Duration : 3 Hours   **Maximum Marks : 80**

1. Implement a menu driven simulator for hypothetical Simple Instruction Computer that provides the following functionalities:
   a. Load – Loading of the program from file into memory
   b. Print – Printing the program loaded in memory.
   c. Run – Executing the loaded program
   The machine has the basic instruction set comprising of following:

Opcode	Mnemonic	Instruction
00	STOP	Stop or half execution
01	ADD	Add memory operand to register contents
02	SUB	Subtract memory operand from register contents
03	MULT	Multiply memory operand to register contents
04	MOVER	Move memory operand contents to register contents
05	MOVEM	Move register contents to memory operand
06	COMP	Compare register and memory operands to set condition code appropriately
07	BC	Branch to second operand depending on condition code specified as first operand
08	DIV	Divide register contents by memory operand
09	READ	Read into memory operand
10	PRINT	Print contents of memory Operand

   Create a file containing the machine language code to find maximum of two numbers by using above instruction opcode set.   [35]

2. Write a simulation program to implement demand paging using LFU page replacement algorithm. Assume the memory of "n" frames. Show the contents of page after every page replacement in a frame and it at end we told number of page faults accordingly. Input the following page reference string.

   Page reference string : 3,4,5,4,3,4,7,2,4,5,6,7,2,4,6   [35]

3. Viva   [10]

Duration : 3 Hours   **Maximum Marks : 80**

1. Implement a menu driven simulator for hypothetical Simple Instruction Computer that provides the following functionalities:
   a. Load – Loading of the program from file into memory
   b. Print – Printing the program loaded in memory.
   c. Run – Executing the loaded program
   The machine has the basic instruction set comprising of following:

Opcode	Mnemonic	Instruction
00	STOP	Stop or half execution
01	ADD	Add memory operand to register
02	DIV	Divide register contents by memory operand
03	SUB	Subtract memory operand from register
04	MULT	Multiply memory operand to register
05	READ	Read into memory operand
06	COMP	Compare register and memory operands to set condition code appropriately
07	MOVEM	Move memory operand contents to register
08	BC	Branch to second operand depending on condition code specified as first operand
09	MOVER	Move register contents to Memory operand
10	PRINT	Print contents of memory Operand

   Create a file containing the machine language code to find minimum of two numbers by using above instruction opcode set.   [35]

2. Write a simulation program to implement Round Robin CPU scheduling algorithm for the given time quantum as input. Also accept the number of processes and arrival time and CPU burst time for each process as input. The output should give the Gant Chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time.   [35]

3. Viva   [10]

Duration : 3 Hours   **Maximum Marks : 80**

1. Implement a menu driven simulator for hypothetical Simple Instruction Computer that provides the following functionalities:
   a. Load – Loading of the program from file into memory
   b. Print – Printing the program loaded in memory.
   c. Run – Executing the loaded program

   The machine has the basic instruction set comprising of following:

Opcode	Mnemonic	Instruction
00	STOP	Stop or half execution
01	READ	Read into memory operand
02	PRINT	Print contents of memory Operand
03	ADD	Add memory operand to register contents
04	SUB	Subtract memory operand from register contents
05	MOVER	Move Memory operand contents to register contents
06	MOVEM	Move register contents to memory
07	MULT	Multiply memory operand to register contents
08	DIV	Divide register contents by memory operand
09	BC	Branch to second operand depending on condition code specified as first operand
10	COMP	Compare register and memory operands to set condition code appropriately

   Create a file containing the machine language code to find $x^y by$ using above instruction opcode set. [35]

2. Write a simulation program to implement demand paging using LRU (Using Counter) page replacement algorithm. Assume the memory of "n" frames. Show the contents of page after every page replacement in a frame and at end show the total number of page faults accordingly. Input the following Page Reference String.
   Page Reference String :12,15,12,18,6,8,11,12,19,12,6,8,12,15,19,8 [35]

3. Viva [10]

CS – 347 Lab Course – I System Programing & Operating system
Duration : 3 Hours   **Maximum Marks : 80**

1. Implement a menu driven simulator for hypothetical Simple Instruction Computer that provides the following functionalities:
   a. Load – Loading of the program from file into memory
   b. Print – Printing the program loaded in memory.
   c. Run – Executing the loaded program
   The machine has the basic instruction set comprising of following:

Opcode	Mnemonic	Instruction
00	STOP	Stop or half execution
01	PRINT	Print contents of memory Operand
02	SUB	Subtract memory operand from register contents
03	MULT	Multiply memory operand to register contents
04	READ	Read into memory operand
05	MOVER	Move Memory operand contents to register contents
06	MOVEM	Move register contents to Memory
07	COMP	Compare register and memory operands to set condition code appropriately
08	BC	Branch to second operand depending on condition code specified as first operand
09	DIV	Divide register contents by memory operand
10	ADD	Add memory operand to register contents

   Create a file containing the machine language code to find factorial of a given number by using above instruction opcode set.   [35]

2. Write a program to implement a toy shell (Command Interpreter). It has its own prompt say "MyShell $". Any normal shell command is executed from this shell (MyShell$) by starting a child process to execute the system program corresponding to the command. It should additionally interpret the following commands:

   list f dirname   :   To print names of all the files in current directory
   list n dirname   :   To print the number of all entries in the current directory.
   list i dirname   :   To print names and inodes of the files in the current directory.   [35]

3. Viva   [10]

1. Implement a menu driven simulator for hypothetical Simple Instruction Computer that provides the following functionalities:
   a. Load – Loading of the program from file into memory
   b. Print – Printing the program loaded in memory.
   c. Run – Executing the loaded program
   The machine has the basic instruction set comprising of following:

Opcode	Mnemonic	Instruction
00	STOP	Stop or half execution
01	ADD	Add memory operand to register contents
02	SUB	Subtract memory operand from register contents
03	MULT	Multiply memory operand to register contents
04	DIV	Divide register contents by memory operand
05	MOVER	Move Memory operand contents to register contents
06	MOVEM	Move register contents to Memory
07	COMP	Compare register and memory operands to set condition code appropriately
08	BC	Branch to second operand depending on condition code specified as first operand
09	READ	Read into memory operand
10	PRINT	Print contents of memory Operand

   Create a file containing the machine language code to find the addition of first "n" numbers by using above instruction opcodes   [35]

2. Write a simulation program to implement sequential (Contiguous) Allocation method for file. Assume disk having "n" number of blocks ranging from 0 to n-1. Randomly mark some of the blocks as allocated and accordingly maintain the list of free blocks. Perform the following operation using following menu.
   - Show Bit Vector
   - Create New File
   - Show Directory
   - Delete File
   - Exit   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line argument; an empty file is opened for editing if no argument is supplied. It should display a '$' prompt to accept the line editing commands. Implement the following commands:

   a   :   Append
   p   :   Print all lines
   s   :   Save
   e   :   Exit   [35]

2. Write a simulation program to implement demand paging using MFU page replacement algorithm. Assume the memory of "n" frame. Show the contents of page after every page replacement in a frame and at end show the total number of page faults accordingly. Input the following Page Reference String.

   Page Replacement String: 8, 5, 7, 8, 5, 7, 2, 3, 7, 3, 5, 9, 4, 6, 2   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line argument. An empty file is opened for editing if no argument is supplied. It should display '$' prompt to accept the line editing commands. Implement the following commands:

   a   :   Append
   d n :   Delete n^{th} line.
   p   :   Print all lines.
   e   :   Exit   [35]

2. Write the simulation program to implement demand paging using FIFO page replacement algorithm. Assume the memory of "n" frames. Show the contents of page after every page replacement in a frame and at end show the total number of page faults accordingly. Input the following Page Reference String.
   Page Reference String: 3,4,5,6,3,4,7,3,4,5,6,7,2,4,6   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line argument. An empty file is opened for editing if no argument is supplied. It should display '$' prompt to accept the line editing commands. Implement the following commands.

    s   :   Save
    a   :   Append
    d n1 n2   :   Delete lines from position n1 to n2
    e   :   Exit   [35]

2. Consider the following snapshot of the system

Processes	Allocation			MAX			Available		
	A	B	C	A	B	C	A	B	C
P₀	0	1	0	7	5	3	3	3	2
P₁	2	0	0	3	2	2			
P₂	3	0	2	9	0	2			
P₃	2	1	1	2	2	2			
P₄	0	0	2	4	3	3			

    Calculate and display the contents of matrix Need.
    Using Safety and Resource-Request Algorithm perform the following operations:

    a) Check whether the current system is in safe state or not.
    b) If a request from process P₁ arrives as (1, 0, 2), can it be granted immediately by keeping system in safe state.   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a program to implement a DFA Driver for any given language. Accept number of states, number of input symbols, set of input symbols, number of final states, set of final states and transition table as input. (Input language should be given by examiner). Write menu driven program that will have options as:
   a. Read DFA
   b. Show Transition Table
   c. Check Acceptance of the Given String.
   d. Exit   [35]

2. Write a simulation program to implement demand paging using FIFO page replacement algorithm. Assume the memory of "n" frames. Show the contents of page after every page replacement in a frame and at end show the total number of page faults accordingly. Input the following Page Reference String.

   Page Reference String : 3,4,5,4,3,47,2,4,5,6,7,2,4,6   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a program to implement DFA driver for the language L = "Set of all string that starts with a, ending with b" over {a,b}.   [35]

2. Write a simulation program to implement a Pre-emptive Shortest Job First (SJF) – CPU scheduling algorithm. Accept the number of Processes as input. Also accept arrival time and CPU burst time for each process as input. The output should give the Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time.   [35]

3. Viva.   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line arguments. An empty file is opened for editing if no argument is supplied. It should display a '$' prompt to accept the line editing commands. Implement the following commands:

   a   :   Append
   p m n   :   Print range of lines from m to n
   i n   :   Insert line at position n.
   e   :   Exit   [35]

2. Write a simulation program to implement a linked allocation method for file. Assume disk having "n" number of blocks ranging from 0 to n-1. Randomly mark some blocks as allocated and accordingly maintain the list of free blocks. Perform the following operation using the following menu:
   - Show Bit Vector
   - Create New File
   - Show Directory
   - Delete File
   - Exit   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a program to implement a DFA driver for the language L = "Set of all strings that containing 101 as substring" over {0,1}. [35]

2. Write a simulation program to implement Pre-emptive Priority CPU scheduling algorithm. Accept the number of processes, arrival time, CPU burst time and priority for each process as input. Priorities should in High to Low order (1 is High). The output should give the Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time. [35]

3. Viva [10]

1. Consider the following Macro definition and Data structures (stored as an array of structure) association with it as follows: [35]

   MACRO
   CALC   &A,&B,&OP=ADD
   MOVER   AREG, &A
   &OP   AREG, &A
   MOVEM   AREG, &A
   MEND

   Macro Definition Table
   MOVER AREG, (P, 1)
   (P, 3) AREG, (P, 2)
   MOVEM AREG, (P, 1)
   MEND

   Parameter Name Table

Parameter Name
&A
&B
&OP

   Macro name Table

Macro Name	MDTPTR	KPDTPTR	PNTPTR	No. Keyword Parameter	No. Positional Parameter
CALC	0	0	0	1	2

   Keyword Parameter Default Value Table

Parameter Name	Default Value
&OP	ADD

   Write a program to expand the above macro definition using given data structure for the following two macro call
   1. CALC   X,Y
   2. CALC   X,Y,&OP=SUB

2. Write a program to implement a toy shell (Command Interpreter). It has its own prompt say "MyShell $". Any normal shell command is executed from this shell (MyShell$) by starting a child process to execute the system program corresponding to the command.

    It should additionally interpret the following commands:

    search f file name :- To search first occurrence of the pattern in the file
    search a file name :- To search all the occurrence of the pattern in the file
    search c file name :- To count the number of occurrence of the pattern in the file.
                                                                                [35]
3. Viva.   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a macro pre-processor program that will create and display the contents of MDT, MNT and PNTAB for the following macro definition: [35]

   ```
   MACRO
   CALC        &A,&B,&REG,&OP
   MOVER       &REG,     &A
   &OP         &REG,     &B
   &MOVEM      &REG,     &A
   MEND
   ```

2. Write a program to implement a toy shell (Command Interpreter). It has its own prompt say "MyShell $". Any normal shell command is executed from this shell (MyShell$) by starting a child process to execute the system program corresponding to the command.

   It should additionally interpret the following commands:

typeline +n filename	:- To print first n lines in the file.
typeline -n filename	:- To print last n lines in the file.
typeline a filename	:- To print all lines in the file. [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a macro pre-processor program that will create and display the contents of MNT, MDT and KPDTAB for the following macro definition

   ```
   MACRO
   INCR           &X=,&Y=,&OP=ADD, &REG=AREG
   MOVER          &REG,    &X
   &OP            &REG,    &Y
   MOVEM          &REG,    &X
   MEND
   ```
   [35]

2. Write a simulation program to implement FCFS CPU-scheduling algorithm. Accept the number of Processes as input. Also accept arrival time and CPU burst time for each process as input. The output should give the Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time. [35]

3. Viva [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write an assembler that will create and display the contents of symbol table. Also construct the Intermediate Code of type variant I for the following error-free assembly program:

```
            START       100
            MOVEM       BREG,    ONE
            MOVER       BREG,    A
            MOVER       BREG,    B
            MOVER       BREG,    C
            PRINT       A
            PRINT       B
            PRINT       C
            STOP
A           DS          1
B           DS          2
C           DS          1
ONE         DC          '1'
            END
```

2. Write a program to implement a toy shell (Command Interpreter). It has its own prompt say "MyShell $". Any normal shell command is executed from this shell (MyShell$) by starting a child process to execute the system program corresponding to the command.

   It should additionally interpret the following commands:

   count c filename   :- To print number of characters in the file.
   count w filename   :- To print number of words in the file.
   count 1 filename   :- To print number of lines in the file.

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write an assembler that will create and display the contents of symbol table. Also construct the Intermediate Code of type variant II for the following error-free assembly program:

```
            START       100
            MOVEM       BREG,    ONE
            MOVER       BREG,    A
            MOVER       BREG,    B
            MOVER       BREG,    C
            PRINT       A
            PRINT       B
            PRINT       C
            STOP
A           DS          1
B           DS          2
C           DS          1
ONE         DC          '1'
            END
```

2. Write a program to implement a toy shell (Command Interpreter). It has its own prompt say "MyShell $". Any normal shell command is executed from this shell (MyShell$) by starting a child process to execute the system program corresponding to the command.

   It should additionally interpret the following commands:

   count c filename   :- To print number of characters in the file.
   count w filename   :- To print number of words in the file.
   count 1 filename   :- To print number of lines in the file.

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line argument; an empty file is opened for editing if no argument is supplied. It should display a '$' prompt to accept the line editing commands. Implement the following commands:

   s   :   Save
   a   :   Append
   c nl n2 :   Copy line from position n1 to position n2
   e   :   Exit   [35]

2. Write a simulation program to implement a Linked Allocation method for file. Assume disk having "n" number of blocks ranging from 0 to n-1. Randomly mark some blocks as allocated and accordingly maintain the list of free blocks. Perform the following operation using the following menu:
   - Show Bit Vector
   - Create New File
   - Show Directory
   - Delete File
   - Exit   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line argument. An empty file is opened for editing if no argument is supplied. It should display a '$' prompt to accept the line editing commands. Implement the following commands:

   s   :   Save
   a   :   Append
   m nl n2 :   Copy line from position n1 to position n2
   e   :   Exit   [35]

2. Write a simulation program to implement a Non-Pre-emptive Shortest Job First (SJF) – CPU scheduling algorithm. Accept the number of Processes and arrival time and CPU burst time for each process as input. The output should give the Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time.   [35]

3. Viva.   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a macro pre-processor program that will create and display the contents of MNT, PNTAB, and KPDTAB for the following macro definition.

   ```
   MACRO
   CALC        &A,&B,&REG=CREG,&OP=MULT
   MOVER       &REG,    &A
   &OP         &REG,    &B
   MOVEM       &REG,    &A
   MEND                                            [35]
   ```

2. Write a simulation program to implement non-pre-emptive Priority CPU scheduling algorithm. Accept the number of Processes and arrival time, CPU burst time and priority for each process as input. Priorities should in High to Low order (1 is High). The output should give the Gantt chart, turnaround time and waiting time for each process. Also display the average turnaround time and average waiting time. [35]

3. Viva. [10]

1. Write an assembler that will display the following errors and warnings:
   a) Symbol used but not defined
   b) Symbol defined but not used
   c) Re-declaration of the Symbol
   Present in the following assembly program:

   ```
               START    100
               READ     X
   LOOP        READ     BREG,    X
               ADD      BREG,    Y
   X           MOVEM    AREG,    Z
               STOP
   X           DS  1
   Y           DS  1
               END
   ```

2. Write a program to implement a toy shell (Command Interpreter). It has its own prompt say "MyShell $". Any normal shell command is executed from this shell (MyShell$) by starting a child process to execute the system program corresponding to the command.

   It should additionally interpret the following commands:

   count c filename   :- To print number of characters in the file.
   count w filename   :- To print number of words in the file.
   count l filename   :- To print number of lines in the file.

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write an assembler that will display the following errors and warnings:
   a. Symbol used but not defined
   b. Symbol defined but not used
   c. Re-declaration of the Symbol
   Present in the following assembly program:

   ```
                START   100
                READ    A
   A            READ    BREG,   A
                ADD     BREG,   B
   LOOP         MOVEM   AREG,   C
                STOP
   A            DS  1
   B            DS  1
                END
   ```

2. Write the simulation program to implement demand paging using FIFO page replacement algorithm. Assume the memory of "n" frames. Show the contents of page after every page replacement in a frame and at end show the total number of page faults accordingly. Input the following Page Reference String.
   Page Reference String: 3,4,5,6,3,4,7,3,4,5,6,7,2,4,6   [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a program to implement DFA driver for the language L = "Set of all strings that containing 101 as substring" over {0, 1}. [35]

2. Write a program to implement a Banker's Algorithm. Accept the total number of processes (n) and resource types (m) as input. Also accept the number of initial instances for each resource type. Allocation and Max of size "nxm" as input and perform the following operations:
   a) Calculate and display the final contents of Available array of size "m"
   b) Calculate and display the contents of Need matrix of size "nxm"
   Using Safety and Resource-Request algorithm perform the following operations:
   a) Check whether system is in safe sate or not.
   b) If request of size "m" arrives from process P_i, can it be granted immediately by keeping system in safe state?
   [35]

3. Viva. [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Write a command line program for line editor. The file to be edited is taken as command line arguments. An empty file is opened for editing if no argument is supplied. It should display a '$' prompt to accept the line editing commands. Implement the following commands:

   i n   :   Insert line at position n.
   d n1 n2 :   Delete line between position n1 and n2
   s   :   Save
   p   :   Print all lines
   e   :   Exit   [35]

2. Consider the following snapshot of the system

Processes	Allocation				MAX				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P_0	0	0	1	2	0	0	1	2	1	5	2	0
P_1	1	0	0	0	1	7	5	0				
P_2	1	3	5	4	2	3	5	6				
P_3	0	6	3	2	0	6	5	2				
P_4	0	0	1	4	0	6	5	6				

   Calculate and display the contents of matrix Need.
   Using Safety and Resource-Request Algorithm perform the following operations:

   a) Check whether the current system is in safe state or not.
   b) If a request from process P_4 arrives as (0, 0, 4, 1), can it be granted immediately by keeping system in safe state.

                                                                                          [35]

3. Viva   [10]

Duration: 3 Hours   **Maximum Marks: 80**

1. Consider the following Macro definition and Data structures (stored as an array of structure) association with it as follows: [35]

   MACRO
   CALC   &A,&B,&OP=
   MOVER   AREG, &A
   &OP   AREG, &B
   MOVEM AREG, &A
   MEND

   Macro Definition Table
   MOVER AREG, (P, 1)
   (P, 3) AREG, (P, 2)
   MOVEM AREG, (P, 1)
   MEND

   Parameter Name Table

Parameter Name
&A
&B
&OP

   Macro name Table

Macro Name	MDTPTR	KPDTPTR	PNTPTR	No. Keyword Parameter	No. Positional Parameter
CALC	0	0	0	1	2

   Write a program to expand the above macro definition using given data structure for the following two macro call
   1. CALC   X,Y,&OP=ADD
   2. CALC   X,Y,&OP=SUB

2. Write a simulation program to implement demand paging using MFU page replacement algorithm. Assume the memory of "n" frame. Show the contents of page after every page replacement in a frame and at end show the total number of page faults accordingly. Input the following Page Reference String.
   Page Replacement String: 3,4,5,4,3,4,7,2,4,5,6,7,2,4,6   [35]

3. Viva   [10]