CA216 Operating Systems

Darragh O'Brien
dobrien@computing.dcu.ie

Office L2.35
School of Computing
Dublin City University

February 11, 2010

Course Home Page

- http://www.computing.dcu.ie/~dobrien/ca216
- Username: ca216
- Password: ca2160910
- Announcements
- Lecture notes
- Lab exercises
- Assignments
- Results
- Further reading
- Useful links

Lectures

Two 1-hour lectures per week (weeks 1-12)

- Thursday, 1000-1100, XG.21
- Friday, 1100-1200, Q1.22

Labs

One 2-hour lab per week (weeks 2-11)

- **DME**/**EE**/**ICE**: Wednesday, 1200-1400, L1.01
- **CASE**: Wednesday, 1400-1600, L1.01

Labs

- Linked from the course home page
- Completing this “homework” will help you. . .
  - Understand lecture material
  - Complete your continuous assessment
  - Do well in the exam
- Lab time has been timetabled for you
- Use it to complete lab exercises and assignments
- For best results, work in pairs

Marking

Continuous assessment

- Worth 30% of your overall mark
- Probably 1 assignment worth 15%
- Probably 1 lab exam worth 15%
- Each will involve Java threads and/or client-server programming

Examination

- Worth 70% of your overall mark
- Usually 5 questions, answer 4

How To Pass

- To pass this module your overall mark must be a passing mark i.e. 40+
- If you fail the module you repeat each failed component
- Feel free to ask questions by email or in class

Overview

- This module investigates what an OS is, what it does and how it does it
- As we shall see, an OS basically does two things:
  1. It makes the machine easier to use
  2. It attempts to share the resources on the machine fairly between users and applications
- For CASE students this module is followed in 3rd year by CA321 *Operating System Design and Implementation* which focuses more on theoretical OS design principles

Approach

- The OS we will focus on is Linux; knowing how issues are handled in at least one OS will help you understand how they are solved in others
- All practical work will be implemented on a Linux platform and includes Java threads and, where required, some simple Unix systems programming in C
- Your favourite Linux, Java and C programming references will be useful
- There are links to online C tutorials off the course home page and the C code in the labs is heavily commented so no need to worry if C is new to you

Textbooks

Required reading

- *Operating Systems with Linux* by O'Gorman
- *Operating Systems Concepts* by Silbershatz, Galvin and Gagne

Supplementary reading

- *Operating Systems* by Nutt
- *Modern Operating Systems* by Tanenbaum
- Resources linked from the course home page

What Is An Operating System?

Some definitions

- Most fundamental piece of software running on computer
- Program that runs automatically computer is switched on
- Software that provides a user-friendly interface between application programs and the hardware: it takes the pain out of dealing directly with hardware by providing a virtual programming environment
- Software that handles resource requests from application programs and that prevents applications from clobbering each another

What Does An Operating System Do?

- Process management
- Memory management
- Input and output
- File systems
- Security
- Networking

Operating System Interfaces

- The OS sits between userland applications and hardware and thus has two interfaces: hardware and software
- Like applications make talking to the OS more user-friendly so an OS makes the hardware more user-friendly
- Applications talk to the OS by invoking the system services it provides; this is “systems programming” i.e. writing code that makes system calls (using the software interface)
- The hardware interface is where the OS meets the hardware and carries out I/O on our behalf; we shall briefly look at both of these interfaces during the course

Why Study Operating Systems?

- There is a minimum amount of knowledge of how an OS works that is required by anyone seriously involved the study of computer science
- Knowledge gained is useful for guiding OS selection for a particular environment and for tuning OS performance

Why Study Operating Systems?

- When solving problems it is useful to be aware of services made available to you by the OS: they may help you in solving your problem
- An OS is a very large piece of software which solves many problems: you may come across similar problems later in your programming career and so will become a better programmer by studying operating systems