Slide 1

Administrivia

- One purpose of the syllabus is to spell out policies (next slides).
- Most information will be on the Web, on either my home page (office hours) or the course Web site (next slide).

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

Slide 2

Course Web Site

- "Course Web site" is meant to point you to pretty much all information for the course — readings, assignments, etc.
- You can find it via TLearn, or via the link from my home page (findable from the department's Web site, `http://www.cs.trinity.edu` or (probably?) by doing a Web search on my name).

Slide 3

Course FAQ

- "What will my grade be based on?" (See syllabus.)
- "What happens if I can't turn in work on time, or I miss a class?" (See syllabus.)
- "What's your policy on collaboration?" (See syllabus.)

Slide 4

Course FAQ, Continued

- "When is the next homework due?" (See "Lecture topics and assignments" page. Homeworks will be assigned at least a week before their due date.)
- "Do I have to use the lab computers for programming assignments?" (No, but that may be the easiest way to make sure they meet my criteria for full credit — I will test on one of these machines.)
- "When are your office hours?" (See my home page.)

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

Slide 5

A Little About Me

- Short version of biography: Undergrad degrees from UT Austin, math and Plan II. More than ten years in what we now call IT (about half doing operating-system-related work on IBM mainframes). Back to school for master's and PhD in computer science. Two years as a postdoc, then at Trinity since Fall 1999.
- I teach a variety of courses, but currently focusing more on courses "close to the machine". My research area (sadly neglected for some years) is parallel computing.
- (What do I do for fun? well . . .)

Slide 6

Why Take This Course?

- (It was required in a previous departmental curriculum. Now it's not, but maybe it should be?)
- "ACM says so" (i.e., curriculum recommendations include course on operating systems). Why? Well . . .
- To be a "computer scientist", need to have a broad understanding of computer systems — and operating system is a key part of a computer system.
- Knowing something about how operating systems work helps you write efficient code.
- Many of our courses "demystify" parts of computer systems (e.g., CS1/CS2 and Computer Design); so does this course.
- It might even be interesting . . . (I hope so!)

Slide 7

What Is An Operating System?

- Definition by example?
- Definition from operating systems textbook?

Slide 8

What Is An Operating System? Continued

- Definition by example:
  - Recent: Windows, Linux, UNIX, OS X (Mac), iOS, ChromeOS, Android . . .
  - Older: MULTICS, VMS, MVS, VM/370, . . . (In the mainframe days, typically each hardware company had its own operating system(s).)
  - Special-purpose O/S's for special-purpose hardware.
- Definition from textbook:
  - Something that provides "virtual machine" for application programs and users ("top down").
  - Something that manages computer's resources ("bottom up").
- Another view — key part of bridging gap between what hardware can do (not much, but very fast) and what users want.

Slide 9

Course Overview

- Brief history of operating systems.
- Review of what hardware can do, what operating system must/should do.
- Discussion of major functions of operating system — problem(s) to be solved, solutions:
  - Process management.
  - Memory management.
  - I/O management.
  - Filesystem management.

  Focus on principles rather than details.

Slide 10

Minute Essay

- (Most lectures will end with a "minute essay" — as a quick check on your understanding, a way for me to get some information, etc., and also to track attendance. Send me your answer by e-mail (no word-processor attachments please), and *please* put "minute essay" and the course in the Subject line.)
- What are your goals for this course?
- What operating systems have you used/installed/experienced?
- What assembly languages do you know well enough to read, and to write a little?
- Anything else you want to tell me? about the course, what you did this summer, . . . ?