

# Operating Systems Course Outline

Rationale: Operating systems are central to computing activities. An operating system is a program that acts as an intermediary between a user of a computer and the computer hardware. Two primary aims of an operating systems are to *manage* resources (e.g. CPU time, memory) and to *control* users and software. Operating system design goals are often contradictory and vary depending of user, software, and hardware criteria. This course describes the fundamental concepts behind operating systems, and examines the ways that design goals can be achieved.

Pre-requisite: A knowledge of C, Pascal, or equivalent will be useful.

## Course Outline:

### I Overview.

1. Background
2. Computer-system structures
3. Operating system structures

### II Process Management.

1. Processes and threads
2. Process synchronization
3. Deadlocks
4. CPU scheduling

### III Storage Management.

1. Memory management
2. Virtual memory
3. File-system interface
4. File-system implementation

### IV I/O Systems.

1. I/O
2. Secondary-storage structure

### V Distributed Systems.

1. Network and distributed system structures
2. Distributed file systems
3. Distributed coordination

## Textbooks:

Lecture notes.

Abraham Silberschatz and Peter Baer Galvin:

Operating System Concepts, Addison-Wesley, 1998 (5th ed.)

## References:

Andrew S. Tanenbaum and Albert S. Woodhull

Operating Systems: Design and Implementation, Prentice Hall, 1997 (2nd ed.)

Grading Scheme:

The final grade will be computed according to the following weight distribution:

- three exercises 30%
- an exam 70%

The exam will be closed-book.

Instructor: Dr Andrew Davison  
Department of Computer Engineering  
Prince of Songkla University  
Hat Yai, Songkhla 90112, Thailand  
dandrew@ratree.psu.ac.th