1: Introduction (1–11)

- Introduction to the course
- Aims and objectives
- Recommended books
- Early computer systems (1614 onwards...)
- The structure of modern computers
- Virtual machines

2: Building everything from bits (12–26)

- Representing data in a computer
- ...unsigned integers
- ...signed integers
- ...text
- ...floating point values
- ...compound data (objects, structures, records)
- and representing instructions too

3: Devices (27–36)

- Serial devices
- Hard drives
- Display hardware
- General techniques:
  - Busses
  - The bus hierarchy
  - Interrupts
  - DMA

4: Operating systems (37–51)

- What is an operating system?
- Historical systems
- Structuring an operating system
- Dual-mode operation and protection
- Kernels and microkernels

5: Processes (52–70)

- The process abstraction
- Scheduler design
- CPU scheduling algorithms:
  - First-come first-served
  - Shortest job first
  - Shortest remaining time first
  - Round robin
  - Static & dynamic priority

6: Memory management (71–80)

- Address binding, relocation
- Logical & physical addresses
- Static & dynamic partitioning
- Fragmentation & compaction

7: Pages and segments (81–94)

- Page table structures
- The TLB
- Multi-level page tables
- Segmentation
- Protection and sharing

8: I/O hardware (95-105)

- Identifying common functions
- Virtual devices
- Polled-mode I/O
- Interrupts and DMA revisited
- Device classes
- Buffering and blocking

9: File management (106–115)

- The directory & storage services
- What is a file?
- Meta-data
- Naming and locating files
- Access control

10: UNIX case study (116-130)

- Background to UNIX
- Structure of the UNIX system
- File systems in UNIX
- Access control

11: UNIX case study (131–143)

- UNIX processes
- System initialization
- Shells, I/O & pipes
- The buffer cache
- CPU scheduling in UNIX

12: Windows NT case study (144–156)

- Design principles and structure
- The HAL and Kernel
- CPU scheduling in Windows NT
- The object manager