CS 390 Unix Programming Environment

Summer 2000
Course Details

<table>
<thead>
<tr>
<th>Instructors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Suchindra Rengan – ‘sachin’ (Section 001)</td>
</tr>
<tr>
<td>Sunil Keyal (Section 501)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Course Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Meetings:</td>
</tr>
<tr>
<td>Tue. &amp; Thu. 12:30 – 1:50 pm, Matheson 306 (Section 001)</td>
</tr>
<tr>
<td>Tue. &amp; Thu. 5:00 – 6:20 pm, Matheson 306 (Section 501)</td>
</tr>
</tbody>
</table>

We will be having labs in Korman 111F.

<table>
<thead>
<tr>
<th>Online Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.mcs.drexel.edu/classes/CS/mcs390">http://www.mcs.drexel.edu/classes/CS/mcs390</a></td>
</tr>
<tr>
<td>Please check the site frequently to get update information. The site will be the primary mechanism of communicating: Announcements, Class Notes, Projects, Labs etc.</td>
</tr>
</tbody>
</table>

Suchindra Rengan - CS390
Course Pre-requisites

- Must have taken MCS171, MCS172 and MCS260
- Strong background of C++ is required.
- If its been long time since you have touched C++, please do brush up on the fundamentals and the Object-Oriented concepts.
Course Details contd...

Objectives

- To effectively use the Unix operating system
- Building software under Unix operating system
- Understanding of Shell & Perl Scripting
- Learn pattern matching and regular expressions
- To develop a client-server application using Java on the Unix Operating System

Text Books

- David Flanagan: *Java in a Nutshell : A Desktop Quick Reference (Java Series)*, 1999
Course Details contd...

- Unix Fundamentals
- Writing, building and debugging code
- Pattern Matching and Regular Expressions
- Shell Scripting
- Java and Socket programming
- Perl/ CGI Scripting
- Memory Management and Process scheduling
- Basic Unix Administration
- Distributed Computing Fundamentals
Course Details contd...

- **Grading Policy**
  - Labs & Assignments – 30 %
  - Mid- Term – 30 %
  - Term Project – 35 %
  - Quizzes, Attendance & Participation – 5 %

Submissions after the late due date will result in NO Grade.

No Make up Quizzes
Lecture 1 topics

- Introduction
- Unix Fundamentals
  - Operating System and Process
  - Unix as an operating system
  - Features
  - Structure
  - File System
Introduction

- Computer hardware requires software to perform useful operations

Types of Software:
  - **System Software**: manages the operation of a computer itself
  - **Application Software**: helps user in specific tasks

System Software that controls and manages the operation of a computer is called an “Operating System”

An Operating system also manages and controls the resources of the computer

Note that **OS** will now on be used in place of Operating System

UNIX is an OS
Operating System

- Operating System is the most fundamental of all system programs
  - OS controls **ALL** of the computer resources
  - OS provides **VALUABLE** services to user programs
  - OS **COORDINATES** the execution of user programs
  - OS **PROVIDES** resources to user programs
- The notion of *Process* is very important when we talk about OS
Process

What is a process?

- A program in execution
- OS controls the execution of a process
- A process can be initiated by an user or the OS
- User initiated process is called as “User Process”
- System initiated process is called as “System Process” or “Kernel Process”
Why UNIX?

- During past 25 years UNIX OS has evolved into a powerful, flexible and versatile OS.
- It works on Multi-Platform
  - PC’s, Micro-computers etc.
- Open Source code standards
  - Ability to adapt the UNIX system in different ways
  - Linux has been developed using these standards
- UNIX OS has many features
Unix Features

- Unix as an operating system has following features:
  - Multitasking
  - Support Multi-user environment
  - Portability
  - Provides Tools
  - Communication and Networking
  - Manages Application Software
Multitasking capability

- Ability to perform *more than one task*
- Increases the efficiency of the processor
- Speeds up work
- E.g. when you are printing a file, while it is printing, you start editing another document
Multi- user support

- Support multiple users to work at the same time
- Saves time by enabling multiple users to work on same set of information at the same time
- Maintains data consistency. **Why?**
- Increases productivity
Portability

- Easy to modify Unix system code for transporting from one system to another
- Saves time and money
- Retraining is avoided
- Programs written in one Unix system can be run on any other system supporting Unix
Unix system–supplied tools

- Tools help in performing variety of specific jobs
- Unix provides flexibility to add tools provided by various vendors
- E.g. editors, compilers, calendar, mail system etc...
Communication and Networking

- Offers programs and utilities that provide services needed to build networked application
- Communication helps in information sharing and processing across a network
- Internet services use Unix as a Web Server
Other Features

- Open source code is adopted as a methodology for developing Unix e.g. Linux
- Unix is available FREE on the Internet
- A standard for distributed computing
  - Special topic presentation
- Lots of Unix flavors: Minix, Linux, Sun O/S, Sun Solaris, HP/UX etc...
Unix Environment

- User
- Application and Tools
- Shell
- Kernel
- UNIX OS
- H/W - Hardware

Suchindra Rengan - CS390
A User

- Types of users
  - Administrator: administers the system
  - General: uses the system
- Can have different privileges
- A user has a login name and password to log into the system
- Considered as an external entity to the system
- Unix Supports multiple execution modes:
  - Kernel / Superuser Mode
  - User Mode
Execution Modes

- Operating System runs in the Kernel mode and user programs run in user mode
- Kernel mode
  - Full access to system resources
  - Direct access to hardware, memory and device drivers
- User mode
  - Access is limited
  - No direct access to hardware, memory and I/O devices
  - Typically consists of only computational instructions
Tools and Applications

- Application programs carry out different types of tasks
- Types of Application programs
  - Horizontal Application: Used in various fields like government, industry and education. E.g. a Word Processor
  - Vertical Application: Industry-specific. E.g. Banking Software
Tools and Applications contd...

- Utilities that are executed as commands
- Performs specific tasks
- Generally supplied with the operating system. E.g. text editing, processing etc.
Shell

- Interface between the user and kernel
- Enables user to execute commands / program
- Besides being a command interpreter, the shell is also a programming language

Types of Shells:
- C Shell (CSH)
- Bourne Shell (bash)
- Korn Shell (KSH) etc.

Supplied by various vendors
Kernel can be considered as the heart of O/S.
Performs input/output services
Handle errors
Control access to the system
Maintain file system
Managing memory

Functions of Kernel include:
complexity

- Insulates user level processes from the hardware
- Through device drivers, system calls
- Interacts directly with hardware of computer
- Interface between the Shell and the Hardware
Unix File System

- Hierarchical file system
- Inverted tree structure
Additional Reading Assgn.

- Read on *Unix Background* and *history*
- Check to see if your accounts on the dunx1.irt.drexel.edu and queen.mcs.drexel.edu are activated
- On Thursday 06/22/2000, we will meet in Korman 111F.
- Please do get a print out of Lecture 2 Notes