Administrivia

Slide 1

- One purpose of the syllabus is to spell out policies, especially about:

  - **–** Course requirements and grading.

  - **–** Exam dates.

  - **–** Late work.

  - **–** Academic integrity.

- Most other information will be on the Web, either on my home page (here, office hours) or the course Web page (here).

  A request: If you spot something wrong with course material on the Web, please let me know!

More Administrivia

Slide 2

- Part of my job is to answer your questions outside class, so if you need help, please ask! in person or by e-mail or phone.

- Some of my office hours are designated as "open lab". At those times I will be in one of the classrooms/labs ready to answer questions.

More Administrivia

Slide 3

- Probably the easiest (though not the only) option for doing the assignments is to use the Linux lab machines.

- You should have physical access (via your TigerCard) to four rooms containing such machines any time the building is open. You should have remote access to any that are booted into Linux.

- Returning students should already have accounts set up. (If you've forgotten your password, go to the ITS help desk and ask for it to be reset.) Accounts have been set up for new students. Username is the same as your Windows/ITS username; password has been mailed to you. To change it, open a terminal window and type `yppasswd`.

What Is This Course About?

Slide 4

- Back story: The primary goal of our traditional first course is to introduce students to programming and algorithmic problem-solving. Another goal of the course, however, is to expose students to certain low-level concepts that contribute to a well-rounded education in computer science. Students who come into the major via other routes may not get this exposure, and they are apt to struggle in later courses.

- CSCI 1120 is a new course intended to cover only the parts of CSCI 1320 that might not be covered by alternative introductory courses.

Course Topics

- Basic C programming, for people who already know how to write programs in some other language (such as Java).

- The Linux/UNIX command-line environment and command-line development tools.

Slide 5

- Basics of computer arithmetic.

- More advanced topis as time permits.

Getting Started with Linux

- (A UNIX person's response to claims that UNIX isn't user friendly: "Sure it is. It's just choosy about its friends.")

- When you log in, you should get a graphical desktop, which should be navigable with what you know from using other graphical environments (though some details are different).

Slide 6

- In Linux, we talk about files and directories; the idea is the same as Windows' files and folders, though again some details are different.

- The graphical system should give you a way to get a terminal window. Once you have that ... (to be continued next time).

Minute Essay

Slide 7

- Tell me about your background:

  What programming classes have you taken (high school or other), and what language(s) did you use?

  Have you had any exposure to a Linux/UNIX command-line interface? Linux/Unix command-line interface?

- What are your goals for this course? Anything else you want to tell me?