Slide 1

Administrivia

- Reminder: Midterm Wednesday.
- Reminder: Homework 3 programming problem due today. If you can't finish by then, turn in what you have, tell me it's preliminary, and submit a finished version later.
- I'm almost done grading the Homework 3 problems turned in last week. I plan to return them in the mailbox outside my office later today. I will try to also grade today anything turned in by 5pm.

Slide 2

Administrivia

- Sample solutions to written problems distributed in hard copy as I get them graded. I've also put copies on Google Drive.
- Sample solutions to programming problems available via course Web site, at the bottom of the "lecture topics etc." page. (The one for Homework 3 will go up probably tomorrow.)

Slide 3

Minute Essay From Last Lecture

- No clear consensus on what went wrong with the Homework 2 written problems, but a few people mentioned that doing the programming problems helped them understand the concepts better.
- Much interest in an extra-credit assignment. I will put one together, as soon as I can. (It's taking more time than I thought to come up with good problems.)

Slide 4

Homework 2 Essays

- Many people mentioned difficulties with the written problems (one person's description was "super darn hard" — not my intent but maybe the effect?). Problems 3 and 5 were mentioned more than others.
- The programming problems seemed to give less trouble in general, though surprisingly few people realized that it really does matter whether you declare variables `volatile`. Several people said doing these problems helped them understand concepts (good!). Several also said the problems weren't as hard as they initially though they might be (also good, and my intent!).

Slide 5

Exam Review

- History of O/S's — nothing very detailed, maybe something about how they evolved, goals of different types.
- Functions to provide — resource manager, virtual machine (key abstractions).
- What's needed from hardware — dual-mode operation, memory protection.
- System calls — what they are, when used and why.

Slide 6

Exam Review, Continued — Processes

- Process abstraction (meaning processes or threads) — "virtual CPU" (registers, PC, PSW, etc.).

  Process abstraction ("heavyweight" processes, not threads) — "address space", maybe information about open files, etc.

  Table on p. 95 of text seems helpful in making this distinction.
- Context switches — what they are, how they work.
- CPU scheduling — how well (or not well) different algorithms work for different systems (batch, interactive, etc.).

Slide 7

Exam Review, Continued — IPC

- Notion of “atomicity”.
- Synchronization mechanisms — shared variables only, shared variables with hardware assist (e.g., TSL), semaphores, monitors.
- Classical problems — mutual exclusion, bounded buffer, dining philosophers. For mutual exclusion, idea of “critical region” and how that might apply to instances of this problem.

Slide 8

Exam Review, Continued — This and That

- Deadlocks — what they are, when they can arise.
- In general, if it’s in the textbook but wasn’t mentioned in class or the homeworks. I probably won’t ask you about it.

Slide 9

Minute Essay

- None really — just tell me you were here, unless you have questions or concerns about the exam?