

**Proposed Syllabi of**

**Computer Programming and System Analysis**

**(Applied Component)**

**University of Mumbai**

**May 2005**

# **T.Y.B.Sc. (Applied Component)**

## *(Computer Programming and System Analysis)*

### **Paper I**

#### **(SSAD, Networking, and Java Programming)**

(Two lectures per week and two periods of practicals per week per batch)

#### **Unit 1. SSAD (Structured System Analysis and Design) (12 Lectures)**

##### **Introduction**

What is a system, characteristics of information system, types of information systems - Transaction Processing System (TPS), Management Information System (MIS), Decision Support System (DSS).

##### **System Development Strategies**

Systems Development Life Cycle (SDLC) method, structured analysis development method, elements of structured analysis - Data Flow Diagrams (DFD), data dictionary.

##### **Tools for determining System Requirements**

What is requirements determination, fact finding techniques, tools for documenting procedures and decisions - decision tree, decision table.

##### **Recommended Book**

*Analysis and Design of Information Systems - James A. Senn (McGraw-Hill International Editions) (Chapters 1 & 3)*

#### **Unit 2. Networking (16 Lectures)**

##### **Introduction**

What is networking, need for networking, network components - nodes, links (point to point and broadcast), network topologies - bus, star, ring, mesh, network services (connection oriented and connectionless).

##### **Network Design**

What is network design, requirements and tasks of a network, LAN, MAN, WAN and VAN.

##### **Network Architectures**

Layering principle, OSI Reference Model, TCP/IP Reference Model, Comparison of OSI and TCP/IP Reference Models.

##### **Network Switching and Multiplexing**

Bridges, interconnecting LANs with bridges, spanning tree algorithm. What is multiplexing, static multiplexing (FDM, TDM, WDM), dynamic multiplexing. What is switching, circuit switching, packet switching.

##### **Routing and Addressing**

Router, route table, routing (direct and indirect), routing characteristics, shortest path routing, Dijkstras's algorithm. TCP/IP internetworking, IP addresses (class based, classless), subnetting and subnet mask, Domain names.

##### **Recommended Book**

*Computer Networks - Andrew S. Tanenbaum (PHI) (Chapter 1: 1.1 - 1.4, chapter 2: 2.5.4, 2.5.5, Chapter 4: 4.7, Chapter 5: 5.2.1 - 5.2.4, 5.5.1 - 5.5.2, 5.6.1 - 5.6.2, Chapter 7: 7.1.1, 7.1.3).*

## **Unit 3. Java Programming (22 Lectures)**

### **Introduction to JAVA Programming**

What is java, history of java, features of java, different types of java programs, java virtual machine, JDK tool.

### **Object Oriented programming**

Object oriented approach, object oriented programming, objects and classes, behavior and attributes, fundamental principles of OOPs (encapsulation, inheritance, polymorphism, data abstraction).

### **Java Basics (Data Concepts)**

Variables and data types, declaring variables, literals, numeric literals, boolean literal, character literals, string literals, keywords, type conversion and casting.

### **Java Operators**

Assignment operator, arithmetic operators, relational operators, logical operators, bitwise operators, incrementing and decrementing operators, conditional operator, precedence and order of evaluation, statements and expressions, string arithmetic.

### **Loops and Controls**

Control statements for decision making: selection statements (if statement, if-else statement, if-else-if statement, switch statement), goto statement, looping (while loop, do while loop and for loop), nested loops, breaking out of loops (break and continue statements).

### **Arrays**

Arrays (one and two dimensional), declaring array variables, creating array objects, accessing array elements.

### **Introduction to Classes and Methods**

Defining classes, creating instance and class variables, creating objects of a class, accessing instance variables of a class. Creating methods, naming methods, accessing methods of a class, constructor methods, overloading methods, finalizer methods and overriding methods.

### **Inheritance, Interfaces and Packages**

Super and sub classes, keywords - 'extends', 'super', 'final', abstract classes, concept of interfaces and packages.

### **Java Applets Basics**

Difference of applets and application, creating applets, life cycle of applet, passing parameters to applets.

### **Graphics, Fonts and Color**

The graphics class, painting the applet, font class, draw graphical figures (oval, rectangle, square, circle, lines) and text using different fonts.

### **Recommended Book**

*The complete reference java 2: Patrick Niaughton, Herbert Schind (TMH).*  
(Chapters 1 - 6, 8 - 9, 12, 21).

## Paper II

**(Intro. to DBMS & RDBMS, SQL Commands & Functions, and C- Language)**

(Two lectures per week and two periods of practicals per week per batch)

### Unit 1. Introduction to DBMS and RDBMS (12 Lectures)

#### Introduction to Database Concepts

Database systems vs file systems, view of data, data models, data abstraction, data independence, three level architecture, database design, database languages - data definition language(DDL), data manipulation language(DML).

#### E - R Model

Basic concepts, keys, E-R diagram, design of E-R diagram schema (simple example).

#### Relational structure

Tables (relations), rows (tuples), domains, attributes, candidate keys, primary key, entity integrity constraints, referential integrity constraints, query languages, normal forms 1,2,and 3 (statements only), translation of ER schemas to relational (database) schemas (logical design), physical design.

#### Recommended Book

*Database System Concepts - Silberschatz, Korth, Sudarshan (McGraw-Hill Int. Edition) - 4th Edition (Chapter 1: 1.1 - 1.5, Chapter 2: 2.1 - 2.5, 2.8 - 2.9, Chapter 3: 3.1, Chapter 7: 7.1, 7.2, 7.7)*

### Unit 2. SQL Commands and Functions (16 Lectures)

#### Handling data

Selecting data using SELECT statement. FROM clause, WHERE clause, HAVING clause, ORDER BY, GROUP BY, DISTINCT and ALL predicates. Adding data with INSERT statement. Changing data with UPDATE statement. Removing data with DELETE statement.

#### Joining Tables

Inner joins, outer joins, cross joins, union.

#### Functions

Aggregate functions-AVG, SUM, MIN, MAX and COUNT. Date functions - DATEADD(), DATEDIFF(), GETDATE(), DATENAME(), YEAR, MONTH, WEEK, DAY. String functions - LOWER(), UPPER(), TRIM(), RTRIM(), PATINDEX(), REPLICATE(), REVERSE(), RIGHT(), SPACE().

#### Creating and Altering tables

CREATE statement, ALTER statement, DROP statement.

#### Views

Simple views, complex views, creating and editing views.

#### Constraints

Types of constraints, KEY constraints, CHECK constraints, DEFAULT constraints, disabling constraints.

#### Indexes

Understanding indexes, creating and dropping indexes, maintaining indexes.

## Recommended Books

1. Professional SQL Server 2000 Programming - Rob Vieira, Wrox Press Ltd, Shroff Publishers & Distributors Pvt Ltd, NewDelhi.( Chapters 4-10).
2. SQL Server 2000 Black Book - Patrick Dalton & Paul Whitehead, Dreamtech Press.

**Note:** Use of ready-made databases like Northwind, Pubs, Foodmart etc is to be encouraged in the practical sessions for the effective demonstration of the commands.

## Unit 3. C- Language (22 Lectures)

### Program Structure

Header and body, use of comments, construction of the program.

### Data Concepts

Variables, constants, and data types, declaring variables.

### Simple Input/Output Operations

Character strings: printf(), scanf(), single characters: getchar(), putchar()

### Operators

Assignment operators, compound assignment operators, arithmetic operators, relational operators, logical operators, increment and decrement operators, conditional operator, precedence and order of evaluation, statements and expressions.

### Type conversions

Automatic and explicit type conversions.

### Loops and Controls

Control statements for decision making: branching (if statement, if-else statement, else-if statement, switch statement), looping (while loop, do while loop and for loop), breaking out of loops (break and continue statements).

### Storage Classes

Automatic variables, external variables, register variables, static variables - scope and functions.

### Functions and Arguments

Global and local variables, function definition, return statement, calling a function (by value, by reference), recursion, recursive functions.

### Strings and Arrays

Arrays (one and two dimensional), declaring array variables, initialization of arrays, accessing array elements, string functions (strcpy, strcat, strchr, strcmp, strlen, strstr, atoi, atof).

### Pointers

Fundamentals, pointer declarations, operators on pointers, passing pointers to functions, pointers and one dimensional array, pointers and two dimensional array.

### Structures

Basics of structures, structures and functions.

## Recommended Books

*Programming in Ansi C - Ram Kumar and Rakesh Agarwal (Tata McGraw Hill)*  
(Chapters 2 - 8).

## Practical

Practicals should be arranged throughout the academic year and not at the end of the year. All programs listed below are to be written in the journal along with a short note on program development (i.e. first write the short note on program development and then write the code). Also a minimum of 8 listing (3 from C-Language, 3 from Java Programming, and 2 from SQL) of the code must be incorporated with the journal. A fee of Rs.1500/- per annum per student be charged to the students for hands on experience in the in-house as experience in the in-house computer system. It is expected that the C-Programming, the Java Programming and SQL will be done on windows platform. C-Programming can be run under either in Turbo C or C++ environment. A suitable JDK kit can be implemented for Java Programming. *Two periods per week per theory paper and 4 periods per week per practical batch are to allocated in the work load.*

### Paper I

#### JAVA Programming

1. Write a program that illustrate the concept of selection statements.
2. Write a program that illustrate the concept of looping, nested loops, breaking out of loop.
3. Write a program that illustrate the concept of arrays (one and two dimensional).
4. Write a Java program to create a Java class
  - (i) with instance variable and methods
  - (ii) with instance variables and without methods
  - (iii) without instance variables and with methods

Create an object of this class that will invoke the instance variables and methods accordingly.

5. Write a program to illustrate the concept of Java class that includes constructor with and without parameters.
6. Write a program to illustrate the concept of Java class that includes overloading methods.
7. Write a program to demonstrate the concept inheritance by defining suitable classes.
8. Write a program to illustrate the idea of Java applet.

#### SSAD and Databases using SQL

1. Single table queries using operators with select columns and restricting rows of output.
2. Multi-table queries using join operators.
3. Queries using aggregate functions, string functions, date functions.
4. Creating, updating, altering and deleting tables and views.
5. Creating tables with defaults, integrity constraints, referential integrity constraints and check constraints both at the column and table levels.
6. Creating and managing indexes.

## Paper II

### C programming

1. Creating and printing frequency distribution.
2. (a) Sum of two matrices of order  $m \times n$  and transpose of a matrix of order  $m \times n$ , where  $m, n \leq 3$ .  
(b) Multiplication of two matrices of order  $m$ , where  $m \leq 3$ , finding square and cube of a square matrix using function.
3. Simple applications of recursive functions (like Factorial of a positive integer, Generating Fibonacci Sequence, Ackerman Function, univariate equation)
4. Sorting of Numbers (using bubble sort, selection sort), and strings.
5. Using arrays to represent a large integer (that cannot be stored in a single integer variable).
6. Counting number of specified characters (one or more) in a given character string.
7. Writing a function to illustrate pointer arithmetic.
8. Using structures to find and print the average marks of five subject along with the name of a student.

### Scheme of Examination

The scheme of examination in the subject of Computer Programming and System Analysis will be as follows:

Examination	Contents	Duration	Marks
Paper - I	SSAD, Networking and Java	3 Hrs	60
Paper - II	DBMS, RDBMS, SQL and C	3 Hrs	60
	Total Marks		120
Practicals	I. Practical slip will contain (1) One program from C (2) One program from Java (3) A set of queries from SQL	3 Hrs	20 20 20
	II. Certified Journal		10
	III. Viva - voce	during practical examination	10
		Total Marks	

## Theory Examination

1. Duration of theory examination will be 3 Hours.
2. Each theory paper shall consists of 6 questions, with internal choice (a), (b) or (p), (q).
3. All questions are compulsory and each question carries 10 marks.
4. The pattern of theory question paper is as follows:

Question No.	Paper I	Paper II
1	SSAD	DBMS/RDBMS
2	Networking	SQL
3	Networking	SQL
4	Java Programming	C Programming
5	Java Programming	C Programming
6	Java Programming	C Programming

## Practical Examination

1. Duration of practical examination will be 3 hours.
2. A certified journal of Computer Programming and System Analysis must contain minimum of 20 programs (mentioned in the syllabus) with at least 7 from Java Programming, 7 from C Programming and 6 from SQL. A minimum of 8 listing (3 from Java Programming, 3 from C programming and 2 from SQL) of the code as well as the output must be incorporated with the journal.
3. The questions in the practical examination to be asked from the practicals mentioned in the syllabus. A few simple modifications may be expected during the examination.
4. Number of students per batch for practical should not exceed 20. Not more than two students be made to do practicals on one computer at a time.



## Practical Examination Slips

### C-Language

1. You are given the following data of weights in Kg varying between  $\underline{a}$  Kg and  $\underline{b}$  Kg of  $\underline{n}$  students in a class.

(Provide the data)

Write a program in C

- (a) to find the mean, variance and standard deviation of the above data.
  - (b) to prepare a frequency table (or cumulative frequency table) with class-intervals of size  $\underline{m}$ .
  - (c) to find the mode of the above data.
  - (d) to print the above information.
2. Given two matrices  $A$  of size  $\underline{m} \times \underline{n}$  and  $B$  of size  $\underline{p} \times \underline{q}$ . Write a program in C
    - (a) to calculate  $A + B$ ,  $A^t$ ,  $B^t$ .
    - (b) to find  $(A + B)^t$ ,  $A^t + B^t$ .
    - (c) verify the identity  $(A + B)^t = A^t + B^t$   
(Here  $t$  denote the transpose of a matrix)
    - (d) to print the above information.
  3. Given two matrices  $A$  and  $B$  are of size  $2 \times 2$ . Write a program in C
    - (a) to find  $|A|$ ,  $|B|$ , where  $| |$  denote determinant of the matrix, using a function.
    - (b) to calculate the product matrices  $AB$  and  $BA$ .
    - (c) to verify the identities  $|AB| = |BA| = |A| \cdot |B|$

**OR**

- (c) to verify the identity  $tr(AB) = tr(BA)$ ,  $tr$  denote the trace of a matrix.
- (d) to print the above information.

**OR**

Given a matrices  $A$  of size  $2 \times 2$ . Write a program in C

- (a) to calculate  $|A|$  where  $| |$  denote determinant of the matrix.
  - (b) to find  $adj(A)$ , the adjoint of matrix  $A$ .
  - (c) to verify the identity  $A \cdot adj(A) = I|A|$ .
  - (d) to print the above information.
4. Given a set  $S$  of  $\underline{n}$  integers. Write a program in C to
    - (a) to sort the elements in  $S$  in ascending order using an array of  $n - 2$  elements by selection sort (**or** Bubble sort) method (for example, sort 12 elements, use an array of 10 elements) and print the sorted elements.
    - (b) find the median of elements in  $S$ .
    - (c) print the above information.

**OR**

Given a set  $S$  of  $n$  integers. Write a program in C to sort the elements in  $S$  in ascending order by selection sort (or Bubble sort) method removing duplicates during the sorting process. and print the sorted array.

5. Write a program in C to
- (a) input a sentence.
  - (b) count the number of occurrences of the given pattern of letters '-' (for example, 'est' or 'ed').
  - (c) find the position of the rightmost occurrence of the pattern '-' in the sentence if the pattern occurs.
  - (d) print the above information.

**OR**

Write a program in C to

- (a) input a sentence.
  - (b) remove all occurrences of the character '-'.
  - (c) print both sentences
6. One way to compute the factorial of a large integer is to use an array to store the answer and partial results, using one element of the array per digit. To find  $n!$ , multiply each element of the array (containing the value of  $(n - 1)!$ ) by  $n$ , taking care to move the carries. Write a program in C to compute factorials of large numbers using this scheme.

**OR**

Write a program in C to

- (a) find the factorial of positive integer  $n$  using recursive function.
- (b) find and print  ${}^n C_r$  for  $0 \leq r \leq n$ .
- (c) calculate and print  $(1 + x)^n$  where  $x = -$  using binomial expansion.

**OR**

Write a program in C to print a table of values of  $e^x$  correct to six place accuracy for  $x = 0$  to  $1$  in increments of  $0.2$  where

$$e^x = 1 + x + \frac{x^2}{2!} + \dots$$

**OR**

The Ackerman's function  $Ak(m, n)$  is defined recursively by

$$\begin{aligned} Ak(0, n) &= n + 1 \text{ for } n \geq 0 \\ Ak(m, 0) &= Ak(m - 1, 1) \text{ for } m \geq 1 \\ Ak(m, n) &= Ak(m - 1, Ak(m, n - 1)) \text{ for } m \geq 1, n \geq 1 \end{aligned}$$

Write a program in C to calculate

- (a)  $Ack(1, n)$ ,  $Ack(1, n + 2)$  where  $n = \dots$
  - (b)  $Ack(2, n)$ , where  $n = \dots$
  - (c)  $Ack(4, 0)$ .
7. Write a program in C to
- (a) sort an array of  $n$  elements using pointers by (selection or bubble sort) method.
  - (b) find the maximum (or minimum) element of the array using pointers and its left most position.
8. The following data is given about  $n$  students:  
Their names, seat numbers, marks obtained in Paper I, Paper II, Paper III and Paper IV out of 100. Write a program in C using structure
- (a) to find the total marks obtained by each student, with percentage.
  - (b) to print the highest marks in each paper with the name(s) and seat number(s) of the student(s).

### Java Programming

1. Write a program in Java that reads three positive integers  $a, b, c$  and determines whether they can form the three sides of a triangle. If yes, determines whether the triangle will be an obtuse angle, or a right angle, or an acute angle triangle. If the triangle is an acute angle triangle, determine further whether the triangle is equilateral, isosceles, or scalene.
2. Write a program in Java
  - (a) that reads an unspecified number of integer argument from the command line.
  - (b) find the mean and standard deviation of those input numbers.
  - (c) print the above information.

**OR**

Write a program in Java that read a positive integer and

- (a) find and print its positive divisors.
  - (b) print also its prime divisors
3. Write a program in Java that read a positive integer  $n (> 2)$  and generate prime numbers in the range 2 through  $n$  using Sieve Method.
4. (a) Write a class whose instances represent a playing card from a deck of cards  
(b) Write a class whose instances represents a deck of cards.  
(c) Write a small program to test your deck and card classes. The program can be as simple as creating a deck of cards and displaying its cards.  
(d) Create an object of above classes that will invoke the instance variables.
5. Write a program in Java to
- (a) create a 'car' class. Initialize the color and the body with blue and wagon respectively.

- (b) Create two constructors of which one is a default constructor that creates a blue car and other contains two arguments 'color' and 'body'.
  - (c) Write a method 'toString' that returns the color and body of the car.
6. A Java class has three overloading methods. The first method accepts no parameters, the second method accepts one string and the third method accepts one string and an integer. Write a Java program to
- (a) display the message "Rose is a beautiful flower twice using first method.
  - (b) display the message Sunflower is a beautiful flower twice using second method.
  - (c) display the message Marigold is a beautiful flower four times using third method.
7. Write a Java program to do the following
- (a) Create a class called 'Number' that accepts an array of ten numbers.
  - (b) Create a sub class called 'subNum' having menu as follows:
    - (1) Display number entered.
    - (2) Sum of Numbers.
    - (3) Average of numbers.
    - (4) Maximum of numbers.
  - (c) Create an appropriate method in the subclasses to execute as per choice and it should continue until you press 'ctrl + a'.
8. Write an applet in Java
- (a) that displays Birthday greeting (using different colors and fonts for the text).
  - (b) add a border to the above applets with small circles and squares (alternating red circles and blue squares)

### **SQL Commands and Functions**

1. Single table queries using operators with select columns and restricting rows of output.
  - (a) Choose a ready made database Northwind / Foodmart / Pubs.
  - (b) Supply queries using SELECT command.
  - (c) Supply queries using SELECT with FROM, WHERE and HAVING clauses and combinations of them.
  - (d) Supply queries using SELECT with ORDER BY, GROUP BY, DISTINCT, ALL and queries along with different clauses.
2. Multi-table queries using join operators.
  - (a) Choose a ready made database Northwind / Foodmart / Pubs.
  - (b) Supply queries using SELECT command with inner joins.
  - (c) Supply queries using SELECT command with outer joins.
  - (d) Supply queries using SELECT command with cross-joins.
  - (e) Supply queries using SELECT command with union.
3. Queries using aggregate functions, string functions, date functions.
  - (a) Choose a ready made database Northwind / Foodmart / Pubs.
  - (b) Supply queries using aggregate functions AVG, SUM, MIN, MAX and COUNT.

- (c) Supply queries using string functions LOWER(), UPPER(), TRIM(), RTRIM(), PATINDEX(), REPLICATE(), REVERSE(), RIGHT() and SPACE().
  - (d) Supply queries using date functions DATEADD(), DATEDIFF(), GETDATE(), DATENAME(), YEAR, MONTH, WEEK and DAY.
  - (e) Supply queries using combinations of aggregate, string and date functions.
4. Creating, updating, altering and deleting tables and views.
- (a) Create a table based on the given schema.
  - (b) Enter specified number of records to the table.
  - (c) Update / delete records of the table.
  - (d) Alter the table based on the given guidelines.
  - (e) Create views using single table.
  - (f) Create views using more than one table.
  - (g) Alter views.
  - (h) Delete table.
5. Creating tables with defaults, integrity constraints, referential integrity constraints and check constraints both at the column and table levels.
- (a) Create tables based on the given constraints.
  - (b) Supply primary keys and foreign keys.
  - (c) Supply a default value to a particular column.
  - (d) Supply a limit for a column using check constraint.
  - (e) Draw ER diagram for the given ER model.
6. Creating and managing indexes.
- (a) Create a table based on the given schema.
  - (b) Enter specified number of records to the table.
  - (c) Create an index to the field specified.
  - (d) Maintain an index.
  - (e) Drop an index.

