

**MODIFIED SCHEME FOR the D.C.S.A. COURSE**  
SEMESTER - I (DCSA) Applicable From Aug 97 onwards

SR. NO.	SUBJECT NO.	NAME OF THE SUBJECT	TEACHING SCHEME			EXAMINATION SCHEME						
			THEORY HR.	TUTO HR.	PRACTICAL HR.	SESSIONAL M.	HR.	THEORY M.	HR.	PRACT/ ORAL	T.W. MARKS	TOTAL
1.	DCSA111	FUNDAMENTALS OF PROGRAMMING	4	--	4	25	1	50	3	50	25	150
2.	DCSA112	ACCOUNTING & FINANCIAL MANAGEMENT	3	1	--	25	1	50	3	--	25	100
3.	DCSA113	FUNDAMENTALS OF COMPUTER ORGANISATION	4	--	2	25	1	50	3	50	25	150
4.	DCSA114	BUSINESS DATA PROCESSING	4	--	2	25	1	50	3	50	25	150
5.	DCSA115	BASIC MICROCOMPUTER APPLICATIONS	2	--	4	25	1	50	3	50	25	150
T O T A L			17	1	12	125	--	250	--	200	125	700

**MODIFIED SCHEME FOR the D.C.S.A. COURSE**  
SEMESTER - II (DCSA) Applicable From Jan 98 onwards

SR. NO.	SUBJECT NO.	NAME OF THE SUBJECT	TEACHING SCHEME			EXAMINATION SCHEME						
			THEORY HR.	TUTO HR.	PRACTICAL HR.	SESSIONAL M.	HR.	THEORY M.	HR.	PRACT ORAL	T.W. MARKS	TOTAL
1	DCSA121	DATA AND FILE STRUCTURES	4	--	2	25	1	50	3	50	25	150
2	DCSA122	OBJECT ORIENTED CONCEPTS & PROGRAMMING	2	--	2	25	1	50	3	50	25	150
3	DCSA123	DATABASE MANAGEMENT SYSTEMS	4	--	2	25	1	50	3	50	25	150
4	DCSA124	COMPUTER ORIENTED NUMERICAL & STATISTICAL METHODS	4	--	2	25	1	50	3	50	25	150
5	DCSA125	SYSTEM ANALYSIS & DESIGN	3	--	--	25	1	50	3	--	25	100
6.	DCSA126	CLIENT- SERVER ARCHITECTURE & INTERFACES	4	--	2	25	1	50	3	50	25	150
7.	DCSA127	SYSTEM DEVELOPMENT PROJECT	--	--	30	25	1	--	--	300	150	450
T O T A L S			21	--	40	150	--	300	--	550	300	1350

## **DCSA111 FUNDAMENTALS OF PROGRAMMING : (Marks 50, Hours 3)**

- # Introductory concepts:
  - Introduction to computers
  - Computer characteristics
  - Modes of operation
  - Types of programming languages
  - Introduction to C
  - Desirable program characteristics
- # Data representation
  - Flow charts, Algorithms
- # Overview of C
- # Constants, variables & Data types
- # Operators & Expression
- # Managing Input & Output Operators
- # Decision Making & Branching
- # Decision Making & Looping
- # Arrays
- # Handling of character strings
- # User-Defined functions
- # Structures & unions
- # Pointers
- # File-Management in C
- # Dynamic Memory Allocation & linked list
- # The preprocessors

*References* : 1. Programming in C by Balaguruswami (Tata McGraw.Hill-95)

## **DCSA112 ACCOUNTING AND FINANCIAL MANAGEMENT (Marks 50, Hours 3)**

### **OBJECTIVES OF THE COURSE**

\* To provide an awareness of the underlying financial structures and procedure of organization so that the interaction of financial systems with information systems can be understood by the Information systems designer.

### **COURSE CONTENT**

#### **1. Principles of Accounting (15%)**

Assets, current assets, fixed assets, liabilities, current liabilities, other liabilities- owners equities; trading account: Accounting records and systems; control accounts and subsidiary ledgers; limitations.

#### **2. Assets and working capital (10%)**

Fixed assets and depreciation; assets acquisition, disposal, replacement depreciation: intangible assets, inventory methods; sources of working capital, funds, cash flows.

#### **3. Interpreting Accounts and Financial Statements (10%)**

Use of ratio in interpreting trading accounts and financial statements, limitations; other methods.

#### **4. Standards for Control (15%)**

Variable costs /fixed costs; cost volume-profit analysis; break-even, marginal and full costing; contributions; standard costings; analysis of variance; computer accounting and algorithms.

#### **5. Budgeting / Forecasting (25%)**

Characteristics of budgets, definition, advantages, proportion, forecasting long term/short term.

#### **6. Project appraisal (25%)**

Methods of capital investment decision - making discounted cash flows, internal rate of return, payback, rate of return, sensitive analysis cost of capital.

### **DCSA113 FUNDAMENTALS OF COMPUTER ORGANISATION (Marks 50, Hours 3)**

1. Number Systems, Boolean algebra, Combinational and sequential [30%] circuit design
2. Arithmetic and logic unit, Memory devices [15%]
3. Buses, interfaces and Control unit [15%]
4. Instruction formats, Addressing modes, Introduction to 8-bit, [10%] 16-bit, 32-bit microprocessors
5. I/O devices : Keyboard, VDU, Floppy and Hard Disks, Different [15%] types of printers
6. Study of IBM PC Compatibles: [15%]
  - \* Different types of Buses
  - \* Functions of Motherboard and various controlles(video, FDC, HDC, Serial and Parallel I/O controller)

#### *References :*

- (1) Digital Computer Fundamentals ( Sixth Edition) By - Thomas Bartee Pub : McGrawHill
- (2) IBM PC and Clones By - B. Govindrajalu
- (3) The 386/486 PC ( Second Edition) By - Harry Fairhead Pub : BPB

### **DCSA114 BUSINESS DATA PROCESSING (Marks 50, Hours 3)**

#### **OBJECTIVES OF THE COURSE**

\* To discuss methods of gathering input data, collecting them and organizing them into files and data base.

\* To enable students to write business application programmes such as inventory control, accounting, production planning and simulation in COBOL.

#### **COURSE CONTENT**

##### **1. Introduction to Business Organizations (10%) :**

Production,stock control, costing,purchase control,sales order processing and accounting,wage accounting,information systems needs - role of management services department.

##### **2. Data capture and Validation (10%) :**

Input form design, punched cards, key to disk systems, optical readers, on-line man-computer dialogues.

### **3. Systems Investigations (10%) :**

Project selection, feasibility analysis, fact gathering, human aspects, system design and implementation and evaluation. Input editing / validation, audit considerations. Computer work load scheduling, documentation and its importance- specification language and methods.

### **4. Business Files - Master Files - Transaction Files (20%) :**

File processing-sorting, searching, merging, matching, summarizing, Direct Access storage and retrieval, File organization techniques.

### **5. Concept of good style In Data Processing Programs Design (50%) :**

Examples of good application programs in COBOL. Report generation in COBOL.

\* COBOL - 85 Standard features are to be taught with an Introduction to 9.x Standard

Laboratory : The laboratory should emphasize on file-organization and file processing techniques.

## **DCSA115 BASIC MICROCOMPUTER APPLICATIONS (Marks 50, Hours 3)**

### **1. Installation and use of Single User Operating Systems :**

Note : Any two widely-used operating systems may be used as examples.

Hardware requirements for Microsoft DOS & Windows installation: MS-DOS, its commands and programming; Customizing and Configuring Windows; Configuring the taskbar, the Start menu; Using the Windows interface to create, print, and store a file; using Windows Explorer, Managing Disk Resources and Utilities; Manage long and short filenames; in a mixed environment; Disk Defragmenter; ScanDisk; Running Applications; Differences between a process and a thread; security considerations; installation methods; Architecture and Memory; the memory usage of a MS-DOS-based application, a 16-bit Windows-based application, and a 32-bit Windows-based application operating in Windows

### **2. Fundamentals of a PC Based DBMS System :**

User interface; Sources of help; Components of DBMS; Application Design Process; Developing the database strategy; Developing the database design; Determining the queries; Determining the forms and reports; Determining other application components; Designing and Creating Tables; Using the table wizards; Setting field and table properties; Indexes and their uses; Setting relationship between tables; Adding and Editing Data; Using existing data; Building Queries; Types of queries; Designing queries; Using the query Wizards; Designing select queries; Designing action queries; Creating other types of queries; Designing and Creating Forms, Using form wizards and builders; Using expressions in forms; Creating subforms; Customizing forms; Displaying and filtering information; Incorporating graphs and object linking and embedding (OLE); Reusing forms; Printing forms; Creating Reports; Using report wizards; Printing reports; Customizing reports by creating controls, adjusting controls, and setting control properties; Using expressions in reports; Sorting and grouping data within a report; Automating with Macros; Creating and editing macros; Running macros; Creating macro groups; Creating an AutoKeys macro; Creating and customizing menus; Creating a startup macro; security requirements; Rolling out an Application; Considerations for distributing an application

### **3. Fundamental of Worksheet Usage :**

Worksheet fundamentals; Embedding enhancing, and modifying charted data; Formatting worksheet data; Producing list-type information; Customization of the user interface for optimal performance; Data Organization; Data Analysis; Data Manipulation; Data Access; Querying

external databases from within the Worksheet; Import and export of data; Integration with other Applications.

*References :*

1. Mastering DOS 6.2 by Robbins (BPB-97)
2. Mastering WINDOWS-95 by Cowart (BPB-97)
3. Mastering MS Office Professional 7.0 for WINDOWS-95
4. Books on line by MICROSOFT

## **DCSA121 DATA AND FILE STRUCTURES (Marks 50, Hours 3)**

### **PREREQUISITES**

DCSA111 Fundamentals of Programming.

### **OBJECTIVES OF THE COURSE**

- \* To introduce basic concepts of data structures.
- \* To introduce concepts and techniques of structuring data on bulk storage devices, and data compression.
- \* Analysis of file organization techniques.

### **COURSE CONTENT**

#### **1. Data structures and Algorithms (25%) :**

The notion of the data structures, primitive and composite data types : stacks, queues, arrays, linked lists, trees and graphs. Algorithms for manipulating Data structures, binary trees and balancing trees. Hash coding.

#### **2. File processing Environment (25%) :**

Definition of record, file, blocking, compaction and database. Introductory overview of database management system.

#### **3. File Organization (30%) :**

Implimentation and tradeoffs of sequential access, index-sequential access, random access, b-trees, Inverted list and Multilist.

#### **4. Searching and Sorting (20%) :**

Internal searching and sorting algorithms, External sort/merge algorithms.

## **DCSA122 OBJECT ORIENTED CONCEPTS & PROGRAMMING (Marks 50, Hours 3)**

### **1. Object Oriented Concepts :**

Object Oriented Development; The Object Modeling Technique ; Objects and Classes; Links and Associations; Advanced Link and Association Concepts; Generalization and Inheritance; Grouping Constructs; Aggregation; Abstract Classes: Generalization as Extension and Restriction; Multiple Inheritance; Metadata; Candidate Keys; Constraints

### **2. Object Oriented Programming Style and Languages :**

Object-Oriented Style; Reusability; Extensibility; Robustness; Class Definitions; Creating Objects; Calling operations; Using Inheritance; Implementing Associations; Object-Oriented Language Features

### **3. Object Oriented Languages :**

An Example : Basic Programming; Output Using cout; Preprocessor Directives; Comments;; Integer Variables; Character Variables; Input with cin; Type float; Manipulators; Type Conversion; Arithmetic Operators; Library Functions; Relational Operators; Loops; Decisions; Logical Operators; Precedence; Other Control Statements; Structures; Enumerated Data Types; Simple Functions; Passing Arguments to Functions; Returning Values from Functions; Reference Arguments; Overloaded Functions; Inline Functions; Default Arguments; Variable and Storage Classes; A Simple Class; Objects as Physical Objects & as Data Types; Constructors; Objects as Function Arguments; returning Objects from Functions; Structures and Classes; Classes, Objects, and Memory; Static Class Data; Array Fundamentals; Arrays as Class Member Data; Arrays of Objects; Strings; Operator Overloading : Overloading Unary Operators; Overloading Binary operators; Data conversion; Inneritance : Derived Class and Base Class; Derived Class Constructors; Overriding Member Functions; Inheritance in the English Distance Class; Class Hierachies; Public and Private Inheritance; Levels of Inheritance; Multiple Inheritance; Containership : Classes within Classes; Addresses and Pointers; Pointers and Arrays; Pointers and Functions; Pointers and Strings; Memory Management : new and delete; Pointers to Objects; Pointers to Pointers; Debugging Pointers; Virtual Function; Friend Functions; Static Functions; Assigment and Copy-Initialization; The this Pointer; Streams; String I/O; character I/O; Object I/O; I/O with Multiple Objects; File Pointers; Disk I/O with Member Functions; Error Handling; Redirection; Command-Line Arguments; Printer Output; Overloading the Extraction and Insertion Operations; Multi-File-Programs; Using the Project Feature

#### **4. Visual Object-Oriented Languages : An Example :**

(a) *Introduction* : Win32-based architecture; Anatomy of a Windows-based application; Displaying and using windows; Event-driven programming; classifications of MFC; Classes in minimal MFC application; Document/view architecture; Non-document/view architecture

(b) *Development Environment* : Developer Studio; AppWizard; Component Gallery; Resource editors; Browser, Debugging; Windows messages; Messge map; Using ClassWizard to manage message handlers, Using WizardBar to handle messages; Writing output to a device, CDC class, Displaying text to the view, GDI objects, Graphics output functions, Transformations, Special Visual effects, Building menus, Adding an accelerator key to a menu; Updating the appearance of menus; Creating shortcut menus; Adding toolbars; Implementing status bars; Designing and creating dialog boxes; Initializing list and combo boxes; Working with modeless dialog boxes

#### *References :*

1. Object Oriented Programming in Turbo C ++ by Robert Lafore (Galgotia - 1994)
2. Object Oriented Modeling and Design by Rumbaugh et.al (PHI -1997)
3. Visual C ++ Book-on-Line - (MICROSOFT)
4. C++ by Balaguruswamy

### **DCSA123 DATA BASE MANAGEMENT SYSTEMS (Marks 50, Hours 3)**

#### **OBJECTIVES OF THE COURSE**

To develop and appreciation of the data resource and the issues in managing of data. In order to achieve this purpose, the course provides technical background on computer system management of data. Within the context of the technical background, the course provides

instruction in defining data needs, functions on data, user oriented data languages, and management of data within organizations. Understanding of data structure and storage. Analysis of file organization techniques, sequential, indexed sequential, multi-list, and inverted files, operating system topics related to data such as dynamic storage allocation and virtual memory. Data-base management functions and data-base management systems. Logical and physical data models. User oriented data languages. The management of the data as a resource.

#### COURSE CONTENT

##### **1. Data Environment (10%) :**

Definition of data, issues in managing data. Uses and needs of data in the organization. Defining data needs. Trade-offs between utilization of data and control of data.

##### **2. Basic Technical Concepts and System Resources for Data (20%) :**

(a) Introduction : The notion of a data structure, primitive and composite data types, basic machine architecture, character codes.

(b) Data Structures : Definition, logical structure, physical implementation, applications and operations for strings, arrays, stacks, queues, linked lists, trees and graphs. Searching and sorting techniques. Data handling facilities of higher level languages.

(c) Operating system topics : Dynamic storage management, virtual memory, role of operating systems in data management.

(d) File Organization : Implementation and tradeoffs of sequential, random, indexed-sequential, B-trees, inverted lists and multi-list organization.

##### **3. Data base Concepts (50%) :**

(a) Database management systems: Introduction and history, CODASYL, hierarchical and relational systems. Role of databases in information systems, goals and objectives of integrated databases. Data base design philosophies. Logical and physical database design conceptual data models. Query interfaces.

(b) Logical data models: data abstraction, entity-relationship model, relational database, normalization, data dictionaries and directories.

(c) Internal data model : Implementations, CODASYL-DSDL, hierarchical models. Physical data base support, memory management, relational systems, network systems, hierarchical systems.

##### **4. Use and management of data bases (20%) :**

(a) Data base management system facilities in building information system application. Use of high-level, user-oriented data language facilities for query, Update and report generation.

## **DCSA124 COMPUTER ORIENTED NUMERICAL AND STATISTICAL METHODS**

(Marks 50, Hours 3)

### **OBJECTIVES OF THE COURSE**

- \* To solve linear and non linear algebraic equations, perform operations of calculus ,fit curves and solve differential equations using a computer.
- \* To appreciate problems due to rounding errors and of convergence.

A laboratory should be an integral part of the course in which students should write programs for realistic numerical problems. Students should become familiar with numerical and statistical programme libraries such as IMSL.

### **COURSE CONTENT**

#### **1. Computer Arithmetic (10%) :**

Floating point representation of numbers,arithmetic operations with normalised floating point numbers and their consequences, error in number representation-pitfalls in computing,error propagation in evaluation.

#### **2. Iterative Methods (20%) :**

Bisection, false position,Secant, Newton-Raphson methods. Successive approximation method, Newton Raphson method for two variables, Discussion of convergence, Solving polynomial equations,Budan's theorem, Bairstow's method, Graeffe's root squaring method.

#### **3. Interpolation and Approximation (20%) :**

polynomial interpolation,Truncation error in interpolation, difference tables and calculus of differences, cubic splines, inverse interpolation. Linear regression and nonlinear regression using least square approximation, Approximation of function by Taylor series and Chebyshev polynomials.

#### **4. Numerical Differentiation and Integration (15%) :**

Differentiation formulas based on polynomial fit, pit-falls in differentiation. Trapezoidal, Simpson's and Gaussian Quadrature formulas.

#### **5. Solution of Simultaneous Linear Equation and Ordinary Differential EQNS(20%) :**

Refinement of solution in Gauss elimination method-pivoting,ILL conditional equations, Gauss-seidal and Gauss Jacobi iterative methods. Taylor series and Euler methods. error analysis. Runge-Kutta methods, predictor-corrector methods, automatic error monitoring and change of step size. Stability of solution.

#### **6. Statistical Methods (10%) :**

Frequency Distributions,data analysis,expectations and moments, correlation and regression

## **DCSA125 SYSTEMS ANALYSIS AND DESIGN (Marks 50, Hours 3)**



## OBJECTIVES OF THE COURSE

- \* To provide the knowledge and skills necessary to develop physical design and implement an operational system from the logical design.
- \* To describe the process of planning for change and the post implementation reviews and changes. Both technological and managerial aspects of system design and implementation are considered.

Case studies should be used as appropriate. Laboratory exercise should include the use of computer - assisted methods for system design. The human engineering aspects of system design should be emphasised.

## COURSE CONTENT

### **1. Application Development Strategies :**

Selection from alternative strategies for application development using, for example, contingency theory. Development alternatives: adoption of packages for new development (outside developer or in-house development). Development methodologies: life cycle, prototyping etc. Influences on development strategy: master plan, organisational environment, development organisation and resources, and information system structure and resources.

### **2. Application System Development Life cycle :**

Overview of the phases of application system development life cycle and their interrelationship. Problem identification and feasibility assessment, requirements determination, logical and physical design, planning to accommodate change, program development, implementation, and post implementation evaluation. Emphasis on phased development approach in planning and completing the study project. Requirements for documenting and auditability.

### **3. Problem Need Identification and Feasibility Assessment :**

Sources of problems and needs. Defining the "real" need/problem. Problem analysis: degree of uncertainty, usefulness of information system application, programmability volatility.

Preliminary application requirements determination: variable versus standardized outputs, reporting, data acquisition techniques, application life expectancy, ownership/maintenance responsibility. Preliminary specification : user-system interfaces, mode of operation, input/output, software, hardware.

Feasibility assessment: economic, technical operational and schedule feasibility; and performance to information system master plan. Justification/approval by allocation mechanism: steering committee, pricing of information services, payback, cost/benefit analysis. Suggested deliverables by students: feasibility analysis report and oral presentation plus general application objectives report.

### **4. Information Requirements Determination :**

Strategies for obtaining information requirements for an application : eliciting user definition of requirements, studying and modifying information delivered by an existing system, deriving

requirements through study of utilizing system, normative derivation from characteristics of utilizing system, interactive discovery of requirements through use of a prototype system.

Techniques for information requirements determination: document study, system study, observation, interviewing. Communication skills, listening, writing, presenting, Management/inter-personal skills: conflict resolution, negotiation. Requirements documentation methods, narrative, graphics layouts, requirements language. Methods for providing assurance that requirements are correct and complete. Selecting strategies and techniques. Planning the process.

Suggested deliverables by students: (1) live interviews with video tape critique, and (2) detailed requirements specifications report plus oral presentation.

### **5. Requirement Analysis and Logical Specification :**

Description of system logical data flows, files, and processes by graphical and automated design techniques : top-down design and hierarchical decomposition, tests for logical completeness and consistency. Data dictionaries: content, format, and organisation; automated versus manual implementations. Process description methods such as structured English and decision tables, Documenting an existing system. Logical design for target system: data flows; files, and processes; selection of interfaces (automated versus manual) ; the user interface. User views of data for data base design. File requirements: transport volume, response times, integrity, security, etc.

Suggested deliverables by students: general logical design report; documentation plan, walkthrough performance.

### **6. Quality Assurance Review of Logical Design :**

Dimensions for review: Satisfaction of user objectives, costs and benefits, logical completeness and consistency tests, quality of user interface, participants and organisation for review progress. Walkthrough. Certification documents. User sign-off. Suggested deliverables by students: design walkthrough plan, walkthrough performance and report.

### **7. Detailed Logical Design :**

Logical software design. Modularization control and data flows, process interaction, process organisation, module determination, module specifications. Logical file design, data sharing file structure and logical access methods, data representation. Logical database design: alternative models such as the Entity Relationship database design: alternative models such as the Entity Relationship model scheme levels (Conceptual, external, internal). Representation of user views, View integration: Logical access paths. Improvement of logical view: normalization, aggregation, abstraction, etc.

### **8. Physical Design :**

File design: structure, access, Data base interface: alternatives, logical scheme Design. System integrity: security; privacy, auditability, error recovery, system backup. Preliminary procedures design. Design of controls. Software test design: control, extreme values, etc. Development of implementation plan: strategies, coordination. Suggested deliverables by students: physical system design report and implementation plan (programme structure: test plan, conversion plan, training plan, operations procedures chart).

## **DCSA126 CLIENT-SERVER ARCHITECTURE & INTERFACES (Marks 50, Hours 3)**

### **1. Client-Server Architecture :**

Distributed processing; Cooperative processing; Client/Server processing; Peer-to-peer processing; Client/Server architecture; Time sharing; Resource sharing; How do the client and server communicate ?; Differences between a database server and a file server; Database server architecture; open systems interconnect (ISO) model; Application Architecture : Information system architecture; Function (application) architecture; Data architecture; Network(technology)architecture; application partitioning models; Distributed presentation; Remote presentation; Distributed logic; Remote data management; Distributed data management; Developer's view of a client/server application; Application segmentation; User interface; Data validation; Data transfer; Data integrity; Transaction control; Server processing; Data Architecture : Why is enterprise data connectivity an issue ?; Data access strategies; Distributed database; Centralized database; Heterogeneous data access; Transactional data versus analytic data; Data warehouse; middleware ; Common interface; Common gateway; Common protocol; Technical Architecture : Selection of Platform and Development Tools; Database server selection; Types of front-end development tools; Integration of tools; Database hardware; Computer-aided software engineering (CASE) tools; Selection of Client/Server Application : Objectives; Selection criteria; Case study; selection of pilot project; Service Model: User Services, Business Services, Data Services.

## **2. Client/Server Application System Development Life Cycle (SDLC) :**

A mission critical application; Functional specification; Business rules; Database design; Data distribution; Prototyping; User interface design; Front-end module design; Reusable modules; Goals of the design stage; Where is user interface (UI) design in the SDLC ?; The usability iceberg Process for doing UI design, Requirement gathering; Multiple prototypes

## **3. Client-Server Application Tools - I :**

Visual Basic : Objects, properties, methods, and events; Form, control, and application; Properties and methods of a form; an Object Browser, and basic controls; Changing properties and calling methods with code; adding code to event procedures; Control accessibility; Interacting with the user; Working with code statements; Variables and Procedures : Variables; Constants; Converting data types; procedures; Debugging; Types of errors; Debugging toolbar; Break mode; Debug window; Controlling Program Execution; Conditional statements; Looping statement; Working with Controls : Additional standard controls; Customs controls; Menus; Status Bars and Toolbars; Creating a menu; Pop-up menus; Data Access with Data Control : Overview of a database (table, field record key); Data control overview; Viewing records; Data Form Designer; Entering and editing records; Drag and Drop Editing; Steps for adding drag and drop functionality

## **4. Client Application Tools - II : Power Builder :**

Creating Tables and Windows; Database Painter; Multiple Data Windows and Tables; Using FoxPro As An ODBC Data Source; MS-Word As An OLE Server For Power Builder; Debugging Managing and Running Applications; Help Compiler and Help System; Data Pipeline

*References :*

1. Books-On-Line for Microsoft Visual Basic 4.0
2. Application Development with Power Builder by Rajkumar Sedani (Galgotia - 95)
3. Guide to Client Server by Joe Salemi (BPB-94)

## **DCSA127 SYSTEM DEVELOPMENT PROJECT - I**

### **PREREQUISITES**

## DCSA125 Systems Analysis & Design

After a student has appeared at the theory and practical examinations of the second semester, he can take up the System Development Project for 8 weeks. Thereafter the examination for the System Development Project will be conducted.

### OBJECTIVES OF THE COURSE

To provide the students with experience in analysing, designing, implementation and evaluating information systems.

### INSTRUCTIONAL NOTES

Students are assigned one or more system development projects. The projects involve part or all of the system development cycle. Students work in teams to acquire practical the behavioral considerations in systems development.

ALTERNATIVES :

#### **1. Development of a System for a Local Firm :**

Under supervision of the faculty and the systems analysis staff, students develop a small applications for a computer - based management information system for a local firm. Students might also work as members of established client company teams.

#### **2. Development of a System for a University/College :**

Under the supervision of the faculty and the university administrative data processing unit, students develop a system to benefit the university. Examples are: alumni record and follow - up system, bookstore ordering/accounting, classroom scheduling system.

#### **3. Development for a Hypothetical Application :**

Students develop an application for a hypothetical firm and prepare system development deliverables.

