

COMPUTER SCIENCE



Computers are everywhere. The need and purpose of computers are inevitable. The knowledge of hardware, software, operating principles and the various application oriented tools are essential nowadays for everyone. The computers are amazingly used for communication, producing music, photograph editing, videos, playing sophisticated games and much more. Computer related skills are essential in many careers. The knowledge of computers is useful other than the work places too. Basic computing principles have not changed over the years and will be valid well into the future. By mastering fundamental concepts and terminology a strong base can be developed which will support further learning. The subjects related to the above are given in detail.

PROGRAMMING IN C



C is a programming language which born at “AT & T’s Bell Laboratories” of USA in 1972. It was written by Dennis Ritchie. This language was created for a specific purpose: to design the UNIX operating system.

Program written in C are very efficient and fast. This is due to its variety of data types and powerful operators. It is many time faster than BASIC. This helps developers in saving their valuable time.

C is a powerful and flexible language which helps system developers to deliver various complex tasks with ease. C is used for diverse projects as operating systems, word processors, graphics, spreadsheets, and even compilers for other languages.

C is popular among professional programmers for programming, as a result, a wide variety of C compilers and helpful accessories are available.

C is highly portable language. This means that a C program written for one computer system (an IBM PC, for example) can be run on another system (a DEC VAX system, perhaps) with little or no modification. Portability is enhanced by the ANSI standard for C, the set of rules for C compilers.

COMPUTER SCIENCE SYLLABUS

PROGRAMMING IN C

- UNIT - I C Fundamentals - Character set – Identifier and keywords – Data types – Constants – Variables – Declarations – Expressions – Statements – Arithmetic, unary relational and logical assignment and conditional operators - library functions.
- UNIT - II Data input/output functions – Simple C Programs – Flow of control – Control Structures – Switch, break and continue, go to statements – Comma operator.
- UNIT - III Functions – Defining, Accessing functions – Functions prototypes. – Passing arguments – Recursions – storage classes – Multifile programs.
- UNIT - IV Arrays – Defining and processing – Passing arrays to functions – Multidimensional arrays – Arrays and string – Structures – Passing structures to functions – Self – Referential structures – Unions.
- UNIT - V Pointers – Declarations – Passing pointers to functions – Operation in pointers – Pointers and arrays – Arrays of pointers – Structures and pointers – Files : creating, processing, opening and closing – Bit wise operations.

BOOKS FOR STUDY :

1. Programming with C, Byron S Gottfried, Tata McGraw hill 1988.

BOOKS FOR REFERENCE :

1. Programming in ANSI C, E. Balagurusamy, Tata McGraw hill.
2. Let Us C, Yashavant. P. Kanetkar, BPB Publications, 2001.
3. Programming with C, K.R. Venugopal, Sudeep. R. Prasad, Tata McGraw Hill, 1977.
4. C Made easy, Herbert Schildt Osborn, Tata McGraw hill.
5. Programming in C, D. Ravichandran, New Age International (P) Ltd., Edition 1995.
6. The C Programming Language, Brian, W. Kernighan, Dennis Ritchie, Prentice Hall of India, 1990.
7. Computer Programming in C, V. Rajaraman, Prentice Hall of India, 1994.



VISUAL PROGRAMMING



Visual Basic changed the face of Windows programming by removing the complex burden of writing code for the user interface (UI). By allowing programmers to draw their own UI, it freed them to concentrate on the business problems they were trying to solve. Once the UI is drawn, the programmer can then add code to react to events.

Visual Basic has also been extensible from the very beginning.

When Microsoft introduced Visual Basic 3.0, the programming world changed again. Now you could build database applications directly accessible to users (so called front-end applications) completely with Visual Basic. Microsoft accomplished this task with the introduction of the Data Access Objects (DAO), which allowed programmers to manipulate data with the same ease as manipulating the user interface.

Versions 4.0 and 5.0 extended the capabilities of version 3.0 in order to allow developers to target the new Windows 95 platform. Crucially they also made it easier for developers to write code, which could then be manipulated in order to be used by other language developers. Version 6.0 gave us a new way to access databases with the integration of ActiveX Data Objects (ADO). ADO was developed by Microsoft to aid web developers using Active Server Pages to access databases. With all of the improvements to Visual Basic over the years, it ensured its dominant place in the programming world. It helps developers write robust and maintainable applications in record time.

UNIT – I Customizing a Form and writing Simple Programs: Starting a New Project - The Properties Window - Common Form Properties - Making a Form Responsive - Printing a Visual Representation of a Form - Creating Stand - Alone Windows Programs First Steps in Building the User Interface: The Toolbox - Creating Controls-The Name (Control Name) Property - Properties of Command Buttons - Simple Event Procedures for Command Buttons - Access Keys - Image Controls - Text Boxes-Labels- Navigating Between Controls - Message Boxes - The Grid - The Display in Visual Basic-The ASCII Representation of Forms

First Steps in Programming: Anatomy of a Visual Basic Application - The Code Window - Visual Basic's Editing Tools - Statements in Visual Basic - Variables - Setting Properties with Code - Data Types - Working with Variables - More on Strings - More on Numbers - Example Program: A Mortgage Calculator - Constants - Input Boxes

UNIT – II Displaying Information: Displaying information on a Form - The Format Function - Picture Boxes - Rich Text Boxes - The Printer Object

Controlling Program Flow: Determinate Loops - Indeterminate Loops- Making Decisions (Conditionals) - Select Case - Nested If-Then's - The Go To
Built-In Functions: String Functions - The Like Function and Fuzzy Searching - The Rnd Function- Bit - Twiddling Functions - Numeric Functions - Date and Time Functions- Financial Functions - Tables with the Most Useful Functions.
Writing Your Own Functions and Procedures: Function Procedures - Sub Procedures- Advanced Uses of Procedures and Functions - Using the Object Browser to Navigate Subprograms

UNIT – III Organizing Information via Code: Lists: One-Dimensional Arrays - Arrays with More Than One Dimension - Using Lists and Arrays with Functions - The New Array - Based String - Sorting and Searching - Records (User-Defined Types) - The With Statement- Enums. Organizing Information via Controls: Control Arrays - List and Combo Boxes-The Flex Grid control.
Building Larger Projects: Projects with Multiple Forms - Code Modules: Global Procedures - The Do Events Function and Sub Main - Sub Main - Accessing Windows Functions - Error Trapping - Some General Words on Program Design

UNIT - IV VB Objects and an Introduction to Object-Oriented Programming: The Object Browser - Manipulating Objects Built into Visual Basic - General Object Variables - Collections - Getting Started with Object - Oriented Programming - Creating an Object in Visual Basic - Building Your Own Classes Finishing the Interface: How Visual Basic Displays Work: ZOrder - The Toolbox Revisited - Common Dialog Boxes - The Microsoft Windows Common Controls6.0- Menus - MDI Forms - Making Forms Independent of Resizing and Screen Resolution- Help Systems

Tools and Techniques for Testing, Debugging and Optimization: Testing-Bugs - The Immediate Window - The Debugging Tools

and What They Do-Stopping Programs Temporarily-Final Remarks on Debugging. An Introduction to Graphics: Fundamentals of Graphics - Screen Scales - The Line And Shape Controls - Graphics via Code - Lines and Boxes - Circles, Ellipses and Pie Charts - Curves

UNIT – V Monitoring Mouse Activity: The Mouse Event Procedures - Dragging and Dropping Operations for Controls-Example: Deleting Files via Drag/Drop

Basic File Handling: File Commands-Sequential Files - Making a File Program Robust: Error Trapping - Random - Access Files - Binary Files - Sharing Files - Adding Licensing Screens - Keeping File Information Secret

File System Controls and File System Objects: File System Controls - The File System Object

Connecting to Databases: Getting Connected - Using DAO to Build a Simple Database Interface - Programming with ADO-In Depth-Adding a Data Interface to your Object Models - Working with the Visual Basic Report Designer

BOOKS FOR STUDY:

1. VISUAL BASIC 6 from the GROUND UP By Gary Cornell, Tata McGraw-Hill, Edition 1999.
2. THE COMPLETE REFERENCE VISUAL BASIC 6 By Noel Jerke, Tata McGraw-Hill, Edition 1999.

BOOKS FOR REFERENCE:

1. INTROUCTION TO PROGRAMMING USING VISUAL BASIC By Gary Bronson,Scott Jones Publishers
2. VISUAL BASIC 6 PROGRAMMING BLACK BOOK By Steven Holzer, Dream tech Press, Edition 2004

DATA STRUCTURES THROUGH C++

A data structure is a specialized format for organizing and storing data. General data structure types include the array, the file, the record, the table, the tree, and so on. Any data structure is designed to organize data to suit a specific purpose so that it can be accessed and worked with in appropriate ways. In computer programming, a data structure may be selected or designed to store data for the purpose of working on it with various algorithms.

C++ is a general purpose multi-paradigm programming language. It is an extension of the C language and as such most C code can easily be made to compile in C++. Some of the major additions to C include object-orientated programming, operator overloading, multiple inheritance and exception handling.

The development of C++ began in 1979, seven years after C first made its appearance. Despite being what many consider legacy languages, C and C++ are still the most widely used languages in the software industry. They are used in creating everything from operating systems and embedded software to desktop applications, games and so on.

UNIT – I C++ Programming Basics Loops and Decisions Structures and Functions – Objects and classes.

UNIT – II Arrays – Operator Overloading – Inheritance – Pointers.

UNIT – III Introduction : Data Structures – Operations – Algorithms : Complexity, Time – space trade off, Arrays, Records and Pointers : Linear Arrays – Representation – Traversing – Inserting and Deleting – Sorting – Linear Searching – Binary search – Multidimensional Arrays – Pointers – Records – Representation Of Records in Memory – Parallel Arrays. Linked Lists : Introduction – Representation – Traversing – Searching – Memory Allocation – Insertion – Deletion.

UNIT – IV **Stacks, Queues, Recursion:**

Introduction – Array Representation – Arithmetic Expressions – Quick Sort – Recursion – Queues – Deques.

UNIT V **Trees:** Introduction - Binary Trees – Representing – Traversing – Traversal Algorithms Using Stacks – Header Nodes : Threads. Graphs: Introduction – Terminology – Sequential Representation – Warshall's Algorithm – Linked Representation.

BOOKS FOR STUDY :

1. Theory and Problems of Data structures, Seymour Lipschutz, Schaum's Outline Series, 1986.
2. Object Oriented Programming in Turbo C++, Robert Lafore, Galgotia Publications Pvt Limited, 1995.

BOOKS FOR REFERENCE :

1. Mastering in C++, K R Venugopal, Rajkumar, T Ravishankar, Tata McGraw Hall, 2001.
2. Fundamentals of Data Structures in Pascal, Ellis Horowitz, Sartaj Sahni Galgotia book House, 1984.
3. Data Structures and program Design, Robert Lkrusa, PHI, II Edition.
4. Introduction to Data Structures, Bhagat Singh, Thomas L Naps, Galgotia Book House, 1984.
5. Fundamentals of Computer Algorithms, Ellis Horowitz, Sartaj sahani, Galgotia Book house 1981.
6. Programming with C++, D. Ravichandran, Tata McGraw Hill, II Edition 2003.
7. Object Oriented Programming with C++, E Balaguruswamy, Tata McGraw Hill, Edition 1998.
8. C/C++ Programmer's Guide, Chris H Pappas & William Murray, BPB Publications 1996.



SOFTWARE ENGINEERING



Software Engineering is an approach to developing software that attempts to treat it as a formal process more like traditional engineering than the craft that many programmers believe it is. We talk of crafting an application, refining and polishing it, as if it were a wooden sculpture, not a series of logic instructions.

Manufacturers cannot build complex life-critical systems like aircraft, nuclear reactor controls, medical systems and expect the software to be thrown together. They require the whole process to be thoroughly managed, so that budgets can be estimated, staff recruited, and to minimize the risk of failure or expensive mistakes.

The whole design process has to be formally managed long before the first line of code is written. Enormous design documents- hundreds or thousands of pages long are produced using C.A.S.E. (Computer Aided Software Engineering) tools then converted into Design Specification documents which are used to design code.

One distinguishing feature of Software Engineering is the paper trail that it produces. Designs have to be signed off by Managers and Technical Authorities all the way from top to bottom and the role of Quality Assurance is to check the paper trail. Many Software Engineers would admit that their job is around 70% paperwork and 30% code. It's a costly way to write software and this is why avionics in modern aircraft are so expensive

UNIT – I Introduction – Definition – Size factors – Quality and Productivity factors – Managerial issues – Planning a software project: Defining the problem, Developing a solution strategy, Planning the Development process and Organizational Structure.

UNIT – II Software cost estimation: Cost Factors – Software cost estimation techniques, Staffing – Level estimation and maintenance costs, - software requirements definition: specification, formal specification techniques.

UNIT – III Software design: Fundamental design concepts – Modules and Modularization – Design Notations – Design techniques Design guidelines.

UNIT – IV Verification and validation techniques : Quality assurance – Walkthroughs and Inspections – Static analysis- Symbolic Execution – UNIT – testing and Debugging – System testing – Formal Verification.

UNIT – V Software Maintenance: Enhancing Maintainability during development – Managerial Aspects of Software Maintenance – Configuration Management – Source Code Metrics – Other Maintenance tools and techniques.

BOOKS FOR STUDY :

1. Software Engineering Concepts, Richard Fairley, TMH, Edition 1997

BOOK FOR REFERENCE:

1. Software Engineering, Ian Sommerville, Pearson Education, Edition 1997.
2. Software Engineering, David Gustafson Schaum's Outlines, TMH, 2003.
3. Software Engineering, K K Agarwal & Yogesh Singh New Age International, 2002.
4. Software Engineering : A Practitioners Approach, Roger S Pressman, TMH, 2001.

PROGRAMMING IN JAVA

Java is a high-level object-oriented programming language developed by Sun Microsystems, which became part of Oracle Corporation in 2010. The language is very similar to C++, but has been simplified to make it easier to write bug free code. Most notably, there are no pointers in Java, instead all memory allocation and deallocation is handled automatically.

Despite simplifications like this Java has considerably more functionality than both C and C++, due to its large class library. Java programs also have high performance and can be made very secure, which has contributed to making Java the most popular general purpose programming language in use today.

Another key feature of Java is that it is platform independent. This is achieved by only compiling programs half-way, into platform independent instructions called bytecode. The bytecode is then interpreted, or run, by the Java Virtual Machine (JVM). This means that any system that has this program and its accompanying libraries installed can run Java applications.

UNIT – I The Genesis of Java – An over view of Java – Data Types – Variables - Arrays – Operators – Control statements.

UNIT – II Introduction classes – A closer look at Methods and Classes – A closer look at Methods and Classes – Inheritance.

UNIT – III Packages and Interface – Exception Handling – Multi Threaded programming.

UNIT – IV I/O Streams – Applets and other topics – String Handling – Java Utilities part 1 : The collections frame work – Java Utilities part 2 : More utility Classes.

UNIT – V The Applet classes – Event handling – AWT: Working with Windows, Graphics, Text, using AWT controls Layout Managers and Menus.

BOOKS FOR STUDY :

1. Java 2 (The Complete Reference), Patric. Naughton, Herbert.Schildt, III Edution, Tata McGraw Hill 1999.

BOOKS FOR REFERENCE :

1. Core Java Fundamentals (Volume I), Cay.S.Horstmann, Gray Cornell, Fifth Edition, PHI, 2001.
2. The Java Programming Language, K. Arnold, J Gosling, Second Edition, Addison Wesley, 1996

WEB TECHNOLOGY



Web technologies related to the interface between web servers and their clients. This information includes markup languages, programming interfaces and languages, and standards for document identification and display. HTML (Hypertext Markup Language) is the set of markup symbols or codes inserted in a file intended for display on a World Wide Web browser page. The markup tells the Web browser how to display a Web page's words and images for the user. Each individual markup code is referred to as an element (but many people also refer to it as a tag).

JavaScript is primarily used as a client-side scripting language used for creating dynamic web pages. The scripts are embedded into HTML pages that are then executed on the client's system when they view those web pages. Despite its name JavaScript is a completely different language than Java. JavaScript is an interpreted language, meaning that scripts execute without preliminary compilation.

ASP.NET is the successor to classic ASP. It is not a programming language. Instead, it's a programming framework for building web-based applications. It's part of the .NET Framework and supports the .NET languages such as C#, Visual Basic, and J#.

ASP.NET files are run on the server-side, just as classic ASP and PHP, however unlike these languages ASP.NET is compiled, making it faster than interpreted languages. ASP.NET code is not fully backward compatible with classic ASP, since the VBScript commonly used in ASP has been replaced by full Visual Basic. Although it's still possible to write code in the same way as in classic ASP, with code included in the markup, ASP.NET encourages a completely different approach by allowing code to be separated from the HTML markup, making it easier to program.

UNIT - I Internet Basic – Introduction to HTML – List – Creating Table – Linking document – Frames _ Graphics to HTML Doc – Style sheet – Style sheet basic – Add style to document – Creating style sheet – rules – Style sheet Properties – Font - Text – List – Color and background color – Box – Display properties.

- UNIT – II** Introduction to Java Script – Advantage of Java script – Data type – Variable – Array operator and expression and looping constructor – Function – Dialog box.
- UNIT – III** Java Script – Document object model – Introduction – Object in HTML – Event handling – Window object – Document object – Browser object – Form object – Navigator object – Screen object – Built in object – User defined object – Cookies.
- UNIT – IV** ASP.NET Language structure – Page structure – Page event – Properties and compiler directives. HTML server controls – Anchor, Tables, Forms, Files. Basic web server controls – Labels, Text box, Button, image, links, check and radio button, hyper link. Data list – Web server controls – Check box lists, radio button list, drop down list, list box, data grid, and repeater.
- UNIT – V** Request and response objects, Cookies, Working with data – OLEDB connection class, command class, transaction class, data adapter class, data set class. Advanced issues – email, application issues, working with IIS and page directives, error handling, Security – Authentication – IP address, secure by SSL, and client certificates.

BOOKS FOR STUDY

1. Web Enabled Commercial Application Development using HTML., DHTML, JAVA Script, PERL, CGI, Bay Ross, BPB Publications, 2000.
2. Mastering JavaScript, J Jaworski, BPB Publications, 1999.
3. ASP.NET Developers Guide, G Buczek, TMH, 2002.

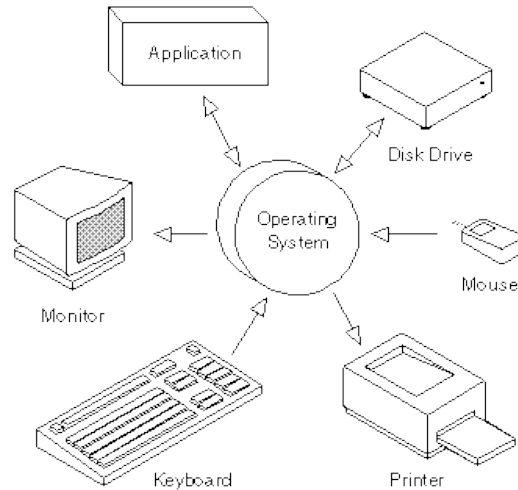
REFERENCE :

1. How to Do Everything with HTML, James H.Pence, Tata McGraw Hill edition 2001.
2. Complete Reference HTML, III Ed, T A Powel, PMH, 2002.



05vcBeginning HTML
(XHTML).flv

OPERATING SYSTEM



The most important program that runs on a computer. Every general-purpose computer must have an operating system to run other programs. Operating systems perform basic tasks, such as recognizing input from the keyboard, sending output to the display screen, keeping track of files and directories on the disk, and controlling peripheral devices such as disk drives and printers.

For large systems, the operating system has even greater responsibilities and powers. It is like a traffic cop -- it makes sure that different programs and users running at the same time do not interfere with each other. The operating system is also responsible for [security](#), ensuring that unauthorized users do not access the system.

Operating systems can be classified as follows:

- **multi-user** : Allows two or more users to run programs at the same time. Some operating systems permit hundreds or even thousands of concurrent users.
- **multiprocessing** : Supports running a program on more than one CPU.
- **multitasking** : Allows more than one program to run concurrently.

- **multithreading** : Allows different parts of a single program to run concurrently.
- **real time**: Responds to input instantly. General-purpose operating systems, such as DOS and UNIX, are not real-time.

Operating systems provide a software platform on top of which other programs, called *application programs*, can run. The application programs must be written to run on top of a particular operating system. Your choice of operating system, therefore, determines to a great extent the applications you can run. For PCs, the most popular operating systems are DOS, OS/2, and Windows, but others are available, such as Linux.

User normally interact with the operating system through a set of commands. For example, the DOS operating system contains commands such as COPY and RENAME for copying files and changing the names of files, respectively. The commands are accepted and executed by a part of the operating system called the command processor or command line interpreter. Graphical user interfaces allow you to enter commands by pointing and clicking at objects that appear on the screen.

UNIT – I OVERVIEW

Introduction: What operating systems do? – Computer system organization-Computer system architecture - Operating system structure - OS operations-Process management-Memory management - Storage Management-Protection and Security - Distributed Systems-Special purpose Systems-Computing Environments

System Structures: 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

UNIT – II PROCESSOR MANAGEMENT

Process - concept: Overview - Process scheduling - Operations on process – Inter process communication

Process scheduling: Basic concepts - Scheduling criteria - Scheduling Algorithms - Multiple - processor scheduling

Synchronization: Background - The critical section problem – Peterson’s solution - Synchronization Hardware – Semaphores – Classic problems of synchronization - Monitors

Dead Locks: System model - Deadlock characterization - Methods for Handling Deadlocks - Deadlock Prevention - Deadlock Avoidance - Deadlock detection - Recovery from Deadlock

UNIT – III MEMORY MANAGEMENT:

Memory-Management Strategies: Background – Swapping - Contiguous Memory Allocation - Paging - Structure of the Page Table - Segmentation-Example: The Intel Pentium

Virtual-Memory Management: Background - Demand Paging - Copy-on-Write -Page replacement - Allocation Frames - Thrashing

UNIT – IV STORAGE MANAGEMENT & I/O SYSTEMS:

File System: File Concept - Access Methods-Directory Structure
Implementing File Systems: File-system structure - File-system Implementation - Directory Implementation - Allocation Methods - Free Space management - Efficiency and Performance
I/O Systems: Overview - I/O Hardware - Application I/O Interface - Kernel I/O Subsystem

UNIT – V CASE STUDIES

Windows XP: History - Design Principles - System Components - Environmental Subsystems - File System - Networking - Programmer Interface

BOOKS FOR STUDY:

1. OPERATING SYSTEM PRINCIPLES (Seventh Edition) By Abraham Silberschatz, Peter Baer Galvin, Greg Gagne, John Wiley Edition

BOOKS FOR REFERENCE:

1. OPERATING SYSTEMS By Harvey.M.Deital, Pearson Education 2001
2. OPERATING SYSTEMS CONCEPTS (Fifth Edition) By Silberschatz, Galvin Addison Wesley Publication,1998



06vcOperating
System.flv

COMPUTER NETWORKS



A **computer network**, often simply referred to as a network, is a collection of hardware components and computers interconnected by communication channels that allow sharing of resources and information.

Networks may be classified according to a wide variety of characteristics such as the medium used to transport the data, communications protocol used, scale, topology, and organizational scope.

The rules and data formats for exchanging information in a computer network

are defined by communications protocols. Well-known communications protocols are Ethernet, a hardware and [Link Layer](#) standard that is ubiquitous in local area networks, and the [Internet Protocol Suite](#), which defines a set of protocols for internetworking, i.e. for data communication between multiple networks, as well as host-to-host data transfer, and application-specific data transmission formats.

UNIT – I Introduction: Uses of computer networks. Network Hardware – Network Software – Reference Models.

UNIT – II The Physical Layer: Guided Transmission Media – Wireless Transmission – Communications Satellites – PSTN.

UNIT – III The Data Link Layer : Data Link Layer Design Issues – Error detection and correction - Elementary data link protocols – Medium Access Control sublayer – Multiple Access Protocols.

UNIT – IV The Network Layer : The Network Layer Design issues – Routing Algorithms – Congestion control algorithms : General Principles of Congestion Control – Congestion Prevention Policies – Congestion Control in Virtual – Circuit Subnets and Datagram Subnets.

UNIT – V The Transport Layer and The Application Layer : Elements of Transport Protocols – DNS – Email.

BOOKS FOR STUDY :

1. Andrew S Tanenbaum (2003) Computer Networks PHI, 4th Edition

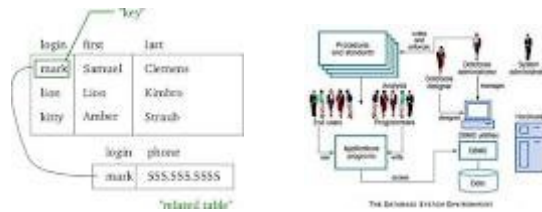
BOOKS FOR REFERENCE :

1. Barry Nance (1998) Introduction to Networking PHI.
2. Fred Halsall (1995) Data Communications, Computer Networks and Open Systems, Addison Wesley.
3. Lamarca (2002) Communication Nwtworks McGraw Hill.
4. Behrouz Forouzan (2002) Introduction of Data Communication and Networking TMH



07vcComputer
Networking.flv

DATABASE MANAGEMENT SYSTEM



A collection of programs that enables you to store, modify, and extract information from a database. There are many different types of DBMSs, ranging from small systems that run on personal computers to huge systems that run on mainframes. The following are examples of database applications:

- computerized library systems
- automated teller machines
- flight reservation systems
- computerized parts inventory systems

From a technical standpoint, DBMSs can differ widely. The terms *relational*, *network*, *flat*, and [hierarchical](#) all refer to the way a DBMS organizes information internally. The internal organization can affect how quickly and flexibly you can extract information.

UNIT – I Introduction: Components of a Database Management – System - Advantages of the Database Management - System Approach - Leading Commercial – Databases - Brief History of Database Management Systems - Application Development - The Feasibility Study System Design: Designing Databases - Class Diagrams - Data Types-Events - Large Projects - Application Design

UNIT – II Data Normalization: Tables, Classes, and Keys - Sample Database for a Video Store - First Normal Form - Second Normal Form-Third Normal Form - Beyond Third Normal Form - Data Rules and Integrity - The Effects of Business Rules- Converting a Class Diagram - Data Dictionary.

Data Queries: Query Basics-Computations - Subtotals and GROUP BY- Multiple Tables

UNIT – III Advanced Queries and Sub queries: OUTER JOINS – SQL SELECT-
SQL Data Definition

Commands – SQL Data Manipulation Commands. Forms, Reports
and Applications: Effective Design of Reports and Forms - Form
Layout - Creating Forms - Direct Manipulation of Graphical
Objects - Reports – Application features

UNIT – IV Database Integrity and Transaction: Procedural Languages - Data
Triggers- Transactions - Multiple Users and Concurrent Access -
ACID Transactions - Data Warehouses and Data Mining: Indexes
- Data Warehouses and online Analytical Processing - OLAP
Concepts OLAP Database Design - Data Mining

UNIT – V Database Administration: Data Administrator - Database
Administrator- Database Structure – Metadata -
Database Tasks by Development Stages - Backup and Recovery
Security and Privacy - Encryption.

Distributed Databases and the Internet: Distributed Databases -
Client/Server Databases - Electronic Commerce
Databases - The Web as a Client/Server System - Data
Transmission Issues in Applications

BOOKS FOR STUDY:

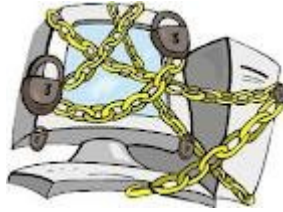
1. DATABASE MANAGEMENT SYSTEMS(Third Edition) By Gerald V. Post, Tata McGraw-Hill- Third Edition

BOOK FOR REFERENCE:

1. AN INTRODUCTION TO DATABASE SYSTEMS BY C.J.DATE, Narosa Publishing House 1990.
2. PRICIPLES OF DATABASE MANAGEMENT SYSTEMS BY Woolman and Hopcroft.
3. DATABASE MANAGEMENT SYSTEMS BY Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill, 2000.



INFORMATION SECURITY



Information security is the process of protecting information. It protects its availability, privacy and integrity. Access to stored information on computer databases has increased greatly. More companies store business and individual information on computer than ever before. Much of the information stored is highly confidential and not for public viewing.

Many businesses are solely based on information stored in computers. Personal staff details, client lists, salaries, bank account details, marketing and sales information may all be stored on a database. Without this information, it would often be very hard for a business to operate. Information security systems need to be implemented to protect this information.

Effective information security systems incorporate a range of policies, security products, technologies and procedures. Software applications which provide firewall information security and virus scanners are not enough on their own to protect information. A set of procedures and systems needs to be applied to effectively deter access to information.

There are people who make a living from hacking or breaking through information security systems. They use their technological skills to break into computer systems and access private information. Firewalls, which are designed to prevent access to a computer's network, can be bypassed by a hacker with the right hardware. This could result in the loss of vital information, or a virus could be planted and erase all information. A computer hacker can gain access to a network if a firewall is shut down for only a minute.

UNIT - 1 Introduction: Security-Attacks-Computer Criminals-Method of defines Program Security: Secure Programs-Non-malicious program Errors- Viruses and other malicious code-Targeted malicious code Controls against program threats.

UNIT – 2 Operating System Security: Protected objects and methods
Protection-Memory address protection-Control of access to
General objects-File protection mechanism- Authentication:
Authentication basics – Password-Challenge - response-
Biometrics.

UNIT – 3 Database Security: Security requirements-Reliability and
Integrity-Sensitive data-interface-Multilevel database-Proposals
For multilevel security.

UNIT – 4 Security in Networks: Threats in networks-Networks security
Control-Firewalls-Intrusion detection systems-Secure e-mail-
Networks And cryptography-Example protocols: PEM- SSL- Ipsec.

UNIT – 5 Administrating Security: Security planning-Risk analysis –
Organisational security policies-Physical security-Legal-Privacy
and Ethical Issues in computer Security – Protecting programs
and data - Information and law-Rights of employees and
employers-Software Failures- Computer crime-Privacy-Ethical
issues in computer society - Case studies of ethics.

Recommended Texts:

1. C.P. Pfleeger, and S.L. Pfleeger, Security in computing, Pearson Education, 4th Edition, 2003.
2. Matt Bishop, Computer Security: Art and Science, Pearson Education, 2003.

Reference Books

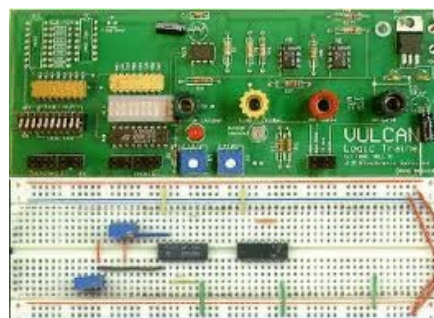
1. Stallings, Cryptography And Networks Security: Principles and practice, 4th Edition, 2006.
2. Kaufman, Pearlman, Speciner, Network Security, Prentice Hall, 2nd Edition, 2003.
3. Eric Maiwald, Network Security: a Beginner's Guide , TMH, 1999.
4. Macro Pistoia, Java Network Security, Pearson Education, 2nd Edition, 1999.
5. Whitman, Mattord, Principles of Information Security, Thomson, 2nd Edition, 2005.

MULTIMEDIA

Animation



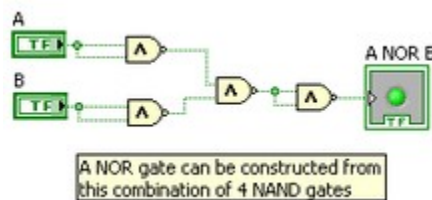
DIGITAL COMPUTER FUNDAMENTALS



The basic organization of the various components and working principles of the computer is also clearly expressed with hands-on experience to know more about the *internal architecture*.

Electronic logic gates differ significantly from their relay-and-switch equivalents. They are much faster, consume much less power, and are much smaller (all by a factor of a million or more in most cases). Also, there is a fundamental structural difference. The switch circuit creates a continuous metallic path for current to flow (in either direction) between its input and its output. The semiconductor logic gate, on the other hand, acts as a high-gain voltage amplifier, which sinks a tiny current at its input and produces a low-impedance voltage at its output. It is not possible for current to flow between the output and the input of a semiconductor logic gate.

Another important advantage of standardized integrated circuit logic families, such as the 7400 and 4000 families, is that they can be cascaded. This means that the output of one gate can be wired to the inputs of one or several other gates, and so on. Systems with varying degrees of complexity can be built without great concern of the designer for the internal workings of the gates, provided the limitations of each integrated circuit are considered.



UNIT-I Binary Systems: Digital Computers and Digital Systems – Binary Numbers – Number Base Conversion – Octal and Hexadecimal Numbers – Complements – Binary Codes – Binary Storage and Registers

UNIT-II Boolean Algebra & Logic Gates: Basic Definitions – Axiomatic Definition of Boolean Algebra – Basic Theorems and Properties and Boolean Algebra – Boolean Functions – Canonical and Standard Forms – Other Logic Operations – Digital Logic Gates.

UNIT-III Simplification of Boolean Functions: The Map Method – Two and Three Variable Maps – Four Variable Map – Product of sums Simplification – NAND and NOR implementation – Don't care Conditions.

UNIT-IV Combinational Logic : Introduction – Design Procedure – Adders – Sub tractors – Code Conversion – Analysis Procedure – Multilevel NAND Circuits – Multilevel NOR Circuits – Exclusive – OR and Equivalence Functions. Combinational Logic with MSI and LSI: Introduction – Binary Parallel Adder - Decimal Adder – Decoders – Multiplexers – Read only Memory (ROM).

UNIT-V Sequential Logic: introduction – Flip Flops – Triggering of Flip Flops – Flip Flops Excitation Tables- Design Procedure – Design of Counters – Design of registers and RAM

BOOKS FOR STUDY:

1. Digital Logic and Computer Design, M Morris Mano, PHI Pvt Ltd., New Delhi,, 2002.

BOOKS FOR REFERENCE:

1. Digital Computer Fundamentals, Thomas C Bartee, Tata McGraw Hill 1991
2. Digital Computer Design, V. Rajaraman, T Radha Krishnan, PHI , 2001.
3. Digital Fundamentals, Thomas L Floyd, Universal Book Stall, New Delhi III Ed.2001.



09vcDigital.flv

MICROPROCESSOR AND ITS APPLICATIONS



A microprocessor, sometimes called a *logic chip*, is a computer processor on a microchip.

The microprocessor contains all, or most of, the central processing unit (CPU) functions and is the "engine" that goes into motion when you turn your computer on. A microprocessor is designed to perform arithmetic and logic operations that make use of small number-holding areas called *registers*. Typical microprocessor operations include adding, subtracting, comparing two numbers, and fetching numbers from one area to another. These operations are the result of a set of instructions that are part of the microprocessor design.

When your computer is turned on, the microprocessor gets the first instruction from the basic input/output system (BIOS) that comes with the computer as part of its memory. After that, either the BIOS, or the operating system that BIOS loads into computer memory, or an application program is "driving" the microprocessor, giving it instructions to perform.

UNIT – I Microprocessor Architecture and its operations : Microprocessor Initiated Operations and the 8085 Bus Organization – Internal Data Operations and the 8085 Registers – Peripheral or Externally Initiated Operations.

UNIT – II **The 8085 MPU:** The 8085 Microprocessor – Microprocessor Communication and Bus Timings – Demultiplexing the Bus – Generating Control Signals – A Detailed Look at the 8085 MPU and Its Architecture – Decoding and Executing an Instruction.

UNIT – III **Instruction Set Of 8085:** Origin of Software – Processor Cycle – Machine cycles – Instruction Format – Addressing modes – Instruction Set – Data Transfer Instructions – Arithmetic Instructions – Logical Instructions – Branching Instructions – Machine Control Instruction – Timing diagram.

UNIT – IV **Programs:** Data Transfer – Arithmetic Operations – Biggest – Square root – Square – Factorial – Sorting of Numbers – Factorial.

Programs: Code Conversion – Hex to BCD – BCD TO hex – ASCII to Binary – Binary to ASCII – BCD to ASCII – ASCII to BCD – Counting the occurrences to the given number.

UNIT – V Memory and I/O Interfacing: I/O Structure of a typical microcomputer – Interfacing I/O devices and peripheral IC`s.

BOOKS FOR STUDY :

1. Microprocessor and Its Applications, A NagoorKani, RBA Publications, 1999.

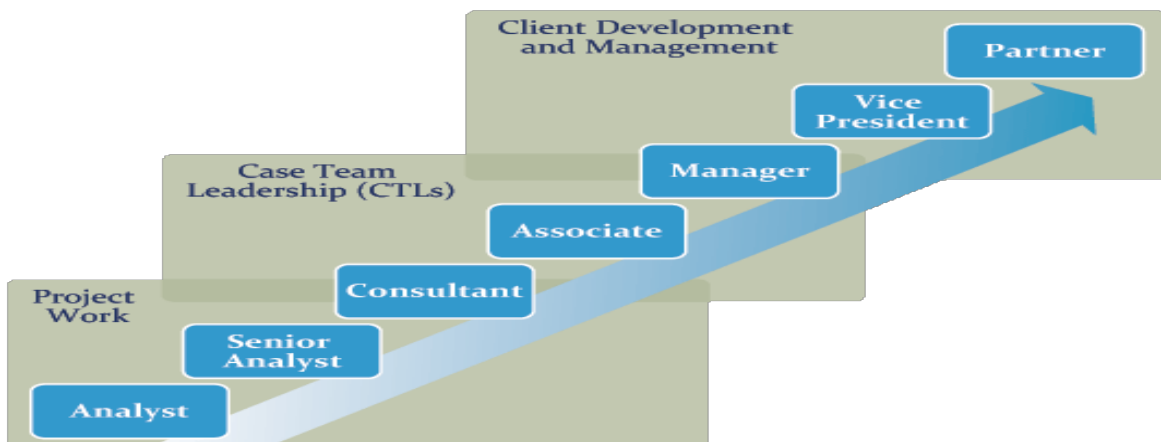
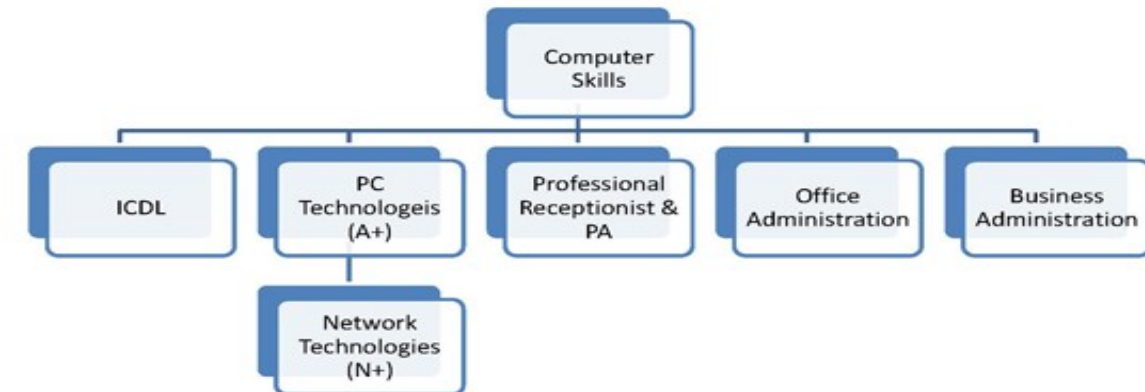
BOOKS FOR REFERENCE:

1. Microprocessor Architecture, Programming and applications with the 8085, Ramesh S Gaonkar, Penram International, 4th Edition.
2. Microprocessor and its Applications, R Theagarajan, S Dhanapal, S Dhanasekaran, New Age International (P) Ltd.,
3. Fundamentals of Microprocessor 8085 Architecture, Programming & Interface, V.Vijayendran, SIV Publishers.



10vcMP 8085.flv

CAREER OPPORTUNITIES



- Career Path 1: Designing and implementing software. This refers to the work of software development which has grown to include aspects of web development, interface design, security issues, mobile computing, and so on. This is the career path that the majority of computer science graduates follow. While a bachelor's degree is generally sufficient for entry into this kind of career, many software professionals return to school to obtain a terminal master's degree. (Rarely is a doctorate involved.) Career opportunities occur in a wide variety of settings including large or small software companies, large or small computer services companies, and large organizations of all kinds (industry, government, banking, healthcare, etc.). Degree programs in software engineering also educate students for this career path.
- Career Path 2: Devising new ways to use computers. This refers to innovation in the application of computer technology. A career path in this area can involve advanced graduate work, followed by a position in a research university or industrial research and development

laboratory; it can involve entrepreneurial activity such as was evident during the dot-com boom of the 1990s; or it can involve a combination of the two.

- Career Path 3: Developing effective ways to solve computing problems. This refers to the application or development of computer science theory and knowledge of algorithms to ensure the best possible solutions for computationally intensive problems. As a practical matter, a career path in the development of new computer science theory typically requires graduate work to the Ph.D. level, followed by a position in a research university or an industrial research and development laboratory.
- Career Path 4: Planning and managing organizational technology infrastructure. This is the type of work for which the new information technology (IT) programs explicitly aim to educate students.

