CSC209: Software Tools and Systems Programming

http://www.cdf.toronto.edu/~csc209h/winter/

Michelle Craig
mcraight@cs.toronto.edu
BA 4258

David Liu
david@cs.toronto.edu
BA 4260
Communication

• **Piazza**
  – Use first for non-personal communication
  – Informative subject lines help
• **Email:** mcraig@cs.toronto.edu
  – Subject should include 209
  – Email is a formal method of communication:
    • Use proper English.
    • State your question clearly, with enough context.
    • **Sign it** (Name and cdf id are the most useful. Do not include your student number.)

• **Anonymous Feedback**
  – Link from the website
• **In person in Office Hours**
This term

• David Liu is teaching the 10 am section
• I will be away at a conference weeks 4 and 5
  – David will teach our class
  – Watch Piazza for extra info about office hours
Inverted/Blended Classroom

• Preparation before class (videos & exercises)
• Hands-on Activities in class
  • Need to bring computers to class (or share)
Course Information

• Check the course information sheet (handed out and on the course web page) for
  – Office hours
  – Contact information
  – Assignment schedule

• The course web page is the official source of announcements. (Piazza)
  http://www.cdf.utoronto.ca/~csc209h/winter

• Make sure you have the prerequisites!
Texts
Assignments

• A1: Shell and very basic C programming
• A2: Pointers, Memory Management (C)
• A3: Fork and pipes (C)
• A4: Processes and Communication (C)

• All code must work on the CDF servers to receive full marks.
• Code that does not compile on CDF will get 0.
• Don’t wait until the last day!
Weekly Online Exercises

• Starting Next Week
• Submitted (mostly) on PCRS
  – later in the term on MarkUs
• 1% per week
• Must submit your work to get marks
• Labs are due by 6pm Friday
• Optional Attendance in BA labs
  – Wed 8-9 pm
  – Fridays 1, 2 or 3 pm
Submitting Assignments

• You will be using SVN to manage and submit your assignments.
• The repositories will be set up this week.
• *Do not wait until the last minute to try to commit your assignment for the first time.*
Plagiarism

• “The work you submit must be your own, done without participation by others. It is an academic offence to hand in anything written by someone else without acknowledgement.”
• You are not helping your friend when you give him or her a copy of your assignment.
• You are hurting your friend when you ask him or her to give you a copy of their assignment.
What is cheating?

• Cheating is
  – copying parts or all of another student’s assignment
  – including code from books, web sites, other courses without attribution
  – getting someone else to do substantial parts of your assignment
  – giving someone else your solution

• Cheating is not
  – helping to find a bug in a friend’s code (be careful)
  – helping each other understand man pages or example code.
What is this Course About?

• Software Tools
  – Efficiently use the Unix Command Line
  – Understand the shell
  – Use Basic Shell Programming
  – Understand and Use Make

• Systems Programming
  – C
  – files
  – processes
  – communication
Self Study Topics

• Using SVN
• Using Unix - some initial guidance
• Learning an editor – vi, emacs, scite, nedit, …
• Learning a debugger – ddd, gdb, eclipse
Demographics Survey
Windows users

• If you want to do some of your work locally on your own machine, you will need to install cygwin.
• Check out cdf site for help

• For general help (not just Windows users):
  – Take advantage of the help centre!
Why Unix?

- Available on a number of platforms.
- Multi-user, multi-programmed.
- Shares computer resources sensibly.
- Permits manipulation of files, processes, and programs.
- Allows inter-process and inter-machine communication.
- Permits access to its operating features.
The Unix Philosophy

• Write programs that do one thing and do it well.
• Write programs to work together.
• Write programs that handle text streams, because that is a universal interface.
Unix Principles

• **Do one basic thing well**
  – with some basic variations

• **Simple input formats**
  – plain text
  – don’t require interactive input
    • stdin to stdout/stderr

• **Simple output format**
  – expected to be input to another tool
Unix Tools Example

• sort
• sed
• standard input/ standard output
• pipes
Unix *is* user-friendly; it’s just choosy about who its friends are.
Basic Tools to Learn

- head, tail
- cd
- mkdir
- ls
- cp
- mv
- rm
- diff
- comm
- cut
- cat
- wc
- grep
- who
Unix manual

- man example
- don’t memorize - look it up!
- it grows on you
For Live Work in Class

• work on CDF

• [www.cdf.toronto.edu](http://www.cdf.toronto.edu) for help pages