours**

Scalars, Points, and Vectors; Three-dimensional Primitives; Coordinate Systems and Frames; Modeling a Colored Cube; Affine Transformations; Rotation, Translation and Scaling; Geometric Objects and Transformations; Transformation in Homogeneous Coordinates; Concatenation of Transformations; OpenGL Transformation Matrices; Interfaces to three-dimensional applications; Quaternions

### **Viewing**

**7 Hours**

Classical and computer viewing; Viewing with a Computer; Positioning of the camera; Simple projections; Projections in OpenGL; Hidden-surface removal; Interactive Mesh Displays; Parallel-projection matrices; Perspective-projection matrices; Projections and Shadows.

### **Lighting and Shading**

**6 Hours**

Light and Matter; Light Sources; The Phong Lighting model; Computation of vectors; Polygonal Shading; Approximation of a sphere by recursive subdivisions; Light sources in OpenGL; Specification of materials in OpenGL; Shading of the sphere model; Global Illumination.

### **Implementation**

**8 Hours**

Basic Implementation Strategies; Four major tasks; Clipping; Line-segment clipping; Polygon clipping; Clipping of other primitives; Clipping in three dimensions; Rasterization; Bresenham's algorithm; Polygon Rasterization; Hidden-surface removal; Antialiasing; Display considerations.

### **Text Books:**

1. Edward Angel: Interactive Computer Graphics A Top-Down Approach with OpenGL, 5<sup>th</sup> Edition, Pearson Education, 2008.  
(Chapters 1 to 7)

### **Reference Books:**

1. Donald Hearn and Pauline Baker: Computer Graphics- OpenGL Version, 3<sup>rd</sup> Edition, Pearson Education, 2004.
2. F.S. Hill Jr.: Computer Graphics Using OpenGL, 2<sup>nd</sup> Edition, Pearson education, 2001.
3. James D Foley, Andries Van Dam, Steven K Feiner, John F Hughes, Computer Graphics, Addison-Wesley 1997.

## UNIX Systems Programming

**Subject Code: 10MCA452**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**6 Hours**

UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards.

UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.

### **UNIX Files**

**6 Hours**

File Types, The UNIX and POSIX File System, The UNIX and POSIX File Attributes, Inodes in UNIX System V, Application Program Interface to Files, UNIX Kernel Support for Files, Relationship of C Stream Pointers and File Descriptors, Directory Files, Hard and Symbolic Links.

### **UNIX File APIs**

**7 Hours**

General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs, General File Class, regfile Class for Regular Files, dirfile Class for Directory Files, FIFO File Class, Device File Class, Symbolic Link File Class, File Listing Program.

### **UNIX Processes**

**7 Hours**

The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes.

### **Process Control**

**7 Hours**

Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, waited, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times.

Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp, tcsetpgrp, and tcgetsid Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups.

### **Signals and Daemon Processes**

**7 Hours**

Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers.

Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Single-instance daemons; Daemon conventions; Client-Server Model.

### **Interprocess Communication**

**6 Hours**

Introduction; Pipes, popen, pclose Functions; Coprocesses; FIFOs; XSI IPC; Message Queues; Semaphores

### **Network IPC: Sockets**

**6 Hours**

Introduction; Socket Descriptors; Addressing; Connection establishment; Data transfer; Socket options; Out-of-band data; Nonblocking and asynchronous I/O.

### **Text Books:**

1. Terrence Chan: Unix System Programming Using C++, Prentice-Hall of India / Pearson Education, 1999.  
(Chapters 1, 5, 6, 7, 8, 9)
2. W.Richard Stevens, Stephen A. Rago: Advanced Programming in the UNIX Environment, 2<sup>nd</sup> Edition, Pearson Education / Prentice-Hall of India, 2005.  
(Chapters 7, 8, 9, 13, 15, 16)

### **Reference Books:**

1. Marc J. Rochkind: Advanced UNIX Programming, 2<sup>nd</sup> Edition, Pearson Education, 2005.
2. Maurice.J.Bach: The Design of the UNIX Operating System, Pearson Education, 1987.
3. Uresh Vahalia: UNIX Internals, Pearson Education, 2001.

## Multimedia Systems

**Subject Code: 10MCA453**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **Introduction, Media and Data Streams, Audio Technology 7 Hours**

Multimedia Elements; Multimedia Applications; Multimedia Systems Architecture; Evolving Technologies for Multimedia Systems; Defining Objects for Multimedia Systems; Multimedia Data Interface Standards; The need for Data Compression; Multimedia Databases.

Media: Perception Media, Representation Media, Presentation Media, Storage Media, Transmission Media, Information Exchange Media, Presentation Spaces & Values, and Presentation Dimensions; Key Properties of a Multimedia System: Discrete & Continuous Media, Independence Media, Computer Controlled Systems, Integration; Characterizing Data Streams: Asynchronous Transmission Mode, Synchronous Transmission Mode, Isochronous Transmission Mode; Characterizing Continuous Media Data Streams.

Sound: Frequency, Amplitude, Sound Perception and Psychoacoustics; Audio Representation on Computers; Three Dimensional Sound Projection; Music and MIDI Standards; Speech Signals; Speech Output; Speech Input; Speech Transmission.

### **Graphics and Images, Video Technology, Computer-Based Animation 7 Hours**

Capturing Graphics and Images Computer Assisted Graphics and Image Processing; Reconstructing Images; Graphics and Image Output Options.

Basics; Television Systems; Digitalization of Video Signals; Digital Television; Basic Concepts; Specification of Animations; Methods of Controlling Animation; Display of Animation; Transmission of Animation; Virtual Reality Modeling Language.

### **Data Compression 12 Hours**

Storage Space; Coding Requirements; Source, Entropy, and Hybrid Coding; Basic Compression Techniques; JPEG: Image Preparation, Lossy Sequential DCT-based Mode, Expanded Lossy DCT-based Mode, Lossless Mode, Hierarchical Mode.

H.261 (Px64) and H.263: Image Preparation, Coding Algorithms, Data Stream, H.263+ and H.263L; MPEG: Video Encoding, Audio Coding, Data Stream, MPEG-2, MPEG-4, MPEG-7; Fractal Compression.

### **Optical Storage Media 6 Hours**

History of Optical Storage; Basic Technology; Video Discs and Other WORMs; Compact Disc Digital Audio; Compact Disc Read Only Memory; CD-ROM Extended Architecture; Further CD-ROM-Based Developments; Compact Disc Recordable; Compact Disc Magneto-Optical; Compact Disc Read/Write; Digital Versatile Disc.

### **Content Analysis 6 Hours**

Simple Vs. Complex Features; Analysis of Individual Images; Analysis of Image Sequences; Audio Analysis; Applications.

### **Data and File Format Standards 7 Hours**

Rich-Text Format; TIFF File Format; Resource Interchange File Format (RIFF); MIDI File Format; JPEG DIB File Format for Still and Motion Images; AVI Indeo File Format; MPEG Standards; TWAIN

### **Multimedia Application Design 7 Hours**

Multimedia Application Classes; Types of Multimedia Systems; Virtual Reality Design; Components of Multimedia Systems; Organizing Multimedia Databases; Application Workflow Design Issues; Distributed Application Design Issues.

#### **Text Books:**

1. Ralf Steinmetz, Klara Narstedt: Multimedia Fundamentals: Vol 1-Media Coding and Content Processing, 2<sup>nd</sup> Edition, Pearson Education, 2003.  
(Chapters 2, 3, 4, 5, 6, 7, 8, 9)
2. Prabhat K. Andleigh, Kiran Thakrar: Multimedia Systems Design, PHI, 2003.  
(Chapters 1, 3, 7)

**Reference Books:**

1. K.R Rao, Zoran S. Bojkovic and Dragorad A. Milovanovic: Multimedia Communication Systems: Techniques, Standards, and Networks, Pearson Education, 2002.
2. Nalin K Sharad: Multimedia information Networking, PHI, 2002.

## Pattern Recognition

**Subject Code: 10MCA454**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**6 Hours**

Machine perception, an example; Pattern Recognition System; The Design Cycle; Learning and Adaptation

### **Bayesian Decision Theory**

**7 Hours**

Introduction, Bayesian Decision Theory; Continuous Features, Minimum error rate, classification, classifiers, discriminant functions, and decision surfaces; The normal density; Discriminant functions for the normal density

### **Maximum-likelihood and Bayesian Parameter Estimation**

**7 Hours**

Introduction; Maximum-likelihood estimation; Bayesian Estimation; Bayesian parameter estimation: Gaussian Case, general theory; Hidden Markov Models

### **Non-parametric Techniques**

**6 Hours**

Introduction; Density Estimation; Parzen windows;  $k_n$  – Nearest- Neighbor Estimation; The Nearest-Neighbor Rule; Metrics and Nearest-Neighbor Classification

### **Linear Discriminant Functions**

**7 Hours**

Introduction; Linear Discriminant Functions and Decision Surfaces; Generalized Linear Discriminant Functions; The Two-Category Linearly Separable case; Minimizing the Perception Criterion Functions; Relaxation Procedures; Non-separable Behavior; Minimum Squared-Error procedures; The Ho-Kashyap procedures

### **Stochastic Methods**

**6 Hours**

Introduction; Stochastic Search; Boltzmann Learning; Boltzmann Networks and Graphical Models; Evolutionary Methods

### **Non-Metric Methods**

**6 Hours**

Introduction; Decision Trees; CART; Other Tree Methods; Recognition with Strings; Grammatical Methods

### **Unsupervised Learning and Clustering**

**7 Hours**

Introduction; Mixture Densities and Identifiability; Maximum-Likelihood Estimates; Application to Normal Mixtures; Unsupervised Bayesian Learning; Data Description and Clustering; Criterion Functions for Clustering

### **Text Books:**

1. Richard O. Duda, Peter E. Hart, and David G. Stork: Pattern Classification, 2<sup>nd</sup> Edition, Wiley-Interscience, 2001.

### **Reference Books:**

1. Earl Gose, Richard Johnsonbaugh, Steve Jost: Pattern Recognition and Image Analysis, Pearson Education, 2007.



## Principles of User Interface Design

**Subject Code: 10MCA455**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**8 Hours**

**Usability of Interactive Systems:** Introduction, Usability Requirements, Usability measures, Usability Motivations, Universal Usability, Goals for our profession

**Guideline, principles, and Theories:** Introduction, Guidelines, principles, Theories, Object-Action Interface Model

### **Development Processes**

**5 Hours**

**Managing Design Processes:** Introduction, Organizational Design to support Usability, The Three pillars of design, Development Methodologies, Ethnographic Observation, Participatory Design, Scenario Development, Social Impact statement for Early Design Review, Legal Issues.

### **Evaluating Interface Designs**

**7 Hours**

Introduction, Expert Reviews, Usability Testing and Laboratories, Survey Instruments, Acceptance Tests, Evaluation During Active Use, Controlled Psychologically Oriented Experiments.

**Software Tools:** Introduction, Specification Methods, Interface-Building Tools, Evaluation and Critiquing Tools.

### **Interaction Styles**

**8 Hours**

**Direct Manipulation and Virtual Environments:** Introduction, Examples of Direct Manipulation, 3D Interfaces, Teleoperation, Virtual and Augmented Reality.

**Menu Selection, Form Fillin, and Dialog Boxes:** Introduction, Task-Related Menu Organization, Single Menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus, Data Entry with Menus: Form Fillin, Dialog Boxes, and Alternatives, Audio Menus and Menus for small Displays.

### **Command and Natural Languages**

**8 Hours**

Introduction, Functionality to Support User's Tasks, Command-Organization Strategies, The Benefits of Structure, Naming and Abbreviations, Natural Language in Computing.

**Interaction Devices:** Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory interfaces, Displays-Small and Large, Printers.

### **Design Issues**

**6 Hours**

**Quality of Service:** Introduction, Models of Response-Time Impacts, Expectations and Attitudes, User Productivity, Variability in Response Time, Frustrating Experiences.

**Balancing Function and Fashion:** Introduction, Error Messages, Nonanthropomorphic Design, Display Design, Window Design, Color.

### **User Manuals, Online Help, and Tutorials**

**5 Hours**

Introduction, Paper versus Online Manuals, Reading from Paper Verses from Displays, Shaping the Content of the Manuals, Online Manuals and Help, Online Tutorials, Demonstrations, and Guides, Online Communities for User Assistance, The Development Process.

### **Information Search and Visualization:**

**5 Hours**

Introduction, Search in Textual Documents and Database Querying, Multimedia Document Searches, Advanced Filtering and Search Interfaces, Information Visualization

### **Text Books:**

1. Ben Shneiderman: Designing the User Interface, 4<sup>th</sup> Edition, Pearson Education, 2009. (Chapters 1 to 9 and 11 to 14)

### **Reference Books:**

1. Alan J Dix et. al.: Human-Computer Interaction, II Edition, Prentice-Hall India, 1998.
2. Eberts: User Interface Design, Prentice-Hall, 1994.
3. Wilber O Galitz: The Essential Guide to User Interface Design - An Introduction to GUI Design, Principles and Techniques, Wiley-Dreamtech India Pvt. Ltd, 1998.

## Advanced Computer Networks

**Subject Code: 10MCA456**  
**Hours/Week: 04**  
**Total Hours: 52**

**IA Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

<b>Review of Network Models</b>	<b>5 Hours</b>
Layered tasks; The OSI model and layers in the OSI model; TCP / IP protocol suite; Addressing	
<b>SONET / SDH</b>	<b>5 Hours</b>
Architecture; SONET layers; SONET frames; STS multiplexing; SONET networks; Virtual tributaries	
<b>Frame Relay and ATM</b>	<b>4 Hours</b>
Frame relay; ATM and ATM LANs	
<b>IPv6, Address Mapping and Error Reporting</b>	<b>6 Hours</b>
IPv6: Advantages, Packet format, and Extension headers; Transition from IPv4 to IPv6: Dual stack, Tunneling, and Header translation; Address mapping: ARP, RARP, BOOTP, and DHCP; Error reporting: ICMP.	
<b>Multicast Routing Protocols</b>	<b>4 Hours</b>
Unicast, multicast and broadcast; Applications; Multicasting routing; Routing protocols.	
<b>SCTP</b>	<b>4 Hours</b>
SCTP services; SCTP features; Packet format; An SCTP association; Flow control; Error control; Congestion control.	
<b>Congestion Control and Quality of Service</b>	<b>6 Hours</b>
Data traffic; Congestion and congestion control; Congestion control in TCP, Frame relay; Quality of Service; Techniques to improve QoS; Integrated services; Differentiated services	
<b>Multimedia</b>	<b>6 Hours</b>
Digitizing audio and video; Audio and video compression; Streaming stored audio / video; Streaming live audio / video; Real-time interactive audio / video; RTP; RTCP; VoIP.	
<b>Mobile Ad-Hoc Networks, Wireless Sensor Networks</b>	<b>12 Hours</b>
Overview of wireless ad-hoc networks; Routing in ad-hoc networks; Routing protocols for ad-hoc networks; Security of ad-hoc networks. Sensor networks and protocol structures; Communication energy model; Clustering protocols; Routing protocols; Zigbee technology and IEEE 802.15.4	

### Text Books:

1. Behrouz A. Forouzan, Data Communications and Networking, 4<sup>th</sup> Edition, Tata McGraw-Hill, 2006.  
(Chapters 2, 17, 18, 20.3, 20.4, 21.1, 21.2, 22.4, 23.4, 24, 29)
2. Nader F. Mir: Computer and Communication Networks, Pearson, 2007.  
(Chapters 19, 20 excluding 20.5)

### References:

1. William Stallings: Data and Computer Communication, 8<sup>th</sup> Edition, Prentice Hall India, 2007.
2. William A. Shay: Understanding Data Communications and Networks, 3<sup>rd</sup> Edition, Thomson, 2003.
3. Godbole: Data Communications and Networks, Tata McGraw-Hill, 2002.
4. Micael A. Gallo & William M. Handcock: Computer Communications and Networking Technologies, Thomson, 2003.

**J2EE Laboratory**

**Subject Code: 10MCA46**  
**Hours/Week: 3**  
**Total Hours: 42**

**I.A Marks: 50**  
**Exam Hours: 03**  
**Exam Marks: 50**

1. Write a JAVA Program to insert data into Student DATA BASE and retrieve info based on particular queries (queries can be given which covers all the topics of 2<sup>nd</sup> UNIT).
2. Write a JAVA Servlet Program to implement a dynamic HTML using Servlet (user name and password should be accepted using HTML and displayed using a Servlet).
3. Write a JAVA Servlet Program to Download a file and display it on the screen (A link has to be provided in HTML, when the link is clicked corresponding file has to be displayed on Screen)
4. Write a JAVA Servlet Program to implement RequestDispatcher object (use include() and forward() methods).
5. Write a JAVA Servlet Program to implement and demonstrate get() and Post methods(Using HTTP Servlet Class).
6. Write a JAVA Servlet Program to implement sendRedirect() method(using HTTP Servlet Class).
7. Write a JAVA Servlet Program to implement sessions (Using HTTP Session Interface).
8.
  - a. Write a JAVA JSP Program to print 10 even and 10 odd number.
  - b. Write a JAVA JSP Program to implement verification of a particular user login and display a welcome page.
9. Write a JAVA JSP Program to get student information through a HTML and create a JAVA Bean Class, populate Bean and display the same information through another JSP.
10. Write a JAVA JSP Program which uses <jsp:plugin> tag to run a applet.
11. Write a JAVA JSP Program which implements nested tags and also uses TagSupport Class.
12. An EJB application that demonstrates Session Bean.
13. An EJB application that demonstrates Entity Bean.
14. An EJB application that demonstrates MDB.

## Web Programming Laboratory

**Subject Code: 10MCA47**  
**Hours/Week : 03**  
**Total Hours : 42**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 50**

1. Develop and demonstrate a XHTML file that includes Javascript script for the following problems:
  - a) Input: A number n obtained using prompt  
Output: The first n Fibonacci numbers
  - b) Input: A number n obtained using prompt  
Output: A table of numbers from 1 to n and their squares using **alert**
2. a) Develop and demonstrate, using Javascript script, a XHTML document that collects the USN ( the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.  
b) Modify the above program to get the current semester also (restricted to be a number from 1 to 8)
3. a) Develop and demonstrate, using Javascript script, a XHTML document that contains three short paragraphs of text, stacked on top of each other, with only enough of each showing so that the mouse cursor can be placed over some part of them. When the cursor is placed over the exposed part of any paragraph, it should rise to the top to become completely visible.  
b) Modify the above document so that when a paragraph is moved from the top stacking position, it returns to its original position rather than to the bottom.
4. a) Design an XML document to store information about a student in an engineering college affiliated to VTU. The information must include USN, Name, Name of the College, Brach, Year of Joining, and e-mail id. Make up sample data for 3 students. Create a CSS style sheet and use it to display the document.  
b) Create an XSLT style sheet for one student element of the above document and use it to create a display of that element.
5. a) Write a Perl program to display various Server informations like Server Name, Server Software, Server protocol, CGI Revision etc.  
b) Write a Perl program to accept UNIX command from a HTML form and to display the output of the command executed.
6. a) Write a Perl program to accept the User Name and display a greeting message randomly chosen from a list of 4 greeting messages.  
b) Write a Perl program to keep track of the number of visitors visiting the web page and to display this count of visitors, with proper headings.
7. Write a Perl program to display a digital clock which displays the current time of the server.
8. Write a Perl program to insert name and age information entered by the user into a table created using MySQL and to display the current contents of this table.
9. Write a PHP program to store current date-time in a COOKIE and display the 'Last visited on' date-time on the web page upon reopening of the same page.
10. Write a PHP program to store page views count in SESSION, to increment the count on each refresh, and to show the count on web page.
11. Create a XHTML form with Name, Address Line 1, Address Line 2, and E-mail text fields. On submitting, store the values in MySQL table. Retrieve and display the data based on Name.
12. Build a Rails application to accept book information viz. Accession number, title, authors, edition and publisher from a web page and store the information in a database and to search for a book with the title specified by the user and to display the search results with proper headings.

**Note: In the examination *each* student picks one question from the lot of *all* 12 questions.**

## ALGORITHMS LABORATORY

**Subject Code: 10MCA48**  
**Hours/Week : 03**  
**Total Hours : 42**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 50**

**Design, develop and implement the specified algorithms for the following problems using C/C++ Language in LINUX / Windows environment.**

1. Sort a given set of elements using the Quicksort method and determine the time required to sort the elements. Repeat the experiment for different values of  $n$ , the number of elements in the list to be sorted and plot a graph of the time taken versus  $n$ .  
The elements can be read from a file or can be generated using the random number generator.
2. Using OpenMP, implement a parallelized Merge Sort algorithm to sort a given set of elements and determine the time required to sort the elements. Repeat the experiment for different values of  $n$ , the number of elements in the list to be sorted and plot a graph of the time taken versus  $n$ . The elements can be read from a file or can be generated using the random number generator.
3.
  - a. Obtain the Topological ordering of vertices in a given digraph.
  - b. Compute the transitive closure of a given directed graph using Warshall's algorithm.
4. Implement 0/1 Knapsack problem using Dynamic Programming.
5. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
6. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal's algorithm.
7.
  - a. Print all the nodes reachable from a given starting node in a digraph using BFS method.
  - b. Check whether a given graph is connected or not using DFS method.
8. Find a subset of a given set  $S = \{s_1, s_2, \dots, s_n\}$  of  $n$  positive integers whose sum is equal to a given positive integer  $d$ . For example, if  $S = \{1, 2, 5, 6, 8\}$  and  $d = 9$  there are two solutions  $\{1, 2, 6\}$  and  $\{1, 8\}$ . A suitable message is to be displayed if the given problem instance doesn't have a solution.
9. Implement any scheme to find the optimal solution for the Traveling Salesperson problem and then solve the same problem instance using any approximation algorithm and determine the error in the approximation.
10. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
11. Implement All-Pairs Shortest Paths Problem using Floyd's algorithm. Parallelize this algorithm, implement it using OpenMP and determine the speed-up achieved.
12. Implement N Queen's problem using Back Tracking.

**Note: In the examination *each* student picks one question from the lot of *all* 12 questions.**

## V SEMESTER

**Object-Oriented Modeling and Design Patterns**

**Subject Code: 10MCA51**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

**Introduction, Modeling Concepts, class Modeling****7 Hours**

What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history

Modeling as Design Technique: Modeling; abstraction; The three models.

Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.

**Advanced Class Modeling, State Modeling****6 Hours**

Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips.

State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.

**Advanced State Modeling, Interaction Modeling****6 Hours**

Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips.

Interaction Modeling: Use case models; Sequence models; Activity models.

Use case relationships; Procedural sequence models; Special constructs for activity models.

**Process Overview, System Conception, Domain Analysis****7 Hours**

Process Overview: Development stages; Development life cycle.

System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement.

Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis.

**Application Analysis, System Design****7 Hours**

Application Analysis: Application interaction model; Application class model; Application state model; Adding operations.

Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example.

**Class Design, Implementation Modeling, Legacy Systems****7 Hours**

Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example.

Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing.

Legacy Systems: Reverse engineering; Building the class models; Building the interaction model; Building the state model; Reverse engineering tips; Wrapping; Maintenance.

**Design Patterns, Idioms****12 Hours**

What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description

Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber.

Management Patterns: Command processor; View handler.

Idioms: Introduction; what can idioms provide? Idioms and style; Where to find idioms; Counted Pointer example

**Text Books:**

1. Michael Blaha, James Rumbaugh: Object-Oriented Modeling and Design with UML, 2<sup>nd</sup> Edition, Pearson Education, 2005.

(Chapters 1 to 17, 23)

2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2006.  
(Chapters 1, 3.5, 3.6, 4)

**Reference Books:**

1. Grady Booch et al: Object-Oriented Analysis and Design with Applications, 3<sup>rd</sup> Edition, Pearson, 2007.
2. Brahma Dathan, Sarnath Ramnath: Object-Oriented Analysis, Design, and Implementation, Universities Press, 2009.
3. Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado: UML 2 Toolkit, Wiley- Dreamtech India, 2004.
4. Simon Bennett, Steve McRobb and Ray Farmer: Object-Oriented Systems Analysis and Design Using UML, 2<sup>nd</sup> Edition, Tata McGraw-Hill, 2002.

## System Modeling and Simulation

**Subject Code: 10MCA52**

**Hours/Week: 04**

**Total Hours: 52**

**IA Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**8 Hours**

When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study.

The basics of Spreadsheet simulation, Simulation example: Simulation of queuing systems in a spreadsheet.

### **General Principles, Simulation Software**

**6 Hours**

Concepts in Discrete-Event Simulation: The Event-Scheduling / Time-Advance Algorithm, World Views, Manual simulation Using Event Scheduling; List processing.

Simulation in Java; Simulation in GPSS

### **Statistical Models in Simulation**

**6 Hours**

Review of terminology and concepts; Useful statistical models; Discrete distributions; Continuous distributions; Poisson process; Empirical distributions.

### **Queuing Models**

**6 Hours**

Characteristics of queuing systems; Queuing notation; Long-run measures of performance of queuing systems; Steady-state behavior of M/G/1 queue; Networks of queues; Rough-cut modeling: An illustration..

### **Random-Number Generation, Random-Variate Generation**

**8 Hours**

Properties of random numbers; Generation of pseudo-random numbers; Techniques for generating random numbers; Tests for Random Numbers

Random-Variate Generation: Inverse transform technique; Acceptance-Rejection technique; Special properties.

### **Input Modeling**

**6 Hours**

Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; Selecting input models without data; Multivariate and Time-Series input models.

### **Estimation of Absolute Performance**

**6 Hours**

Types of simulations with respect to output analysis; Stochastic nature of output data; Absolute measures of performance and their estimation; Output analysis for terminating simulations; Output analysis for steady-state simulations.

### **Verification, Calibration, and Validation; Optimization**

**6 Hours**

Model building, verification and validation; Verification of simulation models; Calibration and validation of models. Optimization via Simulation

### **Text Books:**

1. Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol: Discrete-Event System Simulation, 5<sup>th</sup> Edition, Pearson, 2010.  
(Listed topics only from Chapters 1 to 12)

### **Reference Books:**

1. Lawrence M. Leemis, Stephen K. Park: Discrete – Event Simulation: A First Course, Pearson / Prentice-Hall, 2006.
2. Averill M. Law: Simulation Modeling and Analysis, 4<sup>th</sup> Edition, Tata McGraw-Hill, 2007.



## Topics in Enterprise Architecture - II

**Subject Code: 10MCA53**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **The philosophy of .NET**

**6 Hours**

Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries ( aka Assemblies ), the Role of the Common Intermediate Language , The Role of .NET Type Metadata, The Role of the Assembly Manifest, Compiling CIL to Platform –Specific Instructions, Understanding the Common Type System, Intrinsic CTS Data Types, Understanding the Common Language Specification, Understanding the Common Language Runtime A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime

### **Building C# Applications**

**6 Hours**

The Role of the Command Line Compiler (csc.exe), Building C # Application using csc.exe Working with csc.exe Response Files, Generating Bug Reports , Remaining C# Compiler Options, The Command Line Debugger (cordbg.exe) Using the, Visual Studio .NET IDE, Other Key Aspects of the VS.NET IDE, C# “Preprocessor:” Directives, An Interesting Aside: The System. Environment Class

### **C# Language Fundamentals.**

**8 Hours**

The Anatomy of a Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System, Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C #, String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces.

### **Object- Oriented Programming with C#**

**6 Hours**

Forms Defining of the C# Class, Definition the “Default Public Interface” of a Type, Recapping the Pillars of OOP, The First Pillars: C#’s Encapsulation Services, Pseudo- Encapsulation: Creating Read-Only Fields, The Second Pillar: C#’s Inheritance Supports, keeping Family Secrets: The “Protected” Keyword, Nested Type Definitions, The Third Pillar: C #’s Polymorphic Support, Casting Between .

### **Exceptions and Object Lifetime**

**6 Hours**

Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handling, the System. Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception(System. System Exception), Custom Application-Level Exception(System. System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of “new”, The Basics of Garbage Collection,, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.

### **Interfaces and Collections**

**6 Hours**

Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects ( ICloneable), Building Comparable Objects ( I Comparable ), Exploring the system. Collections Namespace, Building a Custom Container (Retrofitting the Cars Type).

**Callback Interfaces, Delegates, and Events, Advanced Techniques****8 Hours**

Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, , Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using)Events.  
 The Advances Keywords of C#, A Catalog of C# Keywords Building a Custom Indexer, A Variation of the Cars Indexer Internal Representation of Type Indexer . Using C# Indexer from VB .NET. Overloading operators, The Internal Representation of Overloading Operators, interacting with Overload Operator from Overloaded- Operator- Challenged Languages, Creating Custom Conversion Routines, Defining Implicit Conversion Routines, The Internal Representations of Customs Conversion Routines

**Understanding .NET Assemblies.****6 Hours**

Problems with Classic COM Binaries, An Overview of .NET Assembly, Building a Simple File Test Assembly, A C#. Client Application, A Visual Basic .NET Client Application, Cross Language Inheritance, Exploring the CarLibrary's, Manifest, Exploring the CarLibrary's Types, Building the Multifile Assembly, Using Assembly, Understanding Private Assemblies, Probing for Private Assemblies (The Basics), Private A Assemblies XML Configurations Files, Probing for Private Assemblies ( The Details), Understanding Shared Assembly, Understanding Shared Names, Building a Shared Assembly, Understanding Delay Signing, Installing/Removing Shared Assembly, Using a Shared Assembly

**Text Books:**

1. Andrew Troelsen: Pro C# with .NET 3.0, Special Edition, Dream tech Press, India, 2007.  
 Chapters: 1 to 11 (up to pp.369)
2. E. Balagurusamy: Programming in C#, , 5<sup>th</sup> Reprint, Tata McGraw Hill, 2004.  
 (Programming Examples 3.7, 3.10, 5.5, 6.1, 7.2, 7.4, 7.5, 7.6, 8.1, 8.2, 8.3, 8.5, 8.7, 8.8, 9.1, 9.2, 9.3, 9.4, 10.2, 10.4, 11.2, 11.4, 12.1, 12.4, 12.5, 12.6, 13.1, 13.2, 13.3, 13.6, 14.1, 14.2, 14.4, 15.2, 15.3, 16.1, 16.2, 16.3, 18.3, 18.5.18.6)

**Reference Books:**

1. Tom Archer: Inside C#, WP Publishers, 2001.
2. Herbert Schildt: C# The Complete Reference, Tata McGraw Hill, 2004.

## Information Retrieval

**Subject Code: 10MCA541**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Hours: 3**  
**Exam Marks: 100**

### **Introduction, Retrieval Strategies**

**13 Hours**

Introduction; Retrieval Strategies: Vector Space Model; Probabilistic Retrieval strategies, Some More Retrieval Strategies: Language Models; Inference Networks; Extended Boolean Retrieval; Latent Semantic Indexing; Neural Networks; Genetic Algorithms; Fuzzy Set Retrieval.

### **Retrieval Utilities**

**7 Hours**

Relevance feedback; Clustering; Passage-Based Retrieval; N-Grams; Regression Analysis; Thesauri; Semantic Networks; Parsing.

### **Indexing and Searching**

**6 Hours**

Introduction; Inverted Files; Other indices for text; Boolean queries; Sequential searching; Pattern matching; Structural queries; Compression.

### **Cross-Language Information Retrieval and Efficiency**

**6 Hours**

Introduction; Crossing the language barrier; Cross-Language retrieval strategies; Cross language utilities. Duplicate Document Detection.

### **Integrating Structured Data and Text**

**6 Hours**

Review of the relational model; A historical progression; Information retrieval as a relational application; Semi-structured search using a relational schema; Multi-dimensional data model.

### **Parallel Information Retrieval, Distributed Information Retrieval**

**7 Hours**

Parallel text scanning; Parallel indexing; Clustering and classification; Large parallel systems; A theoretic model of distributed information retrieval; Web search; Result fusion; Peer-to-Peer information systems; Other architectures.

### **Multimedia IR**

**7 Hours**

Introduction; data modeling; Query languages; Spatial access methods; A general multimedia indexing approach; One-dimensional time series; Two-dimensional color images; Automatic picture extraction.

### **Text Books:**

1. David A. Grossman, Ophir Frieder: Information Retrieval Algorithms and Heuristics, 2<sup>nd</sup> Edition, Springer, 2004.  
(Chapters 1, 2, 3, 4, 5, 6, 7, 8)
2. Ricardo Baeza-Yates, Berthier Ribeiro-Neto: Modern Information Retrieval, Pearson Education, 1999  
(Chapters 8, 11, 12)

### **Reference Books:**

1. William B. Frakes, Ricardo Baeza-Yates (Editors): Information Retrieval Data Structures & Algorithms, Prentice Hall PTR, 1992.

## Data Warehousing and Data Mining

**Subject Code: 10MCA542**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

**Data Warehousing** **6 Hours**  
 Introduction, Operational Data Stores (ODS), Extraction Transformation Loading (ETL), Data Warehouses Design Issues, Guidelines for Data Warehouse Implementation, Data Warehouse Metadata

**Online Analytical Processing (OLAP)** **6 Hours**  
 Introduction, Characteristics of OLAP systems, Multidimensional view and Data cube, Data Cube Implementations, Data Cube operations, Implementation of OLAP and overview on OLAP Softwares.

**Data Mining** **6 Hours**  
 Introduction, Challenges, Data Mining Tasks, Types of Data, Data Preprocessing, Measures of Similarity and Dissimilarity, Data Mining Applications

**Association Analysis: Basic Concepts and Algorithms** **8 Hours**  
 Frequent Itemset Generation, Rule Generation, Compact Representation of Frequent Itemsets, Alternative methods for generating Frequent Itemsets, FP Growth Algorithm, Evaluation of Association Patterns

**Classification** **12 Hours**  
 Basics, General approach to solve classification problem, Decision Trees, Rule Based Classifiers, Nearest Neighbor Classifiers.  
 Bayesian Classifiers, Estimating Predictive accuracy of classification methods, Improving accuracy of clarification methods, Evaluation criteria for classification methods, Multiclass Problem.

**Clustering Techniques** **8 Hours**  
 Overview, Features of cluster analysis, Types of Data and Computing Distance, Types of Cluster Analysis Methods, Partitional Methods, Hierarchical Methods, Density Based Methods, Quality and Validity of Cluster Analysis

**Web Mining** **6 Hours**  
 Introduction, Web content mining, Text Mining, Unstructured Text, Text clustering, Mining Spatial and Temporal Databases.

**Text Books:**

1. Pang-Ning Tan, Michael Steinbach, Vipin Kumar: Introduction to Data Mining, Addison-Wesley, 2005.
2. G. K. Gupta: Introduction to Data Mining with Case Studies, 3<sup>rd</sup> Edition, PHI, New Delhi, 2009.

**Reference Books:**

1. Arun K Pujari: Data Mining Techniques University Press, 2<sup>nd</sup> Edition, 2009.
2. Jiawei Han and Micheline Kamber: Data Mining - Concepts and Techniques, 2<sup>nd</sup> Edition, Morgan Kaufmann Publisher, 2006.
3. Alex Berson and Stephen J. Smith: Data Warehousing, Data Mining, and OLAP Computing Mc GrawHill Publisher, 1997.

## Supply Chain Management

**Subject Code: 10MCA543**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

**Introduction to Supply Chain, Performance of Supply Chain** **6 Hours**  
 What is a Supply Chain; Decision phases in a supply Chain; Process view of a Supply Chain; The importance of Supply Chain Flows; Examples of Supply Chains.  
 Competitive and Supply Chain strategies; Achieving strategic fit; Expanding strategic scope.

**Supply Chain drivers and Obstacles, Designing Distribution Network** **6 Hours**  
 Drivers of Supply Chain Performance; A framework for structuring drivers; Facilities, Inventory, Transportation, and Information; Obstacles to achieve strategic fit  
 The role of distribution in the Supply Chain; factors influencing distribution network design; Design options for a distribution network; the value of distributors in the Supply Chain; Distribution Networks in practice.

**Network Design** **7 Hours**  
 The role of network design in the Supply Chain; Factors influencing Network design Decisions; A framework for Network Design Decisions; Models for facility Location and Capacity Allocation; making Network Design decisions in practice.  
 The impact of uncertainty on Network design; Discounted cash flow analysis; Representations of uncertainty; Evaluating Network Design decisions using Decision Trees; Making Supply Chain decisions under uncertainty in practice.

**Demand Forecasting, Aggregate Planning** **7 Hours**  
 The role of forecasting in a Supply Chain; Characteristics of forecast; Components of a forecast and forecasting methods; Basic approach of Demand forecasting; Time series forecasting methods; Measures of forecast errors; The role of aggregate planning in a supply Chain; The aggregate planning problem; Aggregate planning strategies.

**Inventory Management** **6 Hours**  
 The role of cycle inventory in a supply Chain; Economies of scale to exploit fixed costs, quantity discounts; Short-term discounting; Managing multi-echelon cycle inventory; Estimating cycle inventory related costs in practice.

**Transportation** **7 Hours**  
 The role of transportation in the Supply Chain; Factors affecting transportation decisions; Modes of transportation and their performance characteristics; Design options for a transportation network; Trade-offs in transportation design; Tailored transportation; Routing and scheduling in transportation; Making transportation decisions in practice.

**Pricing and Revenue Management, Coordination** **7 Hours**  
 The role of revenue management in Supply Chain; revenue management for multiple customer segments, perishable assets, seasonal demand, and bulk and spot contracts; Using revenue management in practice  
 Lack of Supply Chain coordination and Bullwhip effect; Effect of lack of coordination on performance; Obstacles to coordination in the Supply Chain; managerial levers to achieve coordination; Building strategic partnerships and trust within a supply Chain; Achieving coordination in practice.

**IT, Internet and Supply Chain** **6 Hours**  
 The role of IT in the Supply Chain; The Supply Chain IT framework; CRM; Internal SCM; Supplier Relationship Management; The transaction management foundation; The future if IT in SCM; Supply Chain It in practice.  
 The role of E-Business in Supply Chain; The E-Business framework; The B2B addition to the E-Business framework; E-Business in practice

**Text Books:**

1. Sunil Chopra, Pter Meindl: Supply Chain Management Strategy, Planning, and Operation, 2<sup>nd</sup> Edition, Prentice-Hall of India, 2004.  
 (Chapters 1, 2, 4, 4, 5, 6, 7, 8.1 to 8.3, 10, 14, 15, 16, 17, 18)

**Reference Books:**

1. David Simchi-Levi, Philp Kaminky, Edith Simchi-Levi: Designing and Managing The Supply Chain Concepts, Strategies & Case Studies, 3<sup>rd</sup> Edition, Tata McGraw Hill, 2003.
2. R.P. Mohanty, S.G. Deshmukh: Supply Chain Management Theories & Practices, Bizmantra, 2005.
3. Rahul V. Altekar: Supply Chain Management Concepts and Cases, PHI, 2005.
4. M Martin Christopher: Logistics and Supply Chain Management, 2<sup>nd</sup> Edition, Pearson Education, 1998.

## Network Management

**Subject Code: 10MCA544**

**Hours/Week : 04**

**Total Hours : 52**

**IA Marks: 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**7 Hours**

Analogy of Telephone Network Management, Data and Telecommunication Network Distributed computing Environments, TCP/IP-Based Networks: The Internet and Intranets, Communications Protocols and Standards- Communication Architectures, Protocol Layers and Services; Case Histories of Networking and Management – The Importance of topology, Filtering Does Not Reduce Load on Node, Some Common Network Problems; Challenges of Information Technology Managers, Network Management: Goals, Organization, and Functions- Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management.

### **Basic Foundations: Standards, Models, and Language**

**6 Hours**

Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees, Managed Object Perspectives, Communication Model; ASN.1- Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1 from ISO 8824; Encoding Structure; Macros, Functional Model.

### **SNMPv1 Network Management**

**13 Hours**

Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview.

The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base.

The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, Functional Model

### **SNMP Management – RMON**

**6 Hours**

Remote Monitoring, RMON SMI and MIB, RMON1- RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups, RMON2 – The RMON2 Management Information Base, RMON2 Conformance Specifications; ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON.

### **Broadband Network Management: ATM Networks**

**6 Hours**

Broadband Networks and Services, ATM Technology – Virtual Path-Virtual Circuit, TM Packet Size, Integrated Service, SONET, ATM LAN Emulation, Virtual LAN; ATM Network Management – The ATM Network Reference Model, The Integrated Local Management Interface, The ATM Management Information Base, The Role of SNMP and ILMI in ATM Management, M1 Interface: Management of ATM Network Element, M2 Interface: Management of Private Networks, M3 Interface: Customer Network Management of Public Networks, M4 Interface: Public Network Management, Management of LAN Emulation, ATM Digital Exchange Interface Management.

### **Broadband Network Management**

**6 Hours**

Broadband Access Networks and Technologies – Broadband Access Networks, broadband Access Technology; HFCT Technology – The Broadband LAN, The Cable Modem, The Cable Modem Termination System, The HFC Plant, The RF Spectrum for Cable Modem; Data Over Cable Reference Architecture; HFC Management – Cable Modem and CMTS Management, HFC Link Management, RF Spectrum Management, DSL Technology; Asymmetric Digital Subscriber Line Technology – Role of the ADSL Access Network in an Overall Network, ADSL Architecture, ADSL Channeling Schemes, ADSL Encoding Schemes; ADSL Management – ADSL Network Management Elements, ADSL Configuration Management, ADSL Fault Management, ADSL Performance Management, SNMP-Based ADSL Line MIB, MIB Integration with Interfaces Groups in MIB-2, ADSL Configuration Profiles.

### **Network Management Applications**

**8 Hours**

Configuration Management- Network Provisioning, Inventory Management, Network Topology, Fault Management- Fault Detection, Fault Location and Isolation Techniques, Performance Management – Performance Metrics, Data Monitoring, Problem Isolation, Performance Statistics; Event Correlation Techniques – Rule-Based Reasoning, Model-Based Reasoning, Case-Based Reasoning, Codebook correlation Model, State Transition Graph Model, Finite State Machine Model, Security Management –

\*\*\*

Policies and Procedures, Security Breaches and the Resources Needed to Prevent Them, Firewalls, Cryptography, Authentication and Authorization, Client/Server Authentication Systems, Messages Transfer Security, Protection of Networks from Virus Attacks, Accounting Management, Report Management, Policy-Based Management, Service Level Management.

**Text Books:**

1. Mani Subramanian: Network Management- Principles and Practice, Pearson Education, 2003.

**Reference Books:**

1. J. Richard Burke: Network management Concepts and Practices: a Hands-On Approach, PHI, 2008.



## Compiler Design

**Subject Code: 10MCA545**

**Hours/ Week: 04**

**Total Hours: 52**

**IA Marks: 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction, Lexical analysis**

**8 Hours**

Language processors; The structure of a Compilers; The evolution of programming languages; The science of building a compiler; Applications of Compiler technology; Programming language basics; Lexical analysis: The Role of Lexical Analyzer; Input Buffering; Specifications of Tokens; Recognition of Tokens.

### **Syntax Analysis - 1**

**6 Hours**

Introduction; Context-free Grammars; Writing a Grammar; Top-down Parsing

### **Syntax Analysis – 2**

**6 Hours**

Bottom-up Parsing; Introduction to LR Parsing; Simple LR.

### **Syntax Analysis – 3**

**6 Hours**

More powerful LR parsers; Using ambiguous grammars; Parser Generators.

### **Syntax-Directed Translation**

**6 Hours**

Syntax-Directed definitions; Evaluation order for SDDs; Applications of Syntax-directed translation; Syntax-directed translation schemes

### **Intermediate Code Generation**

**8 Hours**

Variants of syntax trees; Three-address code; Types and declarations; Translation of expressions; Type checking; Control flow; Back patching; Switch statements; Intermediate code for procedures.

### **Run-Time Environments**

**6 Hours**

Storage Organization; Stack allocation of space; Access to non-local data on the stack; Heap management; Introduction to garbage collection

### **Code Generation**

**6 Hours**

Issues in the design of Code Generator; The Target language; Addresses in the target code; Basic blocks and Flow graphs; Optimization of basic blocks; A Simple Code Generator.

### **Text Books:**

1. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman: Compilers- Principles, Techniques and Tools, 2<sup>nd</sup> Edition, Addison-Wesley, 2007.  
(Chapters 1, 3.1 to 3.4, 4, 5.1 to 5.4, 6, 7.1 to 7.5, 8.1 to 8.6)

### **Reference Books:**

1. Charles N. Fischer, Richard J. leBlanc, Jr.: Crafting a Compiler with C, Pearson Education, 1991.
2. Andrew W Apple: Modern Compiler Implementation in C, Cambridge University Press, 1997.
3. Kenneth C Loudon: Compiler Construction Principles & Practice, Thomson Education, 1997.

## Software Architectures

**Subject Code: 10MCA546**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Introduction**

**6 Hours**

The Architecture Business Cycle: Where do architectures come from? Software processes and the architecture business cycle; What makes a “good” architecture? What software architecture is and what it is not; Other points of view; Architectural patterns, reference models and reference architectures; Importance of software architecture; Architectural structures and views.

### **Architectural Styles and Case Studies**

**7 Hours**

Architectural styles; Pipes and filters; Data abstraction and object-oriented organization; Event-based, implicit invocation; Layered systems; Repositories; Interpreters; Process control; Other familiar architectures; Heterogeneous architectures.

Case Studies: Keyword in Context; Instrumentation software; Mobile robotics; Cruise control; Three vignettes in mixed style.

### **Quality**

**6 Hours**

Functionality and architecture; Architecture and quality attributes; System quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities.

Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics; Relationship of tactics to architectural patterns; Architectural patterns and styles.

### **Architectural Patterns – 1**

**7 Hours**

Introduction; From mud to structure: Layers, Pipes and Filters, Blackboard.

### **Architectural Patterns – 2**

**7 Hours**

Distributed Systems: Broker; Interactive Systems: MVC, Presentation-Abstraction-Control

### **Architectural Patterns – 3**

**6 Hours**

Adaptable Systems: Microkernel; Reflection.

### **Some Design Patterns**

**6 Hours**

Structural decomposition: Whole – Part; Organization of work: Master – Slave; Access Control: Proxy.

### **Designing and Documenting Software Architecture**

**7 Hours**

Architecture in the life cycle; Designing the architecture; Forming the team structure; Creating a skeletal system.

Uses of architectural documentation; Views; Choosing the relevant views; Documenting a view; Documentation across views.

### **Text Books:**

1. Len Bass, Paul Clements, Rick Kazman: Software Architecture in Practice, 2<sup>nd</sup> Edition, Pearson Education, 2003.  
(Chapters 1, 2, 4, 5, 7, 9)
2. Frank Buschmann, Regine Meunier, Hans Rohnert, Peter Sommerlad, Michael Stal: Pattern-Oriented Software Architecture, A System of Patterns, Volume 1, John Wiley and Sons, 2006.  
(Chapters 2, 3.1 to 3.4)
3. Mary Shaw and David Garlan: Software Architecture- Perspectives on an Emerging Discipline, Prentice Hall of India / Pearson Education, 2007.  
(Chapters 1.1, 2, 3)

### **Reference Books:**

1. E. Gamma, R. Helm, R. Johnson, J. Vlissides: Design Patterns- Elements of Reusable Object-Oriented Software, Addison-Wesley, 1995.
2. **Web site for Patterns:** <http://www.hillside.net/patterns/>

## Information and Network Security

**Subject Code: 10MCA551**

**Hours/Week : 04**

**Total Hours : 52**

**I.A. Marks : 50**

**Exam Hours: 03**

**Exam Marks: 100**

**Planning for Security** **6 Hours**  
Introduction; Information Security Policy, Standards, and Practices; The Information Security Blue Print; Contingency plan and a model for contingency plan.

**Security Technology** **12 Hours**  
Introduction; Physical design; Firewalls; Protecting Remote Connections  
Introduction; Intrusion Detection Systems (IDS); Honey Pots, Honey Nets, and Padded cell systems; Scanning and Analysis Tools

**Cryptography** **8 Hours**  
Introduction; A short History of Cryptography; Principles of Cryptography; Cryptography Tools; Attacks on Cryptosystems.

**Introduction to Network Security, Authentication Applications** **8 Hours**  
Attacks, services, and Mechanisms; Security Attacks; Security Services; A model for Internetwork Security; Internet Standards and RFCs  
Kerberos, X.509 Directory Authentication Service.

**Electronic Mail Security** **6 Hours**  
Pretty Good Privacy (PGP); S/MIME

**IP Security** **6 Hours**  
IP Security Overview; IP Security Architecture; Authentication Header; Encapsulating Security Payload; Combining Security Associations; Key Management.

**Web Security** **6 Hours**  
Web security requirements; Secure Socket layer (SSL) and Transport layer Security (TLS); Secure Electronic Transaction (SET)

**Text Books:**

1. Michael E. Whitman and Herbert J. Mattord: Principles of Information Security, 2<sup>nd</sup> Edition, Thomson, 2005.  
(Chapters 5, 6, 7, 8; Exclude the topics not mentioned in the syllabus)
2. William Stallings: Network Security Essentials: Applications and Standards, Pearson Education, 2000.  
(Chapters: 1, 4, 5, 6, 7, 8)

**Reference Book:**

1. Behrouz A. Forouzan: Cryptography and Network Security, Special Indian Edition, Tata McGraw-Hill, 2007.

## Software Testing

**Subject Code: 10MCA552**

**Hours/Week: 4**

**Total Hours: 52**

**I.A. Marks: 50**

**Exam Hours: 3**

**Exam Marks: 100**

**A Perspective on Testing, Examples:**

**6 Hours**

Basic definitions, Test cases, Insights from a Venn diagram, Identifying test cases, Error and fault taxonomies, Levels of testing. Examples: Generalized pseudocode, The triangle problem, The NextDate function, The commission problem, The SATM (Simple Automatic Teller Machine) problem, The currency converter, Saturn windshield wiper.

**Boundary Value Testing, Equivalence Class Testing, Decision Table-Based Testing:**

**7 Hours**

Boundary value analysis, Robustness testing, Worst-case testing, Special value testing, Examples, Random testing, Equivalence classes, Equivalence test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations. Decision tables, Test cases for the triangle problem, NextDate function, and the commission problem, Guidelines and observations.

**Path Testing, Data Flow Testing:**

**7 Hours**

DD paths, Test coverage metrics, Basis path testing, guidelines and observations. Definition-Use testing, Slice-based testing, Guidelines and observations.

**Levels of Testing, Integration Testing:**

**6 Hours**

Traditional view of testing levels, Alternative life-cycle models, The SATM system, Separating integration and system testing. A closer look at the SATM system, Decomposition-based, call graph-based, Path-based integrations.

**System Testing, Interaction Testing:**

**7 Hours**

Threads, Basic concepts for requirements specification, Finding threads, Structural strategies and functional strategies for thread testing, SATM test threads, System testing guidelines, ASF (Atomic System Functions) testing example. Context of interaction, A taxonomy of interactions, Interaction, composition, and determinism, Client/Server Testing.

**Process Framework:**

**7 Hours**

Validation and verification, Degrees of freedom, Varieties of software. Basic principles: Sensitivity, redundancy, restriction, partition, visibility, Feedback. The quality process, Planning and monitoring, Quality goals, Dependability properties, Analysis, Testing, Improving the process, Organizational factors.

**Fault-Based Testing, Test Execution:**

**6 Hours**

Overview, Assumptions in fault-based testing, Mutation analysis, Fault-based adequacy criteria, Variations on mutation analysis. Test Execution: Overview, from test case specifications to test cases, Scaffolding, Generic versus specific scaffolding, Test oracles, Self-checks as oracles, Capture and replay.

**Planning and Monitoring the Process, Documenting Analysis and Test:**

**6 Hours**

Quality and process, Test and analysis strategies and plans, Risk planning, Monitoring the process, Improving the process, The quality team, Organizing documents, Test strategy document, Analysis and test plan, Test design specifications documents, Test and analysis reports.

**TEXT BOOKS:**

1. Paul C. Jorgensen: Software Testing, A Craftsman's Approach, 3<sup>rd</sup> Edition, Auerbach Publications, 2008.  
(Listed topics only from Chapters 1, 2, 5, 6, 7, 9, 10, 12, 13, 14, 15)
2. Mauro Pezze, Michal Young: Software Testing and Analysis – Process, Principles and Techniques, Wiley India, 2008.  
(Listed topics only from Chapters 2, 3, 4, 16, 17, 20, 24)

**REFERENCE BOOKS:**

1. Aditya P Mathur: Foundations of Software Testing, Pearson, 2008.
2. Srinivasan Desikan, Gopalaswamy Ramesh: Software testing Principles and Practices, 2<sup>nd</sup> Edition, Pearson, 2007.
3. Brian Marrick: The Craft of Software Testing, Pearson, 1995.

## Services Oriented Architecture

**Subject Code: 10MCA553**  
**Hours/Week: 4**  
**Total Hours: 52**

**I.A. Marks: 50**  
**Exam Hours: 3**  
**Exam Marks: 100**

### **Introduction o SOA, Evolution of SOA**

**7 Hours**

Fundamental SOA; Common Characteristics of contemporary SOA; Common tangible benefits of SOA; An SOA timeline (from XML to Web services to SOA); The continuing evolution of SOA (Standards organizations and Contributing vendors); The roots of SOA (comparing SOA to Past architectures).

### **Web Services and Primitive SOA**

**6 Hours**

The Web services framework; Services (as Web services); Service descriptions (with WSDL); Messaging (with SOAP).

### **Web Services and Contemporary SOA**

**13 Hours**

Message exchange patterns; Service activity; Coordination; Atomic Transactions; Business activities; Orchestration; Choreography. Addressing; Reliable messaging; Correlation; Polices; Metadata exchange; Security; Notification and eventing

### **Principles of Service – Orientation**

**7 Hours**

Services-orientation and the enterprise; Anatomy of a service-oriented architecture; Common Principles of Service-orientation; How service orientation principles inter-relate; Service-orientation and object-orientation; Native Web service support for service-orientation principles.

### **Service Layers**

**6 Hours**

Service-orientation and contemporary SOA; Service layer abstraction; Application service layer, Business service layer, Orchestration service layer; Agnostic services; Service layer configuration scenarios

### **Business Process Design**

**7 Hours**

WS-BPEL language basics; WS-Coordination overview; Service-oriented business process design; WS-addressing language basics; WS-ReliableMessaging language basics

### **SOA Platforms**

**6 Hours**

SOA platform basics; SOA support in J2EE; SOA support in .NET; Integration considerations

### **Text Books:**

1. Thomas Erl: Service-Oriented Architecture – Concepts, Technology, and Design, Pearson Education, 2005.

### **Reference Books:**

1. Eric Newcomer, Greg Lomow: Understanding SOA with Web Services, Pearson education, 2005.

## Wireless Networks and Mobile Computing

**Subject Code: 10MCA554**

**Hours/Week: 04**

**Total Hours: 52**

**IA Marks: 50**

**Exam Hours: 03**

**Exam Marks: 100**

### **Mobile Computing Architecture:**

**6 Hours**

Types of Networks, Architecture for Mobile Computing, 3-tier Architecture, Design Considerations for Mobile Computing,

### **Wireless Networks – 1: GSM and SMS**

**7 Hours**

Global Systems for Mobile Communication ( GSM and Short Service Messages ( SMS): GSM Architecture, Entities, Call routing in GSM, PLMN Interface, GSM Addresses and Identities, Network Aspects in GSM, Mobility Management, GSM Frequency allocation.

Introduction to SMS, SMS Architecture, SM MT, SM MO, SMS as Information bearer, applications

### **Wireless Networks – 2: GPRS**

**6 Hours**

GPRS and Packet Data Network, GPRS Network Architecture, GPRS Network Operations, Data Services in GPRS, Applications for GPRS, Billing and Charging in GPRS

### **Wireless Networks – 3: CDMA, 3G and WiMAX**

**7 Hours**

Spread Spectrum technology, IS-95, CDMA versus GSM, Wireless Data, Third Generation Networks, Applications on 3G, Introduction to WiMAX.

### **Mobile Client**

**6 Hours**

Moving beyond desktop, Mobile handset overview, Mobile phones and their features, PDA, Design Constraints in applications for handheld devices.

Mobile IP: Introduction, discovery, Registration, Tunneling, Cellular IP, Mobile IP with IPv6

### **Mobile OS and Computing Environment**

**7 Hours**

Smart Client Architecture, The Client: User Interface, Data Storage, Performance, Data Synchronization, Messaging. The Server: Data Synchronization, Enterprise Data Source, Messaging.

Mobile Operating Systems: WinCE, Palm OS, Symbian OS, Linux, Proprietary OS Client Development: The development process, Need analysis phase, Design phase, Implementation and Testing phase, Deployment phase, Development Tools, Device Emulators.

### **Building, Mobile Internet Applications**

**6 Hours**

Thin client: Architecture, the client, Middleware, messaging Servers, Processing a Wireless request, Wireless Applications Protocol (WAP) Overview,

Wireless Languages: Markup Languages, HDML, WML, HTML, cHTML, XHTML, VoiceXML.

### **J2ME**

**7 Hours**

Introduction, CDC, CLDC, MIDP; Programming for CLDC, MIDlet model, Provisioning, MIDlet life-cycle, Creating new application, MIDlet event handling, GUI in MIDP, Low level GUI Components, Multimedia APIs; Communication in MIDP, Security Considerations in MIDP.

### **Text Books:**

1. Dr. Ashok Talukder, Ms Roopa Yavagal, Mr. Hasan Ahmed: Mobile Computing, Technology, Applications and Service Creation, 2d Edition, Tata McGraw Hill, 2010.
2. Martyn Mallik: Mobile and Wireless Design Essentials, Wiley, 2003.

### **Reference Books:**

1. Raj kamal: Mobile Computing, Oxford University Press, 2007.
2. Iti Saha Misra: Wireless Communications and Networks, 3G and Beyond, Tata McGraw Hill, 2009.

## Storage Area Networks

**Subject Code: 10MCA555**  
**Hours/Week : 04**  
**Total Hours : 52**

**I.A. Marks : 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **Introduction to Information Storage and Management, Storage System Environment 7 Hours**

Information Storage, Evolution of Storage Technology and Architecture, Data Center Infrastructure, Key Challenges in Managing Information, Information Lifecycle  
 Components of Storage System Environment, Disk Drive Components, Disk Drive Performance, Fundamental Laws Governing Disk Performance, Logical Components of the Host, Application Requirements and Disk Performance.

### **Data Protection, Intelligent Storage system 6 Hours**

Implementation of RAID, RAID Array Components, RAID Levels, RAID Comparison, RAID Impact on Disk Performance, Hot Spares  
 Components of an Intelligent Storage System, Intelligent Storage Array

### **Direct-Attached Storage, SCSI, and Storage Area Networks 7 Hours**

Types of DAS, DAS Benefits and Limitations, Disk Drive Interfaces, Introduction to Parallel SCSI, Overview of Fibre Channel, The SAN and Its Evolution, Components of SAN, FC Connectivity, Fibre Channel Ports, Fibre Channel Architecture, Zoning, Fibre Channel Login Types, FC Topologies.

### **NAS, IP SAN 6 Hours**

General – Purpose Service vs. NAS Devices, Benefits of NAS, NAS File I / O, Components of NAS, NAS Implementations, NAS File-Sharing Protocols, NAS I/O Operations, Factors Affecting NAS Performance and Availability. iSCSI, FCIP.

### **Content-Addressed Storage, Storage Virtualization 6 Hours**

Fixed Content and Archives, Types of Archive, Features and Benefits of CAS, CAS Architecture, Object Storage and Retrieval in CAS, CAS Examples.  
 Forms of Virtualization, SNIA Storage Virtualization Taxonomy, Storage Virtualizations Configurations, Storage Virtualization Challenges, Types of Storage Virtualization,

### **Business Continuity, Backup and Recovery 6 Hours**

Information Availability, BC Terminology, BC Planning Lifecycle, Failure Analysis, Business Impact Analysis, BC Technology Solutions.  
 Backup Purpose, Backup Considerations, Backup Granularity, Recovery Considerations, Backup Methods, Backup Process, Backup and restore Operations, Backup Topologies, Backup in NAS Environments, Backup Technologies.

### **Local Replication, Remote Replication 7 Hours**

Source and Target, Uses of Local Replicas, Data Consistency, Local Replication Technologies, Restore and Restart Considerations, Creating Multiple Replicas, Management Interface, Modes of Remote Replication, Remote Replication Technologies, Network Infrastructure.

### **Securing the Storage Infrastructure, Managing the Storage Infrastructure 7 Hours**

Storage Security Framework, Risk Triad, Storage Security Domains, Security Implementations in Storage Networking  
 Monitoring the Storage Infrastructure, Storage Management Activities, Storage Infrastructure Management Challenges, Developing an Ideal Solution.

#### **Text Books:**

1. G. Somasundaram, Alok Shrivastava (Editors): Information Storage and Management, EMC Education Services, Wiley- India, 2009.

#### **Reference Books:**

1. Ulf Troppens, Rainer Erkens and Wolfgang Muller: Storage Networks Explained, Wiley India, 2003.
2. Rebert Spalding: Storage Networks, The Complete Reference, Tata McGraw Hill, 2003.

3. Richard Barker and Paul Massiglia: Storage Area Networks Essentials A Complete Guide to Understanding and Implementing SANs, Wiley India, 2002.



## Web 2.0 and Rich Internet Applications

**Subject Code: 10MCA556**  
**Hours/ Week: 04**  
**Total Hours: 52**

**IA Marks: 50**  
**Exam Hours: 03**  
**Exam Marks: 100**

### **Introduction, Ajax - 1**

**6 Hours**

Web 2.0 and Rich Internet Applications, Overview of Ajax, Examples of usage of Ajax: Updating web page text, Chatting in real time, Dragging and dropping, Downloading images.

Creating Ajax Applications: An example, Analysis of example ajax.html, Creating the JavaScript, Creating and opening the XMLHttpRequest object, Data download, Displaying the fetched data, Connecting to the server, Adding Server-side programming, Sending data to the server using GET and POST, Using Ajax together with XML.

### **Ajax – 2**

**7 Hours**

Handling multiple XMLHttpRequest objects in the same page, Using two XMLHttpRequest objects, Using an array of XMLHttpRequest objects, Using inner functions, Downloading JavaScript, connecting to Google Suggest, Creating google.php, Downloading from other domains with Ajax, HTML header request and Ajax, Defeating caching, Examples.

Building XML and working with XML in JavaScript, Getting the document element, Accessing any XML element, Handling whitespace in Firefox, Handling cross-browser whitespace, Accessing XML data directly, Validating XML, Further examples of Rich Internet Applications with Ajax.

### **Ajax – 3**

**6 Hours**

Drawing user's attention to downloaded text, Styling text, colors and background using CSS, Setting element location in the web pages, Setting the stacking order of web page elements, Further examples of using Ajax. Displaying all the data in an HTML form, Working with PHP server variables, Getting the data in to array format, Wrapping applications in to a single PHP page, Validating input from the user, Validating integers and text, DOM, Appending new elements to a web page using the DOM and Ajax, Replacing elements using the DOM, Handling timeouts in Ajax, Downloading images with Ajax, Example programs.

### **Flex - 1**

**7 Hours**

Introduction: Understanding Flex Application Technologies, Using Flex Elements, Working with Data Services (Loading Data at Runtime), The Differences between Traditional and Flex Web Applications, Understanding How Flex Applications Work, Understanding Flex and Flash Authoring.

Building Applications with the Flex Framework: Using Flex Tool Sets, Creating Projects, Building Applications, Deploying Applications

Framework Fundamentals: Understanding How Flex Applications Are Structured, Loading and Initializing Flex Applications, Understanding the Component Life Cycles, Loading One Flex Application into Another Flex Application, Differentiating Between Flash Player and the Flex Framework, Caching the Framework, Understanding Application Domains, Localization, Managing Layout: Flex Layout Overview, Making Fluid Interfaces, Putting It All Together.

### **Flex – 2**

**7 Hours**

MXML: Understanding MXML Syntax and Structure, Making MXML Interactive

Working with UI Components: Understanding UI Components, Buttons, Value Selectors, Text Components, List-Based Controls, Pop-Up Controls, Navigators, Control Bars

Customizing Application Appearance: Using Styles, Skinning components, Customizing the preloader, Themes, Runtime CSS

### **Flex - 3**

**6 Hours**

ActionScript: Using ActionScript, MXML and ActionScript Correlations, Understanding ActionScript Syntax, Variables and Properties, Inheritance, Interfaces, Handling Events, Error Handling, Using XML

### **Flex - 4**

**7 Hours**

Managing State: Creating States, Applying States, Defining States, Adding and Removing Components, Setting Properties, Setting Styles, Setting Event Handlers, Using Action Scripts to Define States, Managing Object Creation Policies, Handling State Events, Understanding State Life Cycles, When To Use States.

Using Effects and Transitions: Using Effects, Creating Custom Effects, Using Transitions, Creating Custom Transitions.

**Flex - 5****6 Hours**

Working with Data: Using Data Models, Data Binding, Enabling Data Binding for Custom Classes, Data Binding Examples, Building data binding proxies.

Validating and Formatting Data: Validating user input, Formatting Data.

**Text Books:**

1. Steven Holzner: Ajax: A Beginner's Guide, Tata McGraw Hill, 2009.  
(Listed topics from Chapters 3, 4, 6, 7, 11, 12)
2. Chafic Kazon and Joey Lott: Programming Flex 3, O'Reilly, 2009.  
(Listed topics from Chapters 1 to 8, 12 to 15)

**Reference Books:**

1. Getting Started with Flex 3, Jack Herrington and Emily Kim, O'Reilly, 1<sup>st</sup> Edition, 2008.
2. Flex 3: A Beginner's Guide, Michele E. Davis and John A. Phillips, Tata McGraw-Hill, 2008.
3. Essential Actionscript 3.0 – Colin Moock, O'Reilly Publications, 2007.
4. Professional Ajax, Nicholas C Zakas et al, Wrox Publications, 2006.

## Software Design Laboratory

**Subject Code: 10MCA56**  
**Hours/Week: 3**  
**Total Hours: 42**

**I.A Marks: 50**  
**Exam Hours: 03**  
**Exam Marks: 50**

The student has to draw the necessary analysis and design diagrams in UML, using any suitable UML Drawing Tool and implement in Java OR C++ OR C# a program to demonstrate the Design Pattern specified by the Examiner. For each pattern, an example is listed here.

**However, student is free to choose to solve any suitable problem to demonstrate the specified pattern. The Design Pattern is allotted based on lots from the following list:**

- 1) Publisher-Subscriber  
Example: An embedded application; An interrupt-driven module keeps track of temperature of the furnace. When the temperature is beyond preset upper / lower limits, a module that controls the heating element must be informed. Another module that displays an indicator also needs to know of such a change. Further, a log module also needs this information.
- 2) Command Processor  
Example: A simple Text Editor; Facilities provided include making the text bold, making the text in to all upper case; An Undo feature is to be implemented.
- 3) Forwarder-Receiver  
Example: A simple peer-to-peer message exchange scenario; Underlying communication protocol is TCP/IP.
- 4) Client-Dispatcher-Server  
Example: A simplified implementation of RPC
- 5) Proxy  
Example: A highly simplified implementation of a proxy web server.
- 6) Whole-Part  
Example: Implementation of any collection like a set.
- 7) Master-Slave  
Example: A multithreaded implementation of any parallelized divide-and-conquer algorithm

## .Net Laboratory

**Subject Code: 10MCA57**  
**Hours/Week: 3**  
**Total Hours: 42**

**I.A Marks: 50**  
**Exam Hours: 03**  
**Exam Marks: 50**

1. Write a Program in C# to Check whether a number is Palindrome or not.
2. Write a Program in C# to demonstrate Command line arguments Processing.
3. Write a Program in C# to find the roots of Quadratic Equation.
4. Write a Program in C# to demonstrate boxing and unBoxing.
5. Write a Program in C# to implement Stack operations.
6. Write a program to demonstrate Operator overloading.
7. Write a Program in C# to find the second largest element in a single dimensional array.
8. Write a Program in C# to multiply to matrices using Rectangular arrays.
9. Find the sum of all the elements present in a jagged array of 3 inner arrays.
10. Write a program to reverse a given string using C#.
11. 11. Using Try, Catch and Finally blocks write a program in C# to demonstrate error handling.
12. Design a simple calculator using Switch Statement in C#.
13. Demonstrate Use of Virtual and override key words in C# with a simple program
14. Implement linked lists in C# using the existing collections name space.
15. Write a program to demonstrate abstract class and abstract methods in C#.
16. Write a program in C# to build a class which implements an interface which is already existing.
17. Write a program to illustrate the use of different properties in C#.
18. Demonstrate arrays of interface types with a C# program.

**Note: In the examination *each* student picks one question from the lot of *all* 18 questions.**

## Mini Project

**Subject Code: 10MCA58**

**Hours/Week: 3**

**Total Hours: 42**

**I.A Marks: 50**

**Exam Hours: 03**

**Exam Marks: 50**

### NOTES:

- A team of **TWO** students must develop the mini project. However, during the examination, each student must demonstrate the project individually.
- The team may implement a mini project of their choice.
- The team must submit a **Brief Project Report** (25 to 30 Pages) that must include the following:
  - Introduction
  - Requirements
  - Software Development Process Model Adopted
  - Analysis and Design Models
  - Implementation
  - Testing
- The Report must be evaluated for 10 marks, Demonstration for 30 marks and Viva for 10 marks.