	1	Design suitable data structures and implement Pass-I of a two-pass macro-processor. The output of Pass-I (MNT, MDT and intermediate code file without any macro definitions) should be input for Pass-II.
		MACRO
		INCR &X, &Y, &REG1 = AREG
		MOVER &REG1, &X
		ADD &REG1, &Y
		MOVEM &REG1, &X
		MEND
		MACRO
		DECR &A, &B, &REG2 = BREG
		MOVER &REG2 ,&A
		SUB &REG2, &B
		MOVEM &REG2, &A
		MEND
		START 100
		READ N1
		READ N2
		DECR N1, N2
		INCR N1, N2
		STOP
		N1 DS 1
		N2 DS 2
		END

	2	Design suitable data structures and implement Pass-I and Pass-II of a two-pass macro-processor. The output of Pass-I (MNT, MDT and intermediate code file without any macro definitions) should be input for Pass-II.

		Input.txt
		MACRO
		INCR &X, &Y, &REG1
		MOVER &REG1, &X
		ADD &REG1, &Y
		MOVEM &REG1, &X
		MEND
		MACRO
		DECR &A, &B, &REG2
		MOVER &REG2, &A
		SUB &REG2, &B
		MOVEM &REG2, &A
		MEND
		START 100
		READ N1
		READ N2
		INCR N1, N2
		DECR N1, N3
		STOP
		N1 DS 1
		N2 DS 2
		N3 DS 1
		END
		
		MDT.txt
		INCR &X &Y &REG1 = AREG
		MOVER #3 #1
		ADD #3 #2
		MOVEM #3 #1
		MEND
		DECR &A &B &REG2 = BREG
		MOVER #6 #4
		SUB #6 #5
		MOVEM #6 #4
		MEND
		
		MNT.txt
		INCR 0 3
		DECR 5 3
		
		ARG.txt
		&X
		&Y
		&REG1 AREG
		&A
		&B
		&REG2 BREG

	3	Write a program to create a Dynamic Link Library for any mathematical operation and write an application program to test it. (Java Native Interface / Use VB or VC++)
	4	Write a program to solve Classical Problems of Synchronization using Mutex and Semaphore (Reader Writer)
	5	Design a Paper Prototyping for any Banking Website or App.
	6	Design Paper Prototyping for any ERP system.

	7	Write a program to simulate CPU Scheduling Algorithms: FCFS, SJF (Preemptive) Process AT BT P1 10 2 P2 0 10 P3 8 4 P4 5 5
	8	Write a program to simulate CPU Scheduling Algorithms: FCFS, Priority (Non-Preemptive). Process AT BT P1 10 2 P2 0 10 P3 8 4 P4 5 5

	9	Write a program to simulate CPU Scheduling Algorithms: FCFS and Round Robin. Process AT BT P1 10 2 P2 0 10 P3 8 4 P4 5 5
	10	Write a program to simulate Memory placement strategies – best fit, first fit.
	11	Write a program to simulate Memory placement strategies – best fit, worst fit.
	12	Write a program to simulate Page replacement algorithm. 1. FIFO 2. LRU Input reference String :- 2 3 2 1 5 2 4 5 3 2 5 2 No. of frames are:- 3
	13	Write a program to simulate Page replacement algorithm. 1. FIFO 2. OPTIMAL Input reference String :- 2 3 2 1 5 2 4 5 3 2 5 2 No. of frames are:- 3
	14	Write a program to implement Deadlock Avoidance Algorithm
		No. of Resources 4 No. of Process 3 Max Matrix 3 3 2 2 1 1 3 4 1 3 5 0 Allocation Matrix 1 2 2 1 1 0 3 3 1 2 1 0 Available 3 1 1 2
	15	Design suitable data structures and implement pass-I of a two-pass assembler for pseudo-machine in Java using object oriented feature. start 100 movr ax 05 mover bx 10 up: add ax bx movem a ='5' mul ax a origin up ltorg movem b ='8' movem c ='8' ltorg movem b ='7' movem c ='8' ds a 02 dc b 10 ds c 09 next equ up end

	16	Implement Pass-II of two pass assembler for pseudo-machine in Java using object oriented features. The output of assignment-1 (intermediate file and symbol table) should be input for this assignment.

(AD,1) (C,100)
100 (IS,5) (RG,1) (C,05)
101 (IS,5) (RG,2) (C,10)
102 (S,1) (IS,2) (RG,1) (RG,2)
103 (IS,6) (S,2) (L,1)
104 (IS,4) (RG,1) (S,1)
105 (AD,3) (C,102)
102 (DL,1) (C,5)
103 (IS,6) (S,3) (L,2)
104 (IS,6) (S,4) (L,3)
105 (DL,1) (C,8)
106 (DL,1) (C,8)
107 (IS,6) (S,2) (L,4)
108 (IS,6) (S,3) (L,5)
109 (DL,1) (C,02)
110 (DL,2) (C,10)
111 (DL,1) (C,09)
112 (S,5) (AD,4) (S,1)
113 (AD,2)
(DL,1) (C,7)
114 (DL,1) (C,8)

SYMBOL	ADDRESS
up	102
a	109
b	110
c	111
next	102

LITERAL	ADDRESS
5	102
8	105
8	106
7	113
8	114