

MCA 4.1

Operating Systems

Theory: Exam - 75

Sessionals - 25

Overview: Evolution, current status and future trends. Structural overview.

Process concepts: Process states, process control block, process scheduling algorithms.

Support for concurrent processes: Mutual exclusion, shared data, critical sections, busy form of waiting, lock and unlock primitives, synchronization, block and wakeup.

Interprocess communication issues, primitives.

System Deadlock: Prevention, detection and avoidance.

Memory management: Contiguous and non-contiguous allocation, virtual memory, segmentation, paging, page scheduling and replacement algorithms.

File Systems: Management, protection mechanisms.

I/O management, disk scheduling.

Design of UNIX, LINUX- a case study.

An overview of network and distributed operating systems.

BOOKS:

1. Silberschatz A, Galvin P: Operating system concepts, 4th ed. Addison-Wesley Publishing Co.

2. Milenkovic M.: Operating System- Concepts and Design, McGraw Hill 3. Tanenbaum A S: Operating System- Design and Implementation, PHI(EEE).

4. Bach, M.: Design of the UNIX operating system, PHI(EEE).

## 4.2 Formal Languages & Automata Theory

Alphabets, Languages, Grammars.

Finite Automata : Regular Expression, 2-way Finite Automata Regular languages.  
Regular Grammars.

Context free languages : Derivation trees, ambiguity Pushdown automata, DCFLS,  
LL(K) and LALR grammars .

Context Sever time languages : Linear bounded automate

Turing Machines : Techniques for constructions, church's hypothesis recursively  
enumerable languages.

Text Books :

1. Hopcroft and Ullman 'Introduction to Automata Theory languages and automation,  
Addison Wesley.

Theory: Exam - 75

Sessionals - 25

Introduction to software engineering, software development life cycle, planning a software project, problem analysis and requirement specification, DFD and DD, design of solution strategy.

Software Economics:- Cost estimation and evaluation techniques- cost estimation based on COCOMO models and Reileigh model for software cost estimation.

Software design:- Modern design techniques- topdown and bottom up design, structured programming, structured design methodology, modularization, various approaches to programming- automatic programming, defensive programming, redundant programming etc, software complexity. Data and object oriented design methodologies. Aspects of programming language and their effects on software.

Software verification and testing:- Theoretical foundation, black box and white box approaches, integration and system testing.

Software Reliability:- Definitions and concept of reliability, software faults, errors, repair and availability. reliability and availability models.

Case studies.

Books:

1. Pressman, R.S., Software Engineering: A Practitioner's Approach, McGrawHill.
2. Shooman, M, Software Engineering, McGrawHill.
3. Fairley, R.E., Software Engineering Concepts, McGrawHill.



of the DBMS.

Part-III: Operating Systems:

1. Concurrent programming with race condition.
2. Concurrent programming with synchronization primitives.
3. Implementation of low level synchronization primitives for avoiding race conditions.
4. Implementation of high level synchronization tools and their usage in implementing various hardware as well as software resource schedulers.