PROPOSED CURRICULAR STRUCTURE FOR PART – 2 (2ND YEAR) OF THE FULL- TIME DIPLOMA COURSE IN ENGINEERING AND TECHNOLOGY

WEST BENGAL STATE COUNCIL OF TECHNICAL EDUCATION

TEACHING AND EXAMINATION SCHEME FOR DIPLOMA IN ENGINEERING COURSES

COURSE NAME: COMPUTER SOFTWARE TECHNOLOGY

SEMESTER: FOURTH

BRANCH CODE: CSWT

SR. NO.	SUBJECT	CREDITS	PERIODS			EVALUATION SCHEME					
			L	T U	PR	INTERNAL SCHEME			ESE	PR	TOTAL MARK
						TA	CT	Total			
1	Microprocessor &Microcontroller	3+1	3		2	10	20	30	70	50	150
2	Data Structure	3+2	3		3	10	20	30	70	100	200
3	Computer Organization & Architecture	3	3			10	20	30	70		100
4	Operating System	3+2	3		3	10	20	30	70	100	200
5	Communication Technique	3+1	3		2	10	20	30	70	50	150
6	Development of Life Skills-II	1+1	1		2					50	50
7	Professional Practice-II (Visual Basic)	2			3					50	50
Total		25	16		15	50	100	150	350	400	900

STUDENT CONTACT HOURS PER WEEK: 31 HRS.

Theory and Practical Periods of 60 minutes each.

L-Lecture, TU-Tutorials, PR-Practical, TA-Teachers Assessment, CT-Class Test, ESE-End Semester Examination.

MICROPROCESSOR AND MICRO-CONTROLLER

Subject Code	Course offered in	Course Duration	3 lecture contact periods	Full Marks
CSWT / 4 / T4 / MIC	Part – II 2nd Semester	17 weeks	per week	70

OBJECTIVE

This subject will enable the students to comprehend the concepts and working principle of electronics devices and circuits and their application in electronic system. The knowledge acquired by student will help them to troubleshoot and repair electronic circuits and devices.

MODULAR DIVISION OF THE SYLLABUS

GROUP	MODULE	*TOPIC*	*PERIODS*
A	1	INTRODUCTION TO MICROPROCESSORS	2
	2	ARCHITECTURE OF MICROPROCESSORS	13
B	3	PROGRAMMING OF MICROPROCESSORS	14
C	4	INTERFACING OF MEMORY AND I/O PORTS	14
	5	SINGLE CHIP MICROCONTROLLER	6

GROUP	MODULE	OBJECTIVE QUESTIONS				SUBJECTIVE QUESTIONS			
		TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS	TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS
A	1, 2	30	ANY 20	ONE	20	3	FIVE, TAKING AT LEAST ONE FROM EACH GROUP	TEN	10 X 5 = 50
B	3					2			
C	4,5					3			

CONTACT PERIODS: 49 **INTERNAL ASSESSMENT: 2** **TOTAL PERIODS: 51**

	Content (Name of Topic)	Hrs/ Credit	MARKS
	Group A		
Module 1	**INTRODUCTION TO MICROPROCESSORS**		
	Evolution of microprocessors; Specific features of microprocessors; Application of microprocessors.	**2**	
Module 2	**ARCHITECTURE OF MICROPROCESSORS**		
	Explanation of each Functional Block Diagram and Internal Architecture of 8086/8088 – ALU, Registers, Control unit, Clocks, Bus Structure; Address, Data and Control Bus of 8085, 8086/8088; pin Details of 8085, 8086/8088, Introduction to PC range of Microprocessors.	**16**	
	GROUP-B		
Module 3	**PROGRAMMING OF MICROPROCESSORS**		
	Different Addressing modes of 8085,8086/8088; Instruction Cycle of 8085,8086/8088 (including subroutine calls, jumping, comparing, string instructions of 8086); Timing Diagram of different parts of Instruction Cycles; Solving basic problems of Assembly Language Programming using 8085 Trainer Kit and Using any 8086 Assembler or DOS Debug Program.	**18**	
	GROUP-C		
Module 4	**INTERFACING OF MEMORY AND I/O PORTS**		
	Address Space; Memory mapped I/O, I/O mapped I/O; address Decoding and Interfacing of Memory; DMA Description with sequence of steps and control flow, Structure of a generic DMA controller; programmer's model of 8251, Programmer's model of 8255 with its Interfacing; Outputting data to Parallel Port using 8086 Commands in DOS/WIN9x; Interrupts – Hardware and Software interrupts, A brief overview of BIOS Interrupts, An introduction to (i) Disk Access Interrupts (ii) CRT/Graphics Interrupts.	**18**	
Module 5	**SINGLE CHIP MICROCONTROLLER**		
	Programming model of 8051: CPU – Address bus – Data bus – Control bus – Register – Internal RAM and ROM – Ports (serial and parallel) – Timers – Interrupts. ADDRESS MODES: Immediate – Register – Direct – Indirect – Indexed. INSTRUCTION TYPES: Arithmetic – Logical – Data Transfer (Internal/External) – Boolean. **Control Transfer and Special Function Register**	**6**	

REFERENCE BOOKS

1. Microprocessor Architecture, Programming and Applications – Ramesh S Goonkar.
2. Microprocessors and Interfacing – Douglas V Hall
3. Fundamentals of Microprocessors and Microcomputers – B Ram.
4. Advances Microprocessors and interfacing – B Ram.

MICROPROCESSOR AND MICRO-CONTROLLER LAB

DETAIL COURSE CONTENT

Subject Code	Course offered in	Course Duration	Full Marks
CSWT / 4 / PR / MPL	Part – II Second Semester	17 weeks	50

CONTACT PERIODS

2 sessional contact periods per week for 17 weeks

OBJECTIVE

On satisfactory completion of the course, the students should be in a position to develop the skills corresponding to the knowledge acquired in the theoretical subject MICROPROCESSOR & MICROCONTROLLERS.

EXAMINATION SCHEME

1. **Continuous Internal Assessment of 25 marks** is to be carried out by the teachers throughout Part – II Second Semester. **Distribution of marks:** Performance of Job – 15, Notebook – 10.
2. **External Assessment of 25 marks** shall be held at the end of Part – II Second Semester on the entire syllabus. One job per student from any one of the jobs done is to be performed. Job is to be set by lottery system. **Distribution of marks:** On spot job – 15, Viva-voce – 10.

1. To be familiar with 8085-system development kit.
2. To write, test and debug (if necessary) assembly and machine language programs using instruction set of 8085. A list of sample problems is given below.
3. To practice on EPROM programming using SDK8085.
4. To write programs to execute the following**:** —
   (a) display digits through seven-segment display using 8255.
   (b) rolling display-using 8255.
   (c) display hexadecimal digits using 8279.
   (d) development of a counter by 8255 and 8253.
   (e) developments of waveforms using 8255 and 8253.
   (f) receive on-line data through ADC and display.
   (g) develop interfacing program using DAC.
5. To develop program to serve the interrupts of 8055 using SDK.
6. To develop a keyboard interface using 8255.
7. To practice assembly language programming with 8086 / 8088 using PC only.
8. To practice programming with 8051SDK.

LIST OF SAMPLE PROBLEMS FOR MICROPROCESSOR LAB - I

1. Write a program which loads Reg. A, B, C and D with the same constant (example AA), optimise the program in such a way that the least number of program bytes are used. Test the program in single-step mode. After each step test the register of interest.

2. Assume that six types of data are stored at consecutive memory locations, starting at location X. Write a program which loads register E with (X), that is with data contained in memory location X,D with (X+1),c with (X+2) and A(X+3). [A] Use direct addressing; [b] Use indirect addressing.
3. Assume that one byte of a data is stored at memory location X. Write a program which tests bit five of (X), write 00 into (X+1), if bit five is 0, and write FF at the same location if bit five is 1. Test the program in single step and run mode.
4. Write a program which tests the zero condition of a data-byte specified at memory location X. If it is 0, 00 should be stored at location (X+1), and if non-zero FF should be stored at the same location.
5. Write a program which tests the all-one condition of a data- byte specified at memory location X. If all bytes are one, store FF at (X+1), else store 00 at the same location.
6. Four bytes of data are specified at consecutive memory starting at X. Write a program which increments the value of all four bytes by two.
7. Two data bytes are stored at location X and Y. Interchange the data at the two locations using indirect addressing.
8. Two binary numbers are stored at data-memory locations X and (X+1). Add the two numbers and store the result at (X+2)
9. Four unsigned binary numbers are stored at consecutive memory location starting at X. Compute the sum of the four numbers ignoring the possible overflow and store it at location Y. Use indirect addressing.
10. Two unsigned binary numbers are stored at consecutive memory location X and (X+1). Compute the difference (X+1) - (X) and store the result at and the sign (00 if positive, 01 if negative) at (Y+1).
11. A double precision number (that is a sixteen bit unsigned number) is stored at X and (X+1), low order byte at X. Another double precision number is stored at Y and (Y+1).Subtract the two numbers and store the result in W and (W+1)
12. Two 2-digit BCD numbers are stored in consecutive memory locations X and (X+1). Write a program for computing the sum and store the result at location Y (use decimal adjust).
13. Two 2-digit BCD numbers are stored at consecutive memory locations X and (X+1). Compute the difference of the two and store the result at location Y. Use decimal subtraction with the aid of DAA and 10's compliment arithmetic.
14. Implement a time-delay loop (counter) for the generation of milliseconds delay. Determine the exact time- delay by adding up the states of the instructions executed by the program.
15. Write a program for a decimal counter which counts from 00 to a given decimal number with a programmable clock frequency and display the count in the data-field using the corresponding monitor subroutine. The frequency is specified at data memory location X.
16. N binary numbers are stored at consecutive data memory locations starting at X and N is defined at memory location Y. Find the largest number and display it in the data field.
17. N binary numbers are stored at consecutive data memory locations starting at X and N is defined at memory location Y. Rearrange the numbers in ascending and descending order.
18. Write a program for moving a data block, starting from address X to address Y. The addresses X, Y as well as the length of the block are specified at some suitable memory locations.
19. Write a program for displaying (in address field) the hex-character which is depressed. The program should be such that the data can be entered through the keyboard indefinite number of times and at every key depression the display characters get shifted 1 digit to left as the new digit is entered an the least significant digit. A small part of the monitor-program is thus implemented. For this program, use the keyboard's subroutine.
20. Write the following program:
    If key GO command is depressed (Byte 12H will be entered into the accumulator) the subsequent hexadecimal key entries will be displayed in the address field and if key single step command is pressed (15H), subsequent hexadecimal key entries will be displayed in the data field. This problem demonstrates a part of a possible monitor function.
21. Two unsigned binary numbers are stored at data memory locations X and (X+1). Find the product and display in the address field. Find the product by successive addition that is the multiplier is added as often to itself and as corresponds to the value of multiplicant.

22. Divide a sixteen bit number by an eight bit number and display the result in the data field.
    [A] Use successive subtraction method.

    [B] Use the common for division (shift right and subtract)

    [C] Compare [A] with [B].

23. A 2-digit BCD number is stored in memory location X. Convert the number into binary and display the result in the data field.

24. Divide the contents of a memory location X into two 4-bit sections and store them in the memory locations (X+1) and (X+2). Place the 4 most significant bits of memory locations X in the 4 least significant bit positions of memory location (X+1), place the 4 least significant bit positions of memory location (X+2). Clear the 4 most significant bit positions of memory locations (X+2) and (X+2).

25. Add the 16-bit numbers in memory locations X and (X+1) in the memory location (X+2) and (X+2). The most significant 8-bits of the two numbers to be added are in memory locations (X+1) and (X+3). Store the result in memory locations (X+4) and (X+5) with most significant byte in memory location (X+5).

26. Place the larger of the contents of the memory locations X and (X+1) in the memory location (X+2). Assume that the contents of the memory location X contain a number between 0 and 7 inclusive.

27. Calculate the square of the contents of the memory locations X using a table and place the result in the memory location (X+1). Assume memory locations X contain a number between 0 and 7 c.

28. Place the 1's complement of a 16-bit number in memory locations X and (X+1) in memory locations (X+2) and (X+3). The most significant bytes in locations (X+1) and (X+3).

29. Add the 24-bit number in memory locations X,(X+1) and (X+2) to the 24-bit number in memory locations (X+3), (X+4) and (X+5), the least significant 8-bits in memory locations X and (X+3). Store the result in memory locations (X+6) and (X+7) and (X+8) with the most significant bits in (X+8) and the least significant bit in (X+6).

30. Calculate the squares of contents of memory locations X and (X+1) and add them together. Place the result in memory location (X+2). Assume that the memory locations X and (X+1) both contain a number between 0and 7 inclusive.

31. Calculate the sum of a series of numbers. The length or the series is in memory location (X+2) and the series itself begins in memory location (X+2).Store the sum in memory location X and (X+1) eight most significant bits in (X+1).

32. Determine the number of negative elements in a block of data. The length of block is in memory location (X+1) and the block itself starts in memory location (X+2). Store the number of negative elements in the memory location X.

33. Find the largest element in a block of data. The length of the block is in memory location (X+1) and block itself starts in memory location (X+2). Store the maximum in memory location X. Assume that the numbers in the block are all 8-bit unsigned binary numbers.

34. Shift the contents of memory location X left until the most significant bit of the number is 1. Store the result in memory location (X+1) and the numbers of the left shifts required in the memory location (X+2). If the contents of the memory location X are 0, clear both (X+1) and (X+2).

35. Determine the numbers of zeroes, positive (most significant zero but entire number not zero) and negative elements (most significant 1) in block. The length of the block is in memory location (X+3) and the block itself starts in memory location (X+4). Place the number of negative elements in the memory location X, the number of zero elements in memory location (X+1) and the number of positive elements in memory location (X+2).

36. Find the smallest element in a block of data. The length of the block is in memory location (x + 1) and the block itself begins in memory location (x+2). Store the minimum in memory location X. Assume the numbers in the block to be 8-bit unsigned binary numbers.

37. Convert the contents of memory location X to a 7-segment code in memory location (X+1). If the memory location X dopes not contain a single decimal digital, clear memory location (x+1).

38. Convert the contents of memory location X from an ASCII character to a decimal digit and store the result in memory location (X+1). If the contents of memory location X is not the ASCII presentation of the decimal digit, set the content of the memory location (x+1) to FF (hex).

39. Convert two BCD digits in memory location X and (X+1) to a binary number in memory location(x+2). The most significant BCD digit is the one in memory location X.

40. Convert a string of 8 ASCII characters into a binary number and store the result in memory location X. If any of the characters are not either ASCII zero or ASCII one, set memory location(x+1) to FF (hex), otherwise, clear memory location(X+1). The string of characters is in memory location (X+2) through(X+9) with the most significant bit in memory location (x+2).
41. Convert the contents of memory location X to a hexadecimal digit and store the result in memory location (X+1). Assume that the memory location X contains the ASCII representation of hexadecimal digit.
42. Add even parity to a string of 7-bit ASCII characters. The length of the string is in memory location X, and the string starts from (X+1) onwards. Place even parity in the most significant bit of each character, i.e. set MSB if that makes the total number of 1 bit in the word even.
43. Add two multiple word binary numbers. The length of the numbers (in byte) is in memory location Z, the starting addresses of the numbers are in registers DE and HL, and the starting address of the result is in register BC. All the numbers begin with least significant bits.
44. Write a program segment for 8085 that could be used to generate a delay (a) 100s. (b) 10 ms. This means that in a program, before execution of a specific instruction, if a delay of at least 100s is required, one should be able to obtain it by placing this segment just before that instruction.
45. Write a program which will subtract the number in address XX01 by adding the complement of the number to be subtracted. The result should be stored in address in XX02.
46. Write a program which will put two single digit hexadecimal numbers together. The first single digit hexadecimal or hex would be the MSB and the second would be LSB.
47. Write a program which will test bit-3 of hex number with the location of bet-3 in XX00. If bit-3 is high ("1"), no action is taken and the data address location will be in null state.
48. Write a program which will determine if the parity of the number in memory location XX00 is odd or even. If the number is odd a 00 will be stored in XX01 and if the number is even an EE will be stored in XX01.

DATA STRUCTURE

Subject Code	Course offered in	Course Duration	3 lecture contact periods	Full Marks
CSWT / 4 / T / DS	Part – II Second Semester	17 weeks	per week	70

OBJECTIVE

The study of data structure is an essential part of computer science. This subject develops the concept of storage structure of data and helps to write efficient application program.

MODULAR DIVISION OF THE SYLLABUS

GROUP	MODULE	*TOPIC*	*CONTACT PERIODS*
A	1	PROBLEM SOLVING AND SOME CONCEPTS	06
	2	PRIMITIVE & NON PRIMITIVE LINEAR DATA STRUCTURE	20
	3	RECURSION	03
B	4	NON-PRIMITIVE NON-LINEAR DATA STRUCTURES	10
	5	SORTING AND SEARCHING	10

CONTACT PERIODS: 49 **INTERNAL ASSESSMENT: 2** **TOTAL PERIODS: 51**

EXAMINATION SCHEME

GROUP	MODULE	OBJECTIVE QUESTIONS				SUBJECTIVE QUESTIONS			
		TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS	TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS
A	1, 2, 3	15	ANY 20	ONE	1 x 20 = 20	FIVE	FIVE, TAKING AT LEAST TWO FROM EACH GROUP	TEN	10 X 5 = 50
B	4, 5	10				FOUR			

DETAIL COURSE CONTENT

GROUP – A 29 PERIODS

Module 1 PROBLEM SOLVING & SOME CONCEPTS 6

Algorithms and flow charts, concepts of algorithmic complexity (big O notation, small o notation), concepts of structured programming: top-down design. Control structures, concepts of program modules functions and subroutines. (Algorithms are to be described in C like pseudo language)

Module 2 PRIMITIVE & NON-PRIMITIVE LINEAR DATA STRUCTURE 20

2.1 PRIMITIVE DATA STRUCTURES**:** Integer (signed, unsigned, long, short) — Real (float, double, long double) — Character and Boolean data types — their declaration & space usage in computer memory.

2.2 NON-PRIMITIVE DATA STRUCTURES**:**

2.2.1 ARRAY**:** Definition — Declaration initialisation and usage of one and two-dimensional arrays —Numeric and character type arrays — Arrays as parameters — Matrix operations: Addition, subtraction, multiplication, transpose

2.2.2 STRING**:** Definition — Declaration — String operations: String comparison, length of a string, concatenation of two strings, copy of a string, extract a portion of a string, reversing of a string

2.2.3 STACK**:** Definition — Declaration — Operation — Stack implementation using array — Expression evaluation by stack (infix, prefix and postfix)

2.2.4 QUEUE**:** Definition — Declaration — Operation — Priority queue (definition and example)

2.2.5 LINKED LISTS**:** Concepts and representation of linked lists in memory — Array implementation of lists and its limitation — Operation and analysis of singly, doubly and circular linked lists, their comparison and applications (e.g., polynomial arithmetic)

Module 3 RECURSION 3

Basic concepts and examples of recursion e.g. factorial problem, Fibonacci sequence etc. — Direct and indirect recursion and their overhead.

GROUP - B 20 PERIODS

Module 4 Non-Primitive Non-Linear Data Structures 10

4.1 Concepts of non-linear data structures and their examples — Compare it with linear data structure

4.2 TREE**:** Definition and application of tree — Binary tree: Definition and it's Implementation, expression processing by binary tree — Tree traversal (pre-order, post-order and in-order) — Spanning tree concept and its application — Balancing of a tree — AVL tree its definition, construction and rotation — B-tree its definition and use

Module 5 Sorting & Searching 10

5.1 Definition of internal and external Sorting and Searching and their examples

5.2 SORTING**:** Algorithms and their analysis (time and space) — Bubble sort — Insertion sort — Merge sort — Quick sort — Radix sort — Heap sort

5.3 SEARCHING**:** Linear search — Binary search — Hashing — Hash functions, their collisions & resolutions

REFERENCE BOOKS

1. Data Structures using C & C++ / Y. Langsam, M. J. Augenstein and A. M. Tenenbaum / PHI
2. Mastering Algorithms with C / Kyle Loudon, O'Reilly / SPD
3. Introduction to Data Structures with application / Tremblay / Tata McGraw-Hill

DATA STRUCTURE LAB

Subject Code	Course offered in	Course Duration	3 sessional contact periods	Full Marks
CSWT / 4 / PR / DSL	Part – II Second Semester	17 weeks	per week	100

OBJECTIVE

Data Structures are commonly used in many program designs. Here following programs are based on the C language and the following exercises include developing algorithms, writing code, entering the program, compiling and debugging the program, giving test data and executing the program.

MODULAR DIVISION OF THE SYLLABUS

MODULE	*TOPIC*	*PERIODS*
1	ARRAY, STRING, STACK, QUEUE & POINTER RELATED PROBLEMS	15
2	STRUCTURE , UNION , LINKED LIST RELATED PROBLEMS	11
3	RECURSION , SORTING , SEARCHING ,TREE & FILE	25

TOTAL PERIODS: 51

EXAMINATION SCHEME

1. **Continuous Internal Assessment of 50 marks** is to be carried out by the teachers throughout Part – II Second Semester. **Distribution of marks:** Performance of Job – 35, Notebook – 15.
2. **External Assessment of 50 marks** shall be held at the end of Part – II Second Semester on the entire syllabus. One job per student from any one of the jobs done is to be performed. Job is to be set by lottery system. **Distribution of marks:** On spot job – 25, Viva-voce – 25.

DETAIL COURSE CONTENT

Module 1 ARRAY, STRING, STACK, QUEUE & POINTER RELATED PROBLEMS 15

1.1 One and two dimension ARRAY related problems.
1.2 Different STRING operations using different C library functions
1.3 Creation of STACK and its related problems such as expression conversion and evaluation.
1.4 QUEUE, its creation and related problems.
1.5 POINTER related problems.

Module 2 STRUCTURE , UNION , LINKED LIST RELATED PROBLEMS 11

2.1 To create a heterogeneous data structure (using STRUCTURE) and then perform related problems.
2.2 Singly, Doubly and Circular LIST related problems.

Module 3 RECURSION , SORTING , SEARCHING ,TREE & FILE 25

3.1 To write a RECURSIVE function and change it to non-recursive way.
3.2 To write the following different SORTING programs in C**:** —
(a) Bubble sort, (b) Insertion sort, (c) Merge sort, (d) Quick sort, (e) Radix sort, and, (f) Heap sort.
3.3 To construct a binary TREE and traverse its different nodes.
3.4 Binary SEARCH related problems.
3.5 FILE related problems.

REFERENCE BOOKS

1. Classic Data Structures / D. Samanta / PHI
2. Data Structures using C and C++ / Tanenbaum / PHI
3. Programming with C / R. K. Venugopal & Prasad / Tata McGraw-Hill

LIST OF SAMPLE PROBLEMS FOR DATA STRUCTURE LAB

1. To write a program to check whether a word is palindrome or not.
2. To create a two dimensional array of numbers and calculate & display the row & column sum and the grand total.
3. To write a program of matrix multiplication.
4. To write a program to insert (Push) an element into the sack and delete (Pop) an element from the stack using pointer.
5. To write a program to convert an infix expression to a postfix expression.
6. To evaluate a postfix expression.
7. To write a program to insert an element in the queue and delete an element from the queue using pointer.
8. To create a circular queue and add an element and delete an element from a circular queue.
9. To write a program of a structure containing an item name along with the unit price. The user enters the item name and quantity to be purchased. Program print outs total price of item with name using pointer in a structure or array in a structure.
10. To create a single linked list and — (a) insert a node in the list (before header node, in between two nodes, end of the list); (b0 delete a node from the list (1^{st} node, last node, in between two nodes); (c) Concatenate two lists.
11. To create a doubly linked list and — (a) insert a node in the list (before header node, in between two nodes, end of the list); (b0 delete a node from the list (1^{st} node, last node, in between two nodes); (c) Concatenate two lists.
12. To create a circular linked list and insert & delete an element from the list.
13. To write a program to calculate the binomial co-efficient of ${}_nC^r$ of two numbers using recursive function. Also write the same program using function in non-recursive way.
14. To write a program to generate Fibonacci Series using recursive function. Also write the same program using function in non-recursive way.
15. To write a program to search an Item from a list of numbers using—(i)Linear search, (ii)Binary search
16. To write a program to sort a list of numbers using — (i) Heap Sort, (b) Quick Sort, (c) Bubble Sort.
17. To write a program to sort a list of numbers using — (i) Insertion Sort, (b) Merge Sort, (c) Radix Sort.
18. To write a program to create a binary tree and traverse it in pre-order and post-order form.
19. To write a program to create a binary search tree and — (a) insert a new node in the BST, (b) search a node in the BST, (c) delete a node from the BST.
20. To write a program to create a file, read the file, update the file, insert into the file, and, delete from the file. (The file contains, say for example, student first name, middle name, surname, address, phone no., roll no., branch etc.)

Computer Organization & Architecture

Subject Code	Course offered in	Course Duration	3 lecture contact periods	Full Marks
CSWT / 4 / T/COA	Part – II 2nd Semester	17 weeks	per week	70

OBJECTIVE

This subject will help to acquire knowledge of fundamentals of computer system and its organization. It will enable the students to comprehend the organization and working principle of various units of computer system for storing and processing information.

MODULAR DIVISION OF THE SYLLABUS

GROUP	MODULE	*TOPIC*	*CONTACT PERIODS*
A	1	EVOLUTION OF COMPUTER AND BASIC STRUCTURE	06
	2	NUMBER & CHARACTER REPRESENTATION	10
	3	INSTRUCTION, ADDRESSING MODES & REGISTER	10
B	4	MEMORY	14
	5	I / O DEVICES	09

CONTACT PERIODS: 49 **INTERNAL ASSESSMENT: 2** **TOTAL PERIODS: 51**

EXAMINATION SCHEME

GROUP	MODULE	OBJECTIVE QUESTIONS				SUBJECTIVE QUESTIONS			
		TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS	TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS
A	1, 2, 3	15	ANY 20	ONE	1 x 20 = 20	FIVE	FIVE, TAKING AT LEAST TWO FROM EACH GROUP	TEN	10 X 5 = 50
B	4, 5	10				FOUR			

DETAIL COURSE CONTENT

GROUP – A 26 PERIODS

Module 1 EVOLUTION OF COMPUTER AND BASIC STRUCTURE 6

Brief history of devel opment of Babbage's machi ne — Stored program concept — Von Neumann architecture (definition only) — Generations of computers — SSI, LSI & VLSI classification — Micro, Mini, Mainframe & Supercomputers — PC, PC/XT, PC/AT — Functional units, Bus structures

Module 2 NUMBER & CHARACTER REPRESENTATION 10

Bits, Bytes, Fixed & Floating point numbers — Biased representation — Arithmetic operation — B ooth's algorithm — NUMBER REPRESENTATION: Sign magnitude, 1's complement, 2's c omplement forms, Common errors, Round off errors — CODES: BCD, Excess-3, Gray, ASCII, EBCDIC.

Module 3 INSTRUCTION, ADDRESSING MODES & REGISTER 10

Instruction format — Different types of instructions — ADDRESSING MODES**:** Implied, Immediate, Direct, Register, Register Indirect, Indirect, Indexed, Paged — Di fferent CPU registers**:** Programmer accessible & non-accessible, Operational concept of computer

GROUP – B 23 PERIODS

Module 4 MEMORY 14

Concept of words — Memory size — TYPES OF MEMORY: Input processor memory, Main memory, Secondary memory, Cache memory, Virtual memory — MEMORY ORGANIZATION: SRAM, DRAM, ROM, PROM, EPROM, EEPROM etc., Floppy & Hard Disk (Sectors, Tracks, & Cylinders, Accessing mechanisms, Storage capacity), Magnetic tapes, CD-ROM — Memory hierarchy considering size, speed, cost

Module 5 I/O DEVICES 9

Printers**:** Dot Matrix, Inkjet (Line, Laser) — Visual Display Unit — Keyboard — Mouse — Joystick — Scanners — Digitisers

REFERENCE BOOKS

1. Computer Organization / Hamacher, Vranesic, Zaky / M.H
2. Computer Organization and Architecture / William Stallings / Prentice Hall of India, N. Delhi
3. Computer Fundamentals – Architecture and Organization / B Ram / Tata McGraw-Hill
4. Microprocessor / Ajit Pal / Tata McGraw-Hill
5. Computer System Architecture / V. K. Jain / S.K. Kataria & Sons

OPERATING SYSTEM

Subject Code	Course offered in	Course Duration	3 lecture contact periods	Full Marks
CSWT / 4 / T/OS	Part – II Second Semester	17 weeks	per week	70

OBJECTIVE

This subject intends to teach the student about the various functions of an operating system and how it is organized in various layers to perform different functions. These basic concepts can be used for a proper understanding of single-user and multi-user operating systems.

MODULAR DIVISION OF THE SYLLABUS

GROUP	MODULE	TOPIC	CONTACT PERIODS
A	1	INTRODUCTION	8
	2	PROCESS MANAGEMENT	14
	3	DEADLOCK	7
B	4	MEMORY MANAGEMENT	10
	5	FILE MANAGEMENT	6
	6	DISK MANAGEMENT	4

TOTAL PERIODS: 51

CONTACT PERIODS: 49 **INTERNAL ASSESSMENT: 2**

EXAMINATION SCHEME

GROUP	MODULE	OBJECTIVE QUESTIONS				SUBJECTIVE QUESTIONS			
		TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS	TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS
A	1, 2,3	15	ANY 20	ONE	1 x 20 = 20	SIX	FIVE, TAKING AT LEAST TWO FROM EACH GROUP	TEN	10 X 5 = 50
B	4,5	10				FOUR			

DETAIL COURSE CONTENT

Name of Course: DISCRETE MATHEMATICS	
Subject Code : CSWT / 4 / T / OS	Semester: FOURTH
Duration : 17 weeks	Maximum Marks : 100
Teaching Scheme :	Examination Scheme :
Theory: 3 contact hours/week.	Internal Examination : 20 Marks
Tutorial :-	Class Attendance : 5 Marks
Practical : 3 contact hours/week.	End Semester Examination : 70 Marks
Credit : 3	Teacher's Assessment : 5 Marks

	Content (Name of Topic)	Hrs/ Credit	MARKS
	Group A		
Module 1	**INTRODUCTION**		
	1.1 An Introduction to Operating System & its Services 1.2 Various Types of Operating Systems 1.3 Operating System Structure 1.4 Concepts of**:** Process . Files . System Calls . Interrupt . Shell	**8**	

Module 2	**PROCESS MANAGEMENT**		
	2.1 An Introduction to process; Process State & Transition 2.2 Process Cont rol Block, Process Context, Context Switch 2.3 Thread (overview, benefits of threads, user and kernel threads.) 2.4 Process Scheduling (Pre-emptive & Non-Pre-emptive Algorithms) (a) FCFS (Fi rst Come First Serve) Algo; (b) Shortest Job First; (c) Priority Scheduling; (d) Round Robin Scheduling. 2.5 Performance Criteria of Scheduling Algorithm (a) CPU Utilization; (b) Throughput; (c) Turnaround Time; (d) Waiting Time; (e) Response Time. 2.6 Overview of: Inter-process Communication, Process Synchronization , Race Condition ,Critical Section . Semaphore	**14**	
Module 3	**DEADLOCK**		
	3.1 Introduction to Deadlock 3.2 Necessary Condition for Deadlock 3.3 Method for Handling Deadlock (a) Brief Overview of Deadlock Prevention; (b) Deadlock Avoidance (Banker's Algorithm); (c) Deadlock Detection & Recovery.	**7**	
Group B			

Module 4	**MEMORY MANAGEMENT**		
	4.1 Partitioned Memory Management (Static & Dynamic) 4.2 Concept of Fragmentation & Compaction 4.3 Paging, segmentation, segmentation with paging. Virtual Memory, Demand Paging 4.4 Page Replacement Algorithms (FIFO, Optimal, LRU Algorithms)	**10**	
Module 5	**FILE MANAGEMENT**		
	5.1 File Concepts . Types of Files . File Attributes . File Operations 5.2 Access Methods**:** Sequential access . Random access 5.3 Hierarchical Directory System	**6**	
Module 6	**DISK MANAGEMENT**		
	6.1 Disk structure, disk reliability, disk formatting, boot block, bad blocks 6.2. Disk scheduling (FCFS, SSTF, SCAN,C-SCAN)	**4**	

REFERENCE BOOKS

1. Operating Systems, Galvin, John Wiley

2.Operating Systems , Milankovic, TMH

3. Operating System Design & Implementation / Andres's Tanenbaum / Prentice Hall of Indi a, N. Delhi

4. Operating Systems / Stuart E Mandnick & John J Donovan / McGraw-Hill

5.Operating Systems,Prasad,Scitech

OPERATING SYSTEM LAB

Subject Code	Course offered in	Course Duration	3 sessional contact periods	Full Marks
CSWT / 3 / PR/OSL	Part – II Fourth Semester	17 weeks	per week	100

OBJECTIVE

On satisfactory completion of the course, the students should be in a position to develop the skills corresponding to the knowledge acquired in the theoretical subject OPERATING SYSTEMS..

MODULAR DIVISION OF THE SYLLABUS

MODULE	TOPIC	CONTACT PERIODS
1	WINDOWS NT / 2000 BASICS	10
2	OVERVIEW OF UNIX OS	5
3	BASIC UNIX COMMANDS	8
4	BASIC SYSTEM ADMINISTRATION	10
5	SHELL PROGRAMMING	10
6	Process	8

TOTAL PERIODS: 51

EXAMINATION SCHEME

1. **Continuous Internal Assessment of 50 marks** is to be carried out by the teachers throughout Part . II Second Semester. **Distribution of marks:** Performance of Job . 35, Notebook . 15.

2. **External Assessment of 50 marks** shall be held at the end of Part . II Second Semester on the entire syllabus. One job per student from any one of the jobs done is to be performed. Job is to be set by lottery system. **Distribution of marks:** On spot job . 25, Viva-voce . 25.

DETAIL COURSE CONTENT

	Content (Name of Topic)	Hrs/ Credit	MARKS
Module 1	**WINDOWS NT / 2000 BASICS**		
	i) NT / 2000 Features, NT / 2000 File System ii) Creating a New User , Adding an Account to a Group, Creating permission for a system resources, iii) NT / 2000 Server Installation & Configuration. iv) Using Task Manager, Compressing & Uncompressing Disk. Using Event Viewer: (a) application log, (b) security log. Using Performance Monitor,	**10**	
Module 2	**OVERVIEW OF UNIX OS**		
	UNIX as an operating system , Kernel, Shell , User, UNIX File System . Files & Directory, File System Hierarchy.	**5**	
Module 3	**BASIC UNIX COMMANDS**		
	i) Listing Files & Directories. ii) Copying, Deleting, Renaming, Comparing, Splitting, Linking Files. iii) Creating, Navigating, Removing Directories. iv) Setting Access permission of files & directories. v) Using VI editor of UNIX. vi) Paging & Printing Files. vii) Status of users terminals & setting terminal Characteristics. viii) Cutting, Pasting, Sorting of Files. ix) Searching for a pattern in string. x) Process Status, Process Killing	**8**	
Module 4	**BASIC SYSTEM ADMINISTRATION**		
	i) Adding & Modifying Users accounts, Cont rolling Password. ii) Creating & Mounting File System. iii) init process & inittab startup files, Run levels. iv) Managing Disk Space(df , du , cpio) v) Searching Files with find command vi) Using ftp protocol to move files between computers. vii) 'Shutdown' command.	**10**	

Module 5	**SHELL PROGRAMMING**		
	i) Shell Script ii) System variables & shell variables. iii) Looping statements; conditional statements; case statements. iv) Logical operators, Mathematical expression. v) Command line parameters. Positional parameters. vi) String handling.	**10**	
Module 6	**Process**		
	i) Process creation using fork() system call, Parent process & child process ii) Multiple Processes creation iii) Process identification using getpid(),getppid() iv)Waiting for a process, process completion: wait(), sleep(), exit() v) Orphan process, zombie process.	**8**	

COMMUNICATION TECHNIQUE

Subject Code	Course offered in	Course Duration	3 lecture contact periods	Full Marks
CSWT / 4 / T2/CT	Part – II Second Semester	17 weeks	per week	70

O B J E C T I V E

This course concentrates on the field of analog communication and pulse code modulation. It also includes the advantages and disadvantages of digital and analog communications. After passing through the course the students will be acquainted with the basic telephony systems. Upon successful completion of this course the students will be able to: —

1. know the basic requirements of an analog communication system;
2. understand analog modulation including PAM, PWM and PPM;
3. know the functioning of transmitter and receiver;
4. explain the difference between digital and analog communication;
5. discuss the ideas dealing with the operation of the systems like telephony.

.

M O D U L A R D I V I S I O N O F T H E S Y L L A B U S

GROUP	MODULE	TOPIC	CONTACT PERIODS
A	1	**Module 1. Introduction to Electronic Communication**	4
	2	Analog Modulation	6
	3	Transmitting Systems	3
B	4	Demodulation	4
	5	Receiving System	4
C	6	Basic Telephony	4
	7	Pulse Code Modulation	5
D	8	MULTIPLEXING	5
	9	RF MODULATION FOR BASE BAND SIGNAL	5
	10	ANTENNA & WAVEGUIDE	5

TOTAL PERIODS: 51

EXAMINATION SCHEME

GROUP	MODULE	OBJECTIVE QUESTIONS				SUBJECTIVE QUESTIONS			
		TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS	TO BE SET	TO BE ANSWERED	MARKS PER QUESTION	TOTAL MARKS
A	1,2,3	10	20	1	20	3	5(at least 1 from each group)	10	50
B	4,5	5				2			
C	6,7	5				2			
D	8,9,10	10				3			

DETAIL COURSE CONTENT

Name of Course: COMMUNICATION TECHNIQUE	
Subject Code : CSWT / 4 / T2 / C T	Semester: Fourth
Duration : 17 weeks	**Maximum Marks : 100**
Teaching Scheme : 20 +10	**Examination Scheme : 70**
Theory: 3 contact hours/week.	Internal Examination : 20 Marks
Tutorial :-1 contact hour/week	Class Attendance : 5 Marks
Practical : 3 contact hours/week.	End Semester Examination : 70 Marks
Credit : 3	Teacher's Assessment : 5 Marks

	Content (Name of Topic)	Hrs/ Credit	MARKS
	Group A		
Module 1	**Introduction To Electronic Communication**	4	
	Importance of communication – Elements of a communication system – Types of electronic communication - Electromagnetic spectrum –		

	Bandwidth – Basic idea of Fourier series and Fourier transform.		
Module 2	**Analog Modulation**	6	
	Concept and necessity of modulation Definition of amplitude, frequency and phase modulation Derivation of sidebands in AM systems – Evaluation of power – Sideband depth – Percentage of modulation METHODS OF AM**:** Principles of operation of plate modulated Class C amplifier – Balanced modulator Expression of sidebands in FM and PM systems and its interpretation – Modulation index and bandwidth requirement Comparison of AM, FM and PM Basic ideas of Pulse Amplitude Modulation (PAM), Pulse Width Modulation (PWM) and Pulse Position Modulation (PPM) – Principle of generation and reception of PAM, PWM & PPM with block diagram and their applications.		
Module 3	**Transmitting Systems**	3	
	Block diagram and function of different stages of AM and FM broadcast transmitter		
	Group: B		
Module 4	**Demodulation**	4	
	Principle of detection with diode detector		
Module 5	**Receiving System**	**4**	
	Block diagram and principle of operation of super heterodyne receiver – IF amplifier and choice of IF – Mixer and converter – Alignment and tracking – Tone and volume control – Band spreading – Receiver characteristics – Testing Block diagram and principle operation of FM receiver .		
	Group :C		
Module 6	**Basic Telephony**	**4**	
	Telephone transmitter – Receiver – Dial tone, side tone and antisidetone circuits – Handset – Ringer – Switch hook – Hybrid – Local loop – Tone dialling – DTMF ELECTRONIC EXCHANGE**:** Space division switching, time division switching, block diagram of electronic exchange – concept of PBX and EPABX		
Module 7	**Pulse Code Modulation**	**5**	
	Idea of digital communication – Advantages of digital communication over analog communication BASIC STEPS IN PCM SYSTEM**:** Filtering – Sampling – Quantizing – Encoding – Line coding (HDB3, AM1, CM1, NRZ, RZ) Block schematic description of transmitter and receiver of PCM system.		

Group : D			
Module 8	**MULTIPLEXING**	**5**	
	IDEA of multiplexing and its necessity. TYPES of multiplexing**:** TDM and FDM TDM**:** Principles of time division multiplexing and synchronization in a digital communication system. PCM – TDM in modern applications (plesiochronous digital hierarchy and synchronous digital hierarchy). Frequency division multiplexing with practical examples, phase locked loop. Merits and demerits of TDM and FDM.		
Module 9	**RF MODULATION FOR BASE BAND SIGNAL**	**5**	
	Concepts of binary modulation techniques. Principles of amplitude shift keying, frequency shift keying and phase shift keying. Comparison between ASK, FSK and PSK.		
Module 10	**ANTENNA & WAVEGUIDE**	**5**	
	BASIC PRINCIPLES of antenna — Different types of antenna**:** Dipole antenna – Half wave and folded, microwave antenna – Horn antenna, parabolic antenna WAVE GUIDES**:** Rectangular – Circular Wave Guide Modes.		

REFERENCE BOOKS

1. Communication Electronics / Frenzel / Tata McGraw-Hill
2. Electronic Communication System / Kennedy / Tata McGraw-Hill
3. Principles of Communication System / Taub & Schilling / Tata McGraw-Hill
4. Electronic Communication / Roddy & Coolen / Prentice Hall of India, N. Delhi
5. Communication System / Simon Haykin / WI Ltd.
6. Telemetry Principles / D. Patranabis / Tata McGraw-Hill
7. Electronic Communication System / Dungan / Vikash Publishing House

COMMUNICATION TECHNIQUE LAB

Subject Code	Course offered in	Course Duration	2 sessional contact periods	Full Marks
CSWT / 4 / PR/LCTL	Part – II Second Semester	17 weeks	per week	50

OBJECTIVE

On satisfactory completion of the course, the students should be in a position to develop the skills complementary to the knowledge acquired in the theoretical subject COMMINICATION TECHNIQUE.

DETAIL COURSE CONTENT

1. To study the amplitude modulation and demodulation technique.
2. To study the frequency modulation and demodulation technique.
3. To study the frequency spectrum of AM and FM with the help of spectrum analyzer.
4. To study the analog signal sampling and reconstruction of the effect of**: —**
   (a) different sampling frequencies on reconstructed signals;
   (b) varying duty cycle of sampling frequency on the amplitude of reconstructed signal.
5. To study some radio receiver measurements**:** (a) sensitivity, (b) selectivity and (c) fidelity.
6. To study EPABX**:**
   (a) to study the electrical behaviour of different tones – dial tone, ringing tone, ring back tone and busy tone (both subscriber and exchange);
   (b) to study some extension features-redial, burgling, extension privacy, call forwarding, follow me etc.
7. Study of PCM transmission and reconstruction**:—**
   (a) To study the TDM and sampling of analog signal and its PCM form in the transmitter and the demultiplexing and reconstruction at the receiver section; and,
   (b) to study the AD and DA conversion.
8. To generate and detect ASK, FSK and PSK.
9. To be familiar with rectangular and circular wave guide.

Development of Life Skills-II

UNDER CONSTRUCTION

VISUAL BASIC

Subject Code	Course offered in	Course Duration	2 sessional contact periods	Full Marks
CSWT / 4 / PR/VB	Part – II Second Semester	17 weeks	per week	50

EXAMINATION SCHEME

Continues internal assessment of 50 marks is to be carried out by teachers throughout Part – II Second Semester.
Distribution of marks: Performance of Job – 35, Notebook – 15.

DETAIL COURSE CONTENT

Job 1 Concept of VB program. Environment of VB. Concept of project forms etc. Managing with menus. Drag and Drop operation.

Job2 Understanding Properties, Methods, Events-Visual Basic arithmetic operator. Validating and processing user inputs.

Job 3 Visual Basic Programming Fundamental, Understanding Variable names . Variable types- Range of Variable values, Working with String function, Numerical function.

Job 4 Working with Form & Form Events: Form properties . Working with following Form Tool (Tool box, Tool Bar, Menu Bar, Colour Palate), Creating, Opening, Saving and Running VB Projects

Job 5 Custom Control-Picture Box-Label Control Text Box-Command Button-Shape-Frame-Check Box-Radio button-Combo Box-List Box etc.

Job 6 Understanding Focus-Setting Tab Order.

Job 7 Prompting user with Dialog Boxes.

Job 8 Programming with various objects.

Job 9 Working with Control array. Creating and accessing Arrays, Control Arrays, Sorting and Searching

Job 10 Working with Module, class module, MDI, DLL, MDI Form.

Job 11 Creating own menu using menu editor, Advanced controls: Common dialog box, Tree view, List view, rich text box control, windows common controls, status bar, tab control, image list, MS chart.

Job 12 Working with Database management System [MS Access], Data Control, Relational Databases & SQL Data Access Techniques, Using Recordsets.

Job13 File Handling in VB**,** File commands, file handling functions, Sequential files, Random access files, Binary files. Reading information from a file, Adding to an existing file.

REFERENCE BOOKS

1. Beginning Visual Basic 6 by : Peter Wright (SPD/WROX)
2. Visual Basic 6 from the Ground Up, Cornell,TMH
3. Visual Basic 6, CDG, TMH
4. Visual Basic 6,Dietel, Pearson
5. Visual basic 6.0 in 30 days, Krishnan, Scitech