



**DBLN CPSI 3392**

This course is being offered at Griffith College, CAPA's academic partner in Dublin. The Irish academic system differs from the US, particularly with grading. Griffith College professors expect students to undertake a good deal of independent study to achieve a high mark in their classes. For additional information about this class, please contact the Boston Program Advising Team at 1-800-793-0334.

## **Operating System Design**

Continuous Assessment: 50%

Exam: 50%

### **Intended Module Learning Outcomes**

On successful completion of this module learners will be able to:

1. discuss the nature, origin and function of operating systems at a generic level and analyse issues of resource management, especially of processes and memory
2. analyse the role of operating systems in non-traditional environments such as embedded systems, concurrent systems and real-time systems
3. install various operating systems; evaluate and compare the interface, functionality and suitability of different operating systems
4. demonstrate an ability to manipulate an operating system using systems programming
5. display a knowledge of how the operating system relates to the underlying hardware and the software that runs on it

### **Module Objectives**

This module focuses on the concepts involved in the design of an operating system; an understanding of its complexity and its many requirements. It introduces the learner to some fundamental algorithms used in operating systems. It introduces the concept of concurrency in an OS; explores the concept of multiprocessing and distributed operating systems; and it provides a practical knowledge of operating systems.

### **Module Curriculum**

#### **Introduction to operating systems**

- Introduction: What is an Operating System;
- Types of OS: Batch, Time-Sharing, Personal Computer Systems, Parallel Systems, Distributed Systems; Operating System Structures: System Components, Operating System services, System calls, system programs, system structure.
- System Design and Implementation;
- Introduction to DOS, Linux and another commercial desktop operating system; emerging and experimental operating systems to be used for practical work.

## **Processor management**

- Definition of a process; process control block and process image; process states and life cycle.
- I/O bound jobs and CPU bound jobs
- Turnaround time, waiting time, execution time.
- CPU Scheduling: basic concepts and scheduling criteria.
- Scheduling algorithms: FCFS, Shortest job first, Priority, Shortest remaining, round robin

## **Process management**

- Threads - overview and benefits; user and kernel threads.
- Processes: Process concept, operations on processes, cooperating processes, inter-process communication;
- Deadlocks: System model, deadlock characterization, detection, methods for handling deadlocks, recovery from deadlock, deadlock prevention and avoidance.
- Synchronization: critical section problem, Two task solution, synchronization in hardware, Semaphores, classical synchronization problems, Dining philosopher problem.

## **Memory management**

- Memory management: fixed, dynamic and reloadable partitions memory management systems. Contiguous memory allocation, paging and segmentation;
- Virtual memory: demand paging, page wrapping, allocation of frames, thrashing.
- Internal and external fragmentation in different systems.
- Contiguous Memory Allocation
- Page strategies: FIFO, Most recently used, least recently used. Optimal page replacement
- Security aspect of memory management

## **File management**

- File Systems: File concept, access methods, directory structure, protection, file system structure, allocation methods, free space management, directory implementation, efficiency and performance.
- Fixed and variable record files, contiguous files. Sequential, direct and indexed access files.

## **Device management**

- Dedicated and shared devices.

- Sequential and direct storage media devices. Inter record gap, Inter-block gap, Platters, cylinders, tracks and sectors.
- Fragmentation, Addressing on a disk.
- Calculating capacity and speed.
- Blocking and buffering.
- Seek, search and transfer times.
- Seek policies: seek strategies, including FCFS, SSTF, SCAN/LOOK, C-SCAN/C-LOOK, and how they compare

### **Protection and security**

- Protection: Goals of protection, domain of protection, access matrix, implementation of access matrix;
- Security: the security problem, authentication, program threads, system threats, threat monitoring, encryption, computer security classifications.

### **Specialised operating systems**

- Distributed Operating Systems;
- Embedded Operating Systems;
- Real-time Operating Systems.