COURSE CURRICULUM

M.Sc. (Computer Science) /
M. Sc. (Information Technology)

I & II Semester

DEVI AHILYA UNIVERSITY, INDORE (M.P.)
(July – 2007)

M.Sc.(CS) / IT
Course Curriculum

I Semester

Course Code	Title	Scheme of marks		Total
		Theory	Practical	
CS- 401	Computer & Communication Fundamentals	50		
CS-411	Discrete Structures	50		
CS 422	Operating Systems	50	50	100
CS-424	Programming and Problem Solving through C	50	50	100
IC-492	Communication Skills	50		
Total		250	100	350

II Semester

Course Code	Title	Scheme of marks		Total
		Theory	Practical	
CS 402	Computer Architecture and Organisation	50		
CS-422	Data Structures using C++	50	50	100
CS-431	Software Engineering	50		
CS-441	Database Management System	50	50	100
CS – 561	Computer Networks	50		
IC-491	Organization and Management Concepts	50		
Total		300	100	400

Scheme for award of degree shall be same as for MBA (CM).

I SEMESTER

CS- 401 Computer and Communication Fundamentals

UNIT- I	Computer Organization	6 hrs.
Computer Organization : Digital and Analog computers, Major components of a digital computer, Memory addressing capability of a CPU, Word length of a computer, Processing speed of a CPU, Definitions of Hardware, Software and Firmware. Definitions of Dumb, Smart and Intelligent terminals. Binary Systems : Digital Systems, Binary Numbers, Number Base Conversions, Octal and Hexadecimal Numbers, Complements, Signed Binary Numbers, Binary Codes : BCD code, Gray Code, ASCII code, Excess 3 Code, Error detecting Code.		

UNIT- II	Computer Arithmetic	8 hrs.
Computer Arithmetic :Binary representation of Negative Integers using 2's complement and Signed magnitude representation, Fixed point Arithmetic operations on Positive and Signed (Negative) Integers like addition, subtraction, multiplication, Booth algorithm for multiplication, and bit pair(fast) multiplication. Division of positive and negative binary numbers, restoring and non storing algorithm, Floating Point Numbers (IEEE 754 standard) and their representation, NaNs and demoralized numbers.		

UNIT- III	Boolean Algebra and Logic Gates	6 hrs.
Boolean Algebra and Logic Gates : Basic Definitions, Basic Theorems and properties of Boolean Algebra, Boolean Functions, Canonical and standard forms, Other Logic operations, Digital Logic gates, Integrated Circuits. Gate-Level Minimization: The K-Map Method, 3 and 4 variable K-Map, Product of sums simplification, Sum of Products simplification, Don't care conditions, NAND and NOR implementations, Exclusive-OR function.		

UNIT- IV	Combinational Logic	8 hrs.
Combinational Logic: Combinational Circuits, Analysis Procedure, Design Procedure, Binary half adder, binary full adder , binary full subtractor, binary parallel adder, carry propagation delay and Propagation delay calculation of various digital circuits. Carry look ahead generator fast adder ,Decimal Adder, Binary multiplier, Magnitude comparator, Code converters like binary to gray, BCD to excess 3. Decoders, Encoders, Multiplexers, Demultiplexers.		

UNIT- V	Analysis of clocked sequential circuits	6 hrs.
Analysis of clocked sequential circuits: State diagrams, State equations for D, JK and T Flip flops. State reduction methods using all Flip Flops. Mealy and Moore Models. Shift Registers-Serial in Serial out, Serial in Parallel out, Parallel in Serial out and Parallel in Parallel out. Designing of Asynchronous (Ripple) Counters, Design of Synchronous Counters. Synchronous Sequential logic : Sequential circuits, Latches, Flip Flops : SR, D, JK, T. Master Slave JK Flip flop. Characteristic equations and Excitation tables of flip flops.		

UNIT- VI	Communication Systems	6 hrs.
Basics of communication systems, Types of communication, Transmission impairments, analog vs. digital transmission, requirements of communication systems, channel capacity. Shannon's theorem. Data rate of a channel, Physical Communication Media- Bounded Media : Twisted Pair, Coaxial Cable, Optical Fiber. Unbounded Media – Microwave Communication, Radio wave Communication, Satellite Comm. Time Division Multiplexing and Frequency Division Multiplexing. Data communications and its components, Half Duplex and Full Duplex Transmission. Asynchronous and synchronous transmission LAN ,MAN,WAN. Network Topologies- Bus, Star, mesh , Ring. Categories of networks: Introduction of Communication Protocols like OSI and TCP/IP model.		

Required Text(s) :

1. Digital Design by M. Morris Mano. Third addition
2. Computer Architecture By Dr. Rajkamal.
3. Data communications and networking By A. Forouzan
4. Computer Fundamentals – Architecture and Organisation By B. Ram.
5. Computer networks by Andrew Tanenbaum
6. Principles of digital communication system & computer networks By K.V.K.K. Prasad
7. Computer organization and architecture by William Stallings.

Laboratory Assignment

S. No.	Name of the Experiment
1.	Study of Digital Logic gates with Identification numbers and Pin assignments.
2.	Study of IC type 7493 Ripple Counter and listing of ICs required for experiments
3.	Study of Binary counter using IC 7493
4.	Study of Boolean function implementation using universal logic gates.
5.	Study of combinational circuit like Parity Generator
6.	Study of IC type 74155 as 3 x 8 Decoder
7.	Study of Code converter BCD to Seven Segment Decoder (7447) and Seven Segment Display (7730)
8.	Study of IC type 74151 8 x 1 Multiplexer
9.	Study of IC type 7483 4 -Bit Binary Adder and as 4-bit Adder Subtractor
10.	Study of Magnitude Comparators
11.	Study of IC type 7474 dual Positive Edge triggered Flip Flop
12.	Study of IC type 7476 dual JK Master Slave flip flops
13.	Study of IC type 74161 Binary Counter with Parallel Load
14.	Study of IC type 74195 Shift Register with Parallel Load
15.	Study of IC type 74189 16 x 4 RAM

CS - 411 Discrete Structures

UNIT- I	The Foundations: Logic , Sets and Functions	6 hrs.
Introduction, logic, prepositional equivalences, predicates and quantifiers. **Logic , Sets and Functions:** Sets, set operations, fuzzy sets, functions, functions for computer science, sequences and summations		
UNIT- II	Mathematical reasoning	8 hrs.
Methods of proof , mathematical induction. Recursive definitions, recursive algorithms.		
UNIT- III	Combinatorics	6 hrs.
The basics of counting, The Pigeonhole Principle, Permutations and Combinations. Advanced counting techniques, recurrence relations, solving recurrence relations, Algorithms, Complexity of algorithms.		
UNIT- IV	Relations	8 hrs.
Relations and their properties, n-ary relations and their applications, representing relations, closures of relations , equivalence relations , partial orderings. **Relations :** Relations and their properties, n-ary relations and their applications, representing relations, closures of relations , equivalence relations , partial orderings. Equivalence relations , partial orderings and Lattices,Chains and Antichains.		
UNIT- V	Graphs	6 hrs.
Introduction to Graphs, Graph terminology, representing graphs and graph isomorphism. Connectivity, Euler and Hamiltonian Paths, shortest path problems, planar graphs, graph colouring, chromatic number, Euler's formula, Kuratowski's theorem. The four colour problem, Applications of Graph Colouring, Introduction to Trees, applications of trees, tree traversal, trees and sorting, Spanning trees, minimum spanning trees.		
UNIT- VI	Languages and Grammars	6 hrs.
Introduction to Languages and Grammars. Phrase-Structure Grammars,Types of Phrase structure grammars.		

Required Text(s)-

1. Kenneth H. Rosen "Discrete Mathematics and its Applications" , 4th edition , Tata McGraw-Hill Edition.Let us C, Yashavant Kanetkar, BPB Publications.
2. C.L. Liu, "Elements of Discrete Mathematics".
3. Kolman, Busby & Ross "Discrete Mathematical Structures" ,5th edition Pearson Education.
4. Trembly. J.P. & Manohar. P "Discrete Mathematical Structures with Applications to computer Science"

CS - 422 Operating Systems

UNIT- I	Processor Management	6 hrs.
Introduction: Evolution of operating systems, operating system concepts, operating system services, System Calls. Processor Management : Concepts, Algorithms for batch processing, Algorithms for time sharing operating systems, Introduction to real time systems		

UNIT- II	Memory Management	8 hrs.
Concepts, single user memory management, Partition memory allocation, Virtual memory management using paging and segmentation techniques		

UNIT- III	Concurrent Processes	6 hrs.
Mutual exclusion and synchronization, Techniques of inter process communication, Deadlock handling		

UNIT- IV	File Management	8 hrs.
Operations on a file, structure of a file system, Free block list, keeping track of blocks allocated to a file, directory structure, sharing and protection of files, file system Reliability, Unix file system		

UNIT- V	Device Management	6 hrs.
Device Management: Goals of input/output software design, Structure of device hardware and software, Layers of I/O software, structure of device drivers, Disk driver, disk arm scheduling algorithms, terminal driver, clock driver etc.		

UNIT- VI	Network and Distributed Operating Systems	6 hrs.
Introduction to network and distributed operating systems. Case Studies: Unix/Linux, Windows operating system		

Required Text(s):-

1. *Operating System Concepts,* Addison Wesley, 4th Edition, A. Silberschatz and P. Galvin. 1994.
2. *Modern Operating System*, A.S Tanenbaum., Prentice Hall of India
3. Operating systems, 4rth Edition, William Stallings, Pearson Education, 2003.

Laboratory Assignment

Assignment no.1
1. Simulate an environment in which jobs arrive at a processor for execution. The jobs execution time follows:
   (a) Random execution time and random inter-arrival time distribution.
   (b) Poisson's distribution and inter-arrival time follows exponential distribution.
2. Develop the algorithms to implement following processor scheduling algorithms:
   (a)First come first served   (d)Shortest job first
   (b)Round robin   (e)Multi level feedback queue.
   (c)Selfish round robin
3. Compare the performance of above algorithm on a graph for following parameters: (a) Response ratio (b) Utilization of central processing unit.

Assignment no. 2

1. Compare the performance of following page replacement algorithms:
    (a) Random   (c) Optimal page replacement
    (b) First come first served   (d) Least recent used
    The performance must be compared using page fault rate graph.

2. Compare the following variations of LRU page replacement:

    (a) Not recently used   (c) Least frequently used
    (b) Second chance replacement   (d) Most frequently used
    The performance must be compared using page fault rate graph.

3. Compare the memory and processing time requirements of following sorting algorithms using a graph for different sizes of input lists:

    (a) Insertion sort   (d) Merge sort
    (b) Selection sort   (e) Heap sort
    (c) Quick sort

4. Compare the memory and processing time requirements of linear search and binary search techniques using a graph for different sizes of input lists.

Assignment no 3:

1. Study different keyboards and mouse present in the lab. Find the specifications and different character sets used.
2. Study various types of printers and scanners and find data transfer rate and other specifications.
3. Study different types of hard disks and their specifications. Calculate seek time and latency time.
4. Study following I/O devices:
    (a) CD/DVD ROM
    (b) Pen drive

Assignment no 4:

1. Study Windows operating system in terms of :
    (a) Processor management   (c) File management
    (b) Memory management   (d) Inter process communication

Assignment no. 5

1. Study Unix / Linux operating system in terms of:
    (a) Processor management   (c) File management
    (b) Memory management   (d) Inter   process   communication

    2. Practice the following Unix commands and their variations:
       ls, cp, mv, chmod, grep, who, tty, comp, comm., tee, awk, sort, date, cal, wc, test, expr etc.

CS - 424 Programming and Problem Solving through C

UNIT- I	Overview of Problem Solving	8 hrs.
Introduction to Computer based Problem Solving, Programming Concepts with Flowcharting and algorithms, Classification of programming languages, Programming Environment {Assemblers, compilers, interpreters, linkers, and loaders}, Developing and debugging flowcharts for Programming Problem.		
UNIT- II	Fundamentals of C Programming	7 hrs.
Overview of C: Various Constructs of a C Program, Coding style, Data types, Constants and Variables, Expressions and Operators, Basic Input/ Output operations, Formatting Characters, Decision making and Branching, Looping constructs.		
UNIT- III	Arrays and their Applications	10 hrs.
Arrays {one dimensional and multidimensional array}, String Handling, Searching and sorting techniques, Matrices Operations		
UNIT- IV	Advanced Programming Concepts	9 hrs.
Structures and Union, Functions {Standard and User defined Function, Parameter passing, Scope Rules}, Recursion {Using Recursion, Conversion of recursive programs into non-recursive}, Dynamic Memory allocation and pointer {Uses, pitfalls, pointers to various user-defined and standard data types}.		
UNIT- V	More Advanced Programming Concepts	8 hrs.
Pre-Processors: {Define, include, macro's, ifdef,…}, Introduction to File handling, Advanced pointer {Data Structures}		
UNIT- VI	Remaining from C	8 hrs.
Arguments to main, Enumerations and bit fields, Coercion, typedef, Type Casting, Static, global, external, register, Header files creation, Other Programming Techniques.		

Required Text(s)

1. Herbert Schildt, "C++ The Complete Reference", Osborne/McGraw-Hill
2. Let us C, Yashavant Kanetkar, BPB Publications.
3. B.W. Kernighan & D.M. Ritchie, "The C Programming Language", Prentice Hall of India.

Laboratory Assignments

1. Write a C program to display "Hello Computer" on the screen.
2. Write a C program to display Your Name, Address and City in different lines.
3. Write a C program to find the area of a circle using the formula: Area = PI * r2
4. Write a C program to find the area and volume of sphere. Formulas are: Area = 4*PI*R*R Volume = 4/3*PI*R*R*R.
5. Write a C program to print the multiply value of two accepted numbers.
6. Write a C program to convert centigrade into Fahrenheit. Formula: C= (F-32)/1.8.
7. Write a C program to read in a three digit number produce following output (assuming that the input is 347)
    3 hundreds
    4 tens
    7 units

8. Write a C program to read in two integers and display one as a percentage of the other. Typically your output should look like

20 is 50.00% of 40

assuming that the input numbers where 20 and 40. Display the percentage correct to 2 decimal places.
9. Write a C program to find out whether the character presses through the keyboard is a digit or not (using conditional operator).
10. Write a C program to swap variable values of i and j.
11. Write a C program to find the maximum from given three nos.
12. Write a C program to find that the accepted no is Negative, Positive or Zero.
13. Write a program which reads two integer values. If the first is lesser print the message up. If the second is lesser, print the message down if they are equal, print the message equal if there is an error reading the data, print a message containing the word Error.
14. Write a C program that prints the given three integers in ascending order using if – else.
15. Given as input three integers representing a date as day, month, year, print the number day, month and year for the next day's date.

Typical input: "28 2 1992" Typical output: "Date following 28:02:1992 is 29:02:1992".
16. Write a C program for calculator designing using switch /case loop?
17. Write a C program to convert decimal to binary.
18. Write a C program to convert decimal to octal.
19. Write a C program to convert decimal to hexadecimal.
20. Write a C program to find the sum of first 100 natural nos.
21. Write a C program to find the sum of first 100 odd nos. and even nos.
22. Write a C program to display first 25 Fibonacci nos.
23. Write a C program to display first 100 prime nos.
24. Write a C program to find factorial of accepted nos.
25. Write a C program to find the sum of digits of accepted no.
26. Write a C program to print the accepted no and its reverse no.
27. Write a C program to print all the Factors of accepted no.
28. Write a C program to find HCF of two given numbers.
29. Write a C program to find LCM of two given numbers.
30. Write a C program to find all the prime number between two given numbers.
31. Write C programs to print the terms of each of the following series:

i. Sin(x)   ii. Cos(x)   iii. Log (1+x)   iv. log(1-x)   v. e^{x} vi. e^{-x}
32. Write a C program to print the sum of above series.
33. Display the following output on screen (assuming the value for input parameter n=5) :

a.	b.	c.	d.	e.
*	1	A	1	1
**	12	AB	23	23
***	123	ABC	345	456
****	1234	ABCD	4567	78910
*****	12345	ABCDE	56789	101112131415

f.	g.	h.	i.	j.
*****	ABCDE	*	1	1
****	ABCD	***	123	121
***	ABC	*****	12345	12321
**	AB	*******	1234567	1234321
*	A	*********	123456789	123454321

k.	l.	m.	n.	o.
* ** *** **** *****	ABCDE ABCD ABC AB A	1 12 123 1234 12345	***** 0000 *** 00 *	1 10 101 1010 10101

p.	q.	r.	s.	t.
1 01 101 0101 10101	1 22 333 4444 55555	A AB ABC AB A	ABCDEDCBA ABCD DCBA ABC CBA AB BA A A	1 121 12321 1234321 123454321

34. Write a C program to find minimum, maximum, sum and average of the given one dimensional array.
35. Write a C program to perform the basic Matrix operations   addition, subtraction, multiplication, Transpose.
36. Write a program to take a sentence as input and reverse every word of the sentence.
37. Write a C Function for the following task
   a. Calculating Factorial
   b. Find value of a given Fibonacci term
   c. Swapping the values of two variable
   d. Minimum/maximum value from the given input
38. Write User Defined Function and test them in the main program for the following standard function
   a. int myatoi(Char *s)
   b. char *myitoa(int i)
   c. int mystrlen(char *s)
   d. char *mysubstr(char *s, int i, int j)
   e. char *mystrcat(char *s1,char *s2)
   f. int mystrcmp(char *s1, char *s2)
   g. int mystrchr(char *s, char c, int i)
   h. char *mystrrev(char *s)
   i. int mystrend(char *s, char *t)
   j. char *myreplace(char *s, char *old, char *new)
   k. int abs(int i)
   l. char *mytoupper(char *)
   m. char *mytolower(char*)
   n. int isupper(char *s)
   o. int islower(char *s)
   p. int mypower(int a, int b)
   q. int mymod (int a, int b)
39. Write the following recursive C Function
   a. Factorial of a given number
   b. Nth Fibonacci number
   c. Reverse of a given String
   d. Reverse of a give Number
   e. Sin(x)
40. Write a c program to create a new data type Date with the help of structure and typedef. Also write following user defined function for date manipulation.
   a. To return next Date,

        b. To return next Month,

        c. To return next Year,

        d. To add few Days in a date

        e. To add few Months in a date

        f. To add few Years in a date

        g. To return the date of the week of a given date.

        h. To return Month name from the date.

        i. To Display the Date in various format as: Date Display(Date d1, char *format)

               Here Date is the newly created data type. The format string can hold the following values: "DDMMYYYY", "MMDDYY", "MON, DD, YYYY"

41. Write a C program to implement myprintf and myscanf functions using Concept of variable number of arguments.

(using getch, putch, gets and puts function)

42. Write a C program that creates an Employee text file? Records Are empid, empname, designation, qualification, salary, experience, Research work, address, city phone?

43. Write a C program that manipulates the above text file. The program must implements the operation to modify a record, delete a record and append new records.

44. Write C programs for the following operation to work like DOS Commands:

        a. type abc.txt

        b. copy source1.txt source2.txt

        c. copy source1.txt source2.txt source3.txt source4.txt

        d. compare source1.txt source2.txt

        e. concat source1.txt source2.txt

45. Write a C program to open two files containing integers (in sorted order) and merge their contents.

46. Write a C program to count the number of vowels, consonants, digits, spaces, other symbols, words and lines in a given text file.

47. Write C code to check if an integer is a power of 2 or not in a single line?

48. Write a C program to count bits set in an integer?

49. Write a C program to fast multiply by seven.

50. Write a C program to set a particular bit in a given number.

51. Write a C program to reset a particular bit in a given number.

IC - 492 COMMUNICATION SKILLS

Unit	Contents	Hours
I	**Fundaments of Communication (OHP & PPP):** Definitions, importance, forms of communication, process of communication, channels, barriers and strategies to overcome barriers of communication.	06
II	**Listening (PPP):** Def, Importance, Benefits, barriers, approaches, be a better listener, exercises and cases.	04/
III	**Advance Communication (PPP and Exercises on handouts)** Why communication? Art of communication, V3 communication, Key elements of IP communication, Quizzes, exercises and cases / incidents for practice.	10
IV	**Group Discussions:(PPP)** Definitions, importance, process, points to be borne in mind while participating, Dos and Don'ts. Practice- if time permits or to be covered in PDP.	06/
V	**Interview (PPP)** Types of, Points to be borne in mind as an interviewer or an Interviewee, commonly asked questions, Dos and Don'ts. Practice- if time permits or to be covered in PDP.	06/
VI	**Transactional Analysis: (PPP)** Transactional analysis, Johari Window, FIRO-B (PPP)	06/
VII	**Written Communication:** Report writing, documentation, business correspondence, preparation of manuals and project reports.	08
VIII	Tests and presentations (preferably on Sundays / holidays)	08/
	Total Hours	**54/**
Text Readings	1. OB by Fred Luthans 2. OB by Stiphen P. Robbins 3. Masterson, Johan & et. al (1989), "Invitation to Effective Speech Communication, Scott, Foreman and Co. 4. Chturvedi, P.D. and Chaturvedi Mukesh (2004), "Business Communication"Pearson Education, Singapore Pvt. Ltd.	
Reference Readings	1. Business Communication by ICMR, Feb 2001. 2. Toropov Brandon (2000), "Last Minute Interview Tips", Jaico Publishing House, Mumbai. 3. Heller Robert (1998), "Essential DK Managers: Communication Clearly", Dorling Kindersley, London. 4. Decker Bert ( ) "The Art of Communication", 5. Bone Diane (),"The Business of Listening",a Fifty-Minute Series Book Crisp Publications, Inc, California.	

II SEMESTER

CS - 402 Computer Architecture and Organisation

UNIT- I		6 hrs.
Technological trends, measuring performance, Andahl's law. Basic structure of computer hardware: Functional units and components in computer organization: The memory unit, the input and output subsystem, the bus structures, ALU. Program development tools: Compiler, interpreter, assembler.		
UNIT- II		8 hrs.
Instruction sets and processor organizations., Instruction and Interrupt Cycles, Instruction sequencing, Addressing Modes. Instruction. 8088 microprocessor: Architecture. 8088 Assembly language programming.		
UNIT- III		6 hrs.
Processing unit design: Processor micro architecture I - fundamental concepts of data path implementation. Processor micro architecture II- data path implementation, Hardwired control unit, micro programmed Execution.		
UNIT- IV		8 hrs.
Instruction pipelining and parallel processing : Instruction pipelining hazards, Instruction set design influence on pipelining, Example of a pipelined CISC & RISC processor.		
UNIT- V		6 hrs.
Instruction level parallelism: VLIW Processors, Vector processors, Multithreaded processors, Extracting parallelism. Caches: Data Caches, instruction caches, Unified caches. Cache implementations, multilevel caches.		
UNIT- VI		6 hrs.
Virtual memory : Organization, Mapping functions for translating the program pages in virtual to physical address space, Cache and virtual memory		

Required Text(s):

1. Rajkamal, Computer Architecture: Scahaum's outlines by Nicholas Carter Adapted Computer Organization by Hamecher.
2. Computer Organization & Architecture by William stallings
3. Computer Architecture & Parallel Processing, Hwang & Briggs, McGraw Hill

LABORATORY ASSINMENT FOR 8088 MICRO PROCESSOR:
ASSIGNMENT-1
1) Write a program to add two 8 bit numbers. And stores the result at memory location 2005.
2.) A program to add two 16 bit numbers. Store the result in memory address starting from 2000.
3) Write a program which tests if any bit is '0' in a data byte specified at an address 2000. If it is so, 00 would be stored at address 2001 and if not so then FF should be stored at the same address.
4) Write a program to perform 8 bit multiplication
5) Write a program to perform division of two numbers.

ASSIGNMENT-2
1) Write a program, which loads registers, A, B, C and D with an identical constant [e.g. $C0]. Try to optimize the program in such a way that the smallest number of program bytes is used. Test the program in single-step mode. After each step test, the register(s) of interest.
2) Assume that 3 bytes of data are stored at consecutive memory addresses of the data-memory starting at 2000. Write a program which loads register C with (2000), i.e. with data contained at memory address

2000, D with (2001), E with (2002) and A with (2001).
(i)Use register indirect addressing, and
(ii) Use direct addressing mode (LDA instruction), and
Compare (i) and (ii) in terms of memory requirement.
3) Write a program which checks the given number for odd and even parity.
4) Write a program that will count number of one's in given data
5) Write a program that will check that given data byte is even or odd. If number is odd stores it at 2000 otherwise store it at 2001 memory location.

ASSIGNMENT-3
1 .Sixteen bytes of data are specified at consecutive data-memory locations starting at 2000. Write a program which increments the value of all sixteen bytes by 01.
2. Eight bytes (unsigned binary numbers) with upper nibble zero in each byte are stored at consecutive data-memory address starting 2000. Compute the sum of the eight bytes and store it at an address 3000.
3. Two data-bytes are stored at addresses 2000 and 2001. Interchange the data bytes at these two addresses using (i) indirect addressing and (ii) direct addressing. Execute the two programs and get the identical results. Also find the difference in memory space used by these two programs.
4. Write a program, which test if the all-odd bits are '1' in a data byte at a data-memory location 2000. If it is so store 01 at 2001, else stores 00 at the same location.
5. Write a program to find out two's complement of a given data byte.

ASSIGNMENT-4
1. Write a program to compare the two numbers stored at 2000 and 2001 if the number that is stored on 2000 is grater than store 00 at memory address 2005 otherwise store 01 at 2005.
2. Eight bytes are stored at consecutive data-memory address starting from 2000.write a program to add theses bytes. Store the result at memory address 3000.
3. Eight bytes are stored at consecutive data-memory address starting from2000.write a program to arrange these bytes in ascending and descending order.
4. Eight bytes are stored at consecutive data-memory address starting from 2000.write a program to find out the odd numbers from given bytes .And display the result in register A.

CS - 422 Data Structures using C++

UNIT- I	Introduction to Data Structures	(02hrs.)
Definition of data structures and abstract data types.Static and Dynamic implementations. Examples and real life applications, Data Structures: Arrays, Address calculation in a single and multi dimensional array. Sparse matrices		
UNIT- II	**Stacks, Queues and Lists**	**(16hrs.)**
Definition, Array based implementation of stacks, Linked List based implementation of stacks, Examples : Infix, postfix, prefix representation, Applications : Mathematical expression Evaluation

Definition: Queues & Lists: Array based implementation of Queues / Lists, Linked List implementation of Queues / Lists, Circular implementation of Queues and Singly linked Lists, Straight / circular implementation of doubly linked Queues / Lists, Priority queues , Applications | | |
| **UNIT- III** | **Trees & Graphs** | **(15hrs.)** |
| Definition of trees and Binary trees, Properties of Binary trees and Implementation, Binary Traversal - preorder, post order, inorder traversal, Binary Search Trees, Implementations, Threaded trees, Balanced multi way search trees, AVL Trees, Implementations, Applications

Definition of Undirected and Directed Graphs and Networks, The Array based implementation of graphs, Adjacency matrix, path matrix implementation, The Linked List representation of graphs, Shortest path Algorithm, Graph Traversal – Breadth first Traversal, Depth first Traversal, Connectivity of graphs; Connected components of graphs, Weighted Graphs, Applications. | | |
UNIT- IV	**Hashing**	**(2hrs.)**
Definition, Hash function, Collision Resolution Techniques, Hashing Applications		
UNIT- V	**Running time**	**(03hrs.)**
Time Complexity, Big – Oh - notation, Running Times, Best Case, Worst Case, Average Case, Factors depends on running time, Introduction to Recursion, Divide and Conquer Algorithm, Evaluating time Complexity		
UNIT- VI	**Sorting Searching Algorithms**	**(14hrs.)**
Introduction, Sorting by exchange, selection, insertions, Bubble sort, Selection sort, Insertion sort, Pseudo code algorithm and their C++ implementation, Efficiency of above algorithms, Shell sort, Performance of shell sort, Merge sort, Merging of sorted arrays, The merge sort Algorithms, Quick sort Algorithm, Analysis of Quick sort, Picking a Pivot, A partitioning strategy, Heap sort, Heap Construction, Heap sort, bottom – up, Top – down Heap sort approach, Radix sort, Straight Sequential Search, Array implementations, Linked List representations, Binary Search, non – recursive Algorithms, recursive Algorithms, Indexed Sequential Search		

Required Text(s)-

1. Data Structures using C by A. M. Tenenbaum, Langsam, Moshe J. Augentem, PHI Pub.
2. Theory & Problems of Data Structures by Jr. Symour Lipschetz, Schaum's outline by TMH
3. Data Structures and Algorithms by A.V. Aho, J.E. Hopcroft and T.D. Ullman, Original edition, Addison-Wesley, 1999, Low Priced Edition.
4. Fundamentals of Data structures by Ellis Horowitz & Sartaj Sahni, Pub, 1983,AW
5. Data Structures and Program Design in C By Robert Kruse, PHI

Laboratory Assignment

1. Write a program to search an element in a two-dimensional array using linear search.
2. Using iteration & recursion concepts write programs for finding the element in the array using Binary Search Method
3. Write a program to perform following operations on tables using functions only
   a) Addition b) Subtraction c) Multiplication d) Transpose
4. Using iteration & recursion concepts write the programs for Quick Sort Technique
5. Write a program to implement the various operations on string such as length of string concatenation, reverse of a string & copy of a string to another.
6. Write a program for swapping of two numbers using 'call by value' and 'call by reference strategies.
7. Create a stack and perform the valid operations on it.
8. Create a queue and circular queue and perform the valid operations on it.
9. Write a program to implement binary search tree. (Insertion and Deletion in Binary search Tree)
10. Write a program to create a linked list & perform operations such as insert, delete,update, reverse in the link list
11. Write the program for implementation of a file and performing operations such as insert, delete, update a record in the file.
12. Create a linked list and perform the following operations on it a) add a node b) Delete a node
13. Write a program to simulate the various searching & sorting algorithms and compare their timings for a list of 1000 elements.
14. Write a program to simulate the various graph traversing algorithms.
15. Write a program which simulates the various tree traversal algorithms.
16. Generate a list of the ten commonest words in a file. This requires a mapping of string keys to integer values, and the ability to extract the contents into a list of (string,integer) pairs which can be sorted based on the integer component of the pair.
17. Given a tree representing a hierarchy of windows in a GUI, build a list of "pointers" to selected tree nodes (e.g. all buttons in the hierarchy) so that it can be traversed to selectively update that particular set of windows (e.g. by changing the font used for the text of each button). This requires the ability to build arbitrary tree structures and to keep "pointers" to individual nodes outside the tree itself.
18. Implement an efficient sorting algorithm (e.g. quicksort) for a collection of values. This requires that the collection should allow for adequately efficient sorting without the need for a special "sort" primitive. The algorithm should be applicable to a variety of data structures (all proposed data structures?) without any changes. This requires some common protocol for accessing items in different data structures.
19. Generate the set of all possible permutations of a list of values. This requires the ability to recursively permute sublists of the original list.
20. Build a directed graph to represent an non-deterministic finite state automaton (NFA). This requires that the number of edges leading from a particular node can be made arbitrarily large, and the number of edges leading to a particular node can also be made arbitrarily large. Note that matching using an NFA requires sets of states to be manipulated.
21. Implement a stack using another more general data structure (e.g. a list), and use this together with a directed graph representing a maze to find a route through the maze.

Note: At least 5 to 10 more exercises to be given by the teacher concerned.

CS – 431 Software Engineering

UNIT- I	Introduction to Software Engineering & Software Processes	08 hrs.
Software problem, Software engineering problem, Software engineering approach, Software characteristics and Applications. Software processes and its components, characteristics of software processes, Software development processes: Linear Sequential model, Prototyping model, RAD model, Iterative Enhancement model, Spiral model, Component based development, Comparative study of various development models.		

UNIT- II	Project management process & Project Planning	08 hrs.
The people, product, process and project, Phases of project management process, Project life cycle, the W5HH principle. Software configuration management process, Process management process: Capability Maturity Model (CMM). Project estimation (Size & Cost), Project Scheduling, Staffing and personnel planning, Software configuration management plans, Quality assurance plans, Project monitoring plans, Risk management.		

UNIT- III	Software Requirement Analysis and Specification	06 hrs.
Software requirements, Problem analysis (Structured analysis and Object Oriented analysis), Requirements specifications, Validation and Verification, Metrics.		

UNIT- IV	Software Design	07 hrs.
Design principles: Problem partitioning and hierarchy, Abstraction, Modularity, Top-down and Bottom-up strategies. Effective Modular design: functional independency, Cohesion, Coupling. Structured design methodology.		

UNIT- V	Software Quality Assurance	05 hrs.
Quality concept, Quality management system, movements and assurance, Software reviews: formal and technical, Formal approaches to SQA, Statistical software quality assurance, Software reliability, ISO 9000, SQA plan.		

UNIT- VI	Software Testing	06 hrs.
Software testing techniques: Testing fundamentals, White box testing, Black box testing, Testing for specialized environments, architectures and applications. Software testing strategies: A strategic approach to software testing, Strategic issues, Unit testing, Integration testing, Validation testing and system testing, The art of debugging.		

Required Text(s)

1. *An Integrated Approach to Software Engineering*- **Pankaj Jalote**, Narosa Publishing House.
2. *Software Engineering-A practitioner's approach*- **R. S. Pressman**, Tata McGraw-Hill International Editions, New York.
3. *Software Engineering*- **Ian Sommerville**, Pearson Education, New Delhi.
4. *Software Engineering Concepts*-**Richard E. Fairly**, Tata McGraw Hill Inc. New York.
5. *Software Engineering: Principle & Practice*-**W. S. Jawadekar,** Tata McGraw-Hill, New York.
6. *Fundamentals of Software Engineering*-**Rajib Mall**, PHI, New Delhi.

CS - 441 Database Management Systems

UNIT- I	Fundamental of DBMS and RDBMS	8 hrs.
	Overview of Database Systems: Moving from File Systems to Databases, Different Data Models, Codes Rules, database system architecture ER- Diagram. Relational Data models: Domains, Tuples, Attributes, Relations, Characteristics of relations, Keys, Key attributes of relation, Relational database, Schemas, Integrity constraints, Intension and Extension.	
UNIT- II	Relational Query Language and Database Designing	8 hrs.
	Relational Query languages: Relational algebra and relational calculus, Relational algebra operations like select, Project, Join, Division, outer union. Database Design: Introduction to normalization, Normal forms, Functional dependency, Decomposition, Dependency preservation and lossless join, problems with null valued and dangling tuples, multivalued dependencies.	
UNIT- III	Introduction to .NET	8 hrs.
	The .Net Framework and the CLR, CTS, Stack Walk, Garbage Collector, Assemblies, Namespace, Object Oriented Programming by VB. Net: Abstraction, Encapsulation, Inheritance and Polymorphism, Field, Properties, Methods Module, Class. Object Oriented Inheritance: Overloading, Overriding, Shadowing, Application Development Tool: Textbox, Button, Checkbox, Radio Button, List Box, Drop-down lists, Validation Control.	
UNIT- IV	Accessing Database from .NET	8 hrs.
	Datagrid. Data Access with ADO .Net: Connection, Data Adapter, Command, Dataset, Datatable, DataReader.	
UNIT- V	Working on SQL using Oracle	8 hrs.
	Writing Basic SQL Select Statements, Restricting and Sorting Data, Single-Row Functions, Displaying Data from Multiple Tables, Aggregating Data using Group Functions, Sub queries, Manipulating Data, Creating and Managing Tables, Including Constraints, Creating Views	
UNIT- VI	PL/SQL programming using Oracle	8 hrs.
	Overview of PL/SQL Programs, Creating Procedures, Creating Functions, Creating Packages, Oracle Supplied Packages, Creating Database Triggers	

Required Text(s)

1. Henry F. Kourth, Abraham Silverschatz, S. Sudarshan "Database System Concepts", Tata McGraw Hills Fifth Edition.
2. MSDN(VB.NET) and Oracle Manuals (Oracle 9i)
3. Raghu Ramakrishnan , Johannes Gehrke , "Database Management Systems", Tata McGraw Hills Third Edition.

Laboratory Assignment

SQL Assignments
Consider the following schema
EMP(eno,ename,dt-birth,address)
DEPT(dno,dname,floor,budget)
Works-in(eno,dno,pct-time)

An employee can work in any number of departments for a percentage of time determined by pct-time.
On each floor, there can be more than one department. Answer the following queries using SQL.

(i)   Find employees who work in all departments where eno=10 works in.
(ii)   Find employees who work in all departments.
(iii)   Find employees who work in all departments in a floor.
(iv)   Find employees who work for atleast one department on each floor.
(v)   Find employees who do not work all the time.
(vi)   Find departments for which no employee works.
(vii)   Find employees who work only for those departments having maximum budget on that floor
(viii) Find floors having only one department

Q.2 Consider the following schema(refer C.J. date book for clarification or typographical errors)
        Supplier table S(s#, sname, status, city)
        Parts table(p#, pname, color, weight, city)
        Job/Project table J(j#, jname, city)
        Part supplied by supplier for a project with quantity qty SPJ(s#, p#, j#, qty)

(i)   get full details of all the projects
(ii)   get full details for all projects in london
(iii)   get part number for parts with smallest weight
(iv)   get suppliers supplying parts for project J1.
(v)   Get suppliers who supply part P1 for project J1.
(vi)   Get a list of projects where supplier S1 supplies parts
(vii)   Get suppliers who supply red colored parts for project J1.
(viii) Get list of parts supplied to projects executed in london.
(ix)   Get list of local suppliers for a project(supplier and project in same city)
(x)   Get a list of outside suppliers for a project J2.
(xi)   Get list of projects having atleast one outside supplier.
(xii)   Get list of suppliers who supplying parts to project J1 who supply atleast one red colored part to
        other projects.
(xiii) List all suppliers for a project J1.
(xiv)   List all pairs of city values such that a supplier in first city supplies a part in the second city.
(xv)   Get all <city,p#,city) triples such that a supplier in firts city supplies a part in the second city.
(xvi)   Repeat earlier exercise, but do not repeat common city pairs.
(xvii) Get suppliers supplying same part to all projects.
(xviii) Get list of all projects supplied by single supplier.
(xix)   List parts supplied to all projects in london.
(xx)   Get projects using parts only from supplier S1.
(xxi)   Get projects supplied by all suppliers who supply red part.
(xxii) Change the name of project from J6 to video.
(xxiii) Change red color to orange having weight more than 500.
(xxiv) Delete red parts and corresponding SPJ records.
(xxv)   Count the number of projects where supplier S1 supplies parts.
(xxvi) Count the total quantity of part P1 supplied by supplier S1.

Q.3 Consider the following database schema(refer Korth's book for clarification and typographical errors)

Lives(pname, street, city)
Works(pname, cname, salary)
Located-in(cname, city)
Manages(pname, manager-name)

(i)   Find names of all employees for first bank corporation
(ii)   Find the name and city of all employees who work for first bank corporation
(iii)   Find the name, street and city of all employees who work for first bank and earn more than
        10,000/-
(iv)   Find employees who work in same city the company they work for

20

(v)   Find employees living in the same street city their manager lives in.
(vi)   Find all employees who work more than every employee of small bank.
(vii) Find employees who do not work for first bank.
(viii) Assume that the company has branches in several cities. Find all companies having branches in city where first bank has a branch.
(ix)   Find all employees who earn more than average salary of all employees of their company.
(x)   Find company with maximum employees.
(xi)   Find company with smallest payroll.
(xii) Find those companies that pay more, on average , than the average salary of first bank.

.NET Assignments
1. Create a Windows application with two button controls. In the click event for the first button, declare two Integer variables and set their values to any number that you like. Perform any math operation on these variables and display the results in a message box. In the click event for the second button, declare two String variables and set their values to anything that you like. Perform a string concatenation on these variables and display the results in a message box.
2. Create a Windows application with a text box and a button control. In the button's click event, display three message boxes. The first message box should display the length of the string that was entered into the text box. The second message box should display the first half of the string, and the third message box should display the last half of the string.
3. Create a Windows Application with a text box and a Button control. In the Click event of the Button, extract the number from the text box and use a Select Case statement with the numbers 1 through 5. In the Case statement for each number, display the number in a message box. Ensure that you provide code to handle numbers that are not in the range of 1 through 5.
4. Create a Windows Application that contains a ListBox control and a Button control. In the Click event for the button, create a For . . . Next loop that will count from 1 to 10 and display the results in the list box. Then create another For . . . Next loop that will count backwards from 10 to 1 and also display those results in the list box.
5. Create a Windows Application that contains three buttons. Add an enumeration of three names to your code. For the Click event for each button, display a message box containing a member name and value from the enumeration.
6. Create a Windows Application that contains a TextBox control and a Button control. At the form level, create a names array initialized with a single name. In the Click event for the button control, add the code to redimension the array by one element preserving the existing data, add the new name from the text box to the array and to display the last name added to the array in a message box.
   Hint: To determine the upper boundary of the array, use the GetUpperBound(0) method.
7. Create a Windows application with two buttons. Add code to the MouseUp event for the first button to display a MessageBox with a message that the event has fired. Add code to the LostFocus event for the first button to also display a MessageBox with a message that the button has lost focus.
8. Create a Windows application with a toolbar and status bar. At the bottom of the IDE, right-click the ToolStrip control and select the Insert Standard Items menu item to have the standard buttons added to the control. For the Click event for each of the ToolStripButton controls, display a message in the status bar indicating which button was clicked.
9. Create a simple Windows application with a TextBox control and two Button controls. Set the buttons to open a file and to save a file. Use the OpenFileDialog class (not the

21

control) and the SaveFileDialog class to open and save your files. Hint: To use the corresponding classes for the controls use the following statements:

Dim objOpenFileDialog As New OpenFileDialog
Dim objSaveFileDialog As New SaveFileDialog

10. Create a simple Windows application with a Label control and a Button control. Set the button to display the Browse For Folder dialog box with the Make New Folder button displayed. Use My Documents as the root folder at which the dialog starts browsing. Use the FolderBrowserDialog class (not the control) and display the selected folder in the label on your form.

11. Using your Debugging project, add a Try . . . Catch block to the ListCustomer procedure to handle an Exception error. In the Catch block, add code to display a message box with the error message.

12. The Try . . . Catch block that you added in 1 should never throw an error. However, you can throw your own error so that you can test your code in the Catch block. Add a Throw statement as the first line of code in the Try block. Consult the online help for the syntax of the Throw statement.

13. Create a Windows application that will display data to the user from the Authors table in the Pubs database. Use a DataGridView object to display the data. Use the simple select statement here to get the data: Select * From Authors

14. Looking at the DataGridView, it is not very user-friendly. Update the column headings to make more sense. If you know SQL, you can give each column an alias. The current column header names are au_id, au_lname, au_fname, phone, address, city, state, zip, and contract. The solution to this will give each column an alias in SQL.

15. Create a Windows application that uses the OLE DB data provider to read data from the sample Northwind database that gets installed with Microsoft Office. The default location for this database is C:\Program Files\Microsoft Office\Office11\Samples and may vary depending on the version of Microsoft Office that you have and the path that was used during the installation.

Create a SELECT statement that selects the first and last names from the Employee table and use a DataAdapter object to populate a DataTable object with the data. Then read the data from the DataTable object and populate a list box control on your form.

You need to set a reference to the System.Data.dll namespace using the Add Reference dialog box and also import the System.Data and System.Data.OleDb namespaces in your form class.

The results from your application should look similar to the results shown in Figure
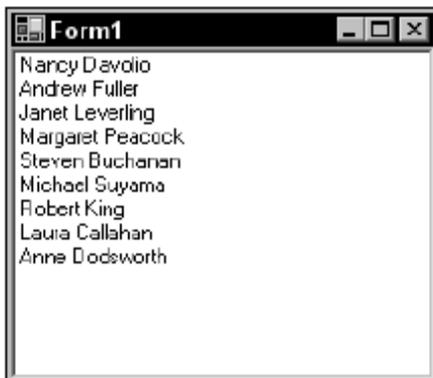
16. Create a Windows application that uses the OleDbDataReader to read data from the Employees table in the Northwind database. Display the results in a list box on a form and design your form so that it looks similar to the form in Exercise 1.

Use the same SQL SELECT statement that you used in the last exercise and follow the steps that you used to set a reference and import the System.Data and System.Data.OleDb namespaces.

17. Create a Windows application containing five Label and five TextBox controls and a BindingNavigator. Bind your data source to the Suppliers table in the Northwind database and select the Company Name, Address, City, Region, and Postal Code fields.

CS – 561 Computer Networks

UNIT- I	Introduction	(06hrs.)
Computer Network, Goals and Applications, Reference models – OSI and TCP/IP. A Comparative study. Network hardware – LAN, MAN and WAN and topologies, LAN components – File server, Workstations, Network Adapter Cards. Connection Oriented and Connection less services, Switching Techniques – Circuit Switching, Packet Switching		

UNIT- II	Data Link Layer	(10hrs.)
Design Issues : Framing, Error Control, Flow Control , Error Detection and Correction, Elementary Data Link Protocols, Sliding window protocol, Data link layer in the Internet – SLIP and PPP.		

UNIT- III	MAC Sublayer	(10hrs.)
Multiple access protocols: Aloha, CSMA Protocols, Collision-Free Protocols, Ethernet : Cabling, Manchester Encoding, MAC Sublayer Protocol, Token bus : MAC Sublayer Protocol, Token Ring : MAC Sublayer Protocol, High speed LANs – Fast Ethernet, FDDI, Wireless LANs, Bridges		

UNIT- IV	Network Layer	(10hrs.)
Design issues, Routing Algorithms: Optimality Principle, Shortest Path Routing, Flooding, Distance Vector Routing, Link State Routing , Hierarchical Routing, Broadcasting Routing, Multicast Routing, The Network Layer in the Internet: Internet Protocol, Internet addressing and Internet Control protocols.		

UNIT- V	Transport Layer	(04hrs.)
Services, The Internet Transport Protocols : TCP and UDP.		

UNIT- VI	Application layer	(5hrs.)
DNS Name Space, Name Servers, FTP, TELNET, WWW, SNMP, HTTP, SMTP , Network Security : Cryptograpy, Symmetric- key Algorithms, Public- key Algorithms, Digital Signatures, E-mail Security.		

Required Text(s)

1. Tanenbaum, Computer Networks , Addison-Wesley, 4th Ed., 1996
2. Data and Computer Communications : W.Stallings, , Prentice-Hall, 5th Ed., 1997.
3. Computer Networking : James F. Kurore & Keith W. Rose , Pearson Education, Third Edition, 2005.
4. Communication Networks : Fundamentals Concepts and Key Architecture : Alberto Leon-Garcia and Indra Widjaja, , Tata McGraw-Hill Publishing Company Limited, ISBN 0-07-0402235-3.
5. Data and Network Communication : Michael A. Miller, Delmar Thomson Learning inc. ISBN 0-07668-1100-X.

IC - 491 Organization and Management Concepts

UNIT- I	Managing Self	10 hrs.
Managing Body, Managing Mind, Managing emotion, Managing habits and attitudes, Managing consciousness		

UNIT- II	Essential Managerial Skills	10 hrs.
Decision Making: Decision making under certainty, uncertainty and risk, Identify real problem, Making Effective decisions, Decision support system Problem Solving: Importance, Steps in solving structured problem, Creativity : Creativity Process, Factors inhibiting creativity Managing Time: The Time Management Matrix, Time Wasters, Managing activities		

UNIT- III	Understanding Management	10 hrs.
Importance of management concepts for technical persons: Requirement of career growth, working in an organization Understanding Organization.: Definition of organization, goals, structure and processes Function of business : Purpose of business, Basic function of enterprise Understanding Management.: Definition of management nature and purpose, Management science and art		

UNIT- IV	Process of management	16 hrs.
Planning : Steps in planning, Types of plans, How to set objectives Organizing : Basis of division of work , Levels and span of management, Departmentalization - Functional Organization, Line and staff organization, Matrix Organization Staffing : Matching individual with job, Selection and placement Directing : Activation, Motivation, Leading, Controlling : Process of control, Requirements of effective controls, Role of IT in control		

UNIT- V	Management by Indian Values	4 hrs.
Understanding Indian psychology, Individual Respect, Trust, Chitta Shuddhi, Self discipline, Inspiration to give, Renunciation and detachment		

UNIT- VI	Ethics In Management	2 hrs.
Institutionalizing Ethics, Factors that raise Ethical Standards		

Required Text(s)

1. Harold Koontz O' Donnel and Weihrich " Essentials of management "
   Tata McGraw Hill 1998
2. R.D Agrawal
   " Organization and Management " Tata McGraw Hill 1995