# IT 307: Operating Systems

**(Elective )**

***Credits: 3***
***Lecture Hours:48***

**Course Objectives**

This module aims to provide the concepts of Operating Systems and Implementation of Systems Utilities for Inter-process communication in a multiprocessor environment.

**Course Description**

Overview, Process Management, Scheduling, Basic Synchronization principles, Memory Management, File Management, Protection and Security, Device Management

**Detailed Course**

**Unit 1: Overview** **LH 5**

- 1.1 Introduction
- 1.2 System Structures
- 1.3 The abstract Model of computing
- 1.4 Resources: files
- 1.5 Processes: Creating Processes (using C functions: FORK, JOIN, and QUIT, )
- 1.6 Threads: C threads

**Unit 2: Process Management** **LH 8**

- 2.1 The system view of processes and resources
- 2.2 Initializing the Operating System
- 2.3 Process address spaces
  - 2.3.1 Creating the address space
  - 2.3.2 Loading the program
  - 2.3.3 Maintaining consistency in the address space
- 2.4 The process abstraction
  - 2.4.1 Process descriptors
  - 2.4.2 Process state diagram
- 2.5 The resource abstraction
- 2.6 Process hierarchy
  - 2.6.1 Refining the process manager
  - 2.6.2 Specializing resource allocation strategies

**Unit 3: Scheduling** **LH 9**

- 3.1 Scheduling Mechanisms
  - 3.1.1 The process scheduler organization
  - 3.1.2 Saving the process context
  - 3.1.3 Voluntary CPU Sharing
  - 3.1.4 Involuntary CPU Sharing
  - 3.1.5 Performance

3.3 Strategy Selection

3.1.1 Partitioning s process into small processes

3.4 Nonpreemptive Strategies

3.1.1 First come first served

3.1.2 Shortest Job next

3.1.3 Priority Scheduling

3.1.4 Deadline scheduling

3.5 Preemptive strategies

3.1.1 Round robin

3.1.2 Multiple-level queues

3.1.3 Monitors

**Unit 4: Basic Synchronization principles** **LH 5**

4.1 Interacting processes

4.1.1 Critical Sections

4.1.2. Deadlock

4.2 Coordinating processes

Semaphores

4.2.1 Principles of operation

Practical considerations

**Unit 5: Memory Management** **LH 8**

5.1 The Basics

5.1.1 Requirements on the primary memory

5.1.2 Mapping the address space to primary memory

5.1.3 Dynamic memory for data structures

5.2 Memory Allocation

5.2.1 Fixed-partition memory strategies

5.2.2 Variable-partition memory strategies

5.2.3 Contemporary Allocation Strategies

5.3 Dynamic Address Resolution

5.3.1 Runtime bound Checking

5.4 Memory Manager Strategies

5.4.1 Swapping

5.4.2 Virtual Memory

5.4.3 Shared-memory Multiprocessors

**Unit 6:File Management** **LH 5**

6.1 File System

6.1.1 File Concept

6.1.2 Access Methods

6.1.3 Directory Structure

6.1.4 File System Mounting

6.1.5 File Sharing

6.1.6 Protection

6.2 Implementing File Systems
- 6.2.1 File System Structure
- 6.2.2 File System Implementation
- 6.2.3 Directory Implementation
- 6.2.4 Allocation Methods
- 6.2.5 Free Space Management

6.3 Secondary Storage Structure
- 6.3.1 Disk Structure
- 6.3.2 Disk Scheduling
- 6.3.3 Disk Management
- 6.3.4 Swap Space Management

6.4 I/O Systems
- 6.4.1 I/O Hardware
- 6.4.2 Application I/O Interface

**Unit 7: Protection and Security** **LH 3**

7.1 Fundamentals
- 7.1.1 Policy and Mechanism
- 7.1.2 Implementing Policy and mechanism
- 7.1.3 Authentication Mechanisms
- 7.1.4 Authorization Mechanisms
- 7.1.5 Encryption

**Unit 8: Device Management** **LH 5**

8.1 Device Management approaches
- 8.1.1 I/O System Organization
- 8.1.2 Direct I/O with Polling
- 8.1.3 Interrupt-Driven I/O
- 8.1.4 Memory-Mapped I/O
- 8.1.5 Direct memory access

8.2 Device Drivers
- 8.2.1 The device driver interface
- 8.2.2 CPU-device interactions
- 8.2.3 I/O optimization

8.3 Some Device Management Scenarios
- 8.3.1 Serial Communications
- 8.3.2 Sequentially accessed storage devices
- 8.3.3 Randomly accessed devices

**Laboratory:**

- Students should implement operating system functionality in their project.

**Text Book:**

- Gary Nutt, Operating Systems A modern Perspective, Second edition, Pearson Education
- Silberschatz, A., Galvin, P. & Gagne, G., Operating System Principles, Seventh Edition, John Wiley & Sons

**References:**

- Andrew S. Tanenbaum, Modern Operating System, PHI
- Andrew S.Tanenbaum, Operating Systems Design and Implementation, Prentice Hall
- James L Peterson & Abraham Silberschatz, Operating System concepts, Addison Wesley
- Thomas W. Doeppner, Operating Systems in Depth, John Wiley & Sons