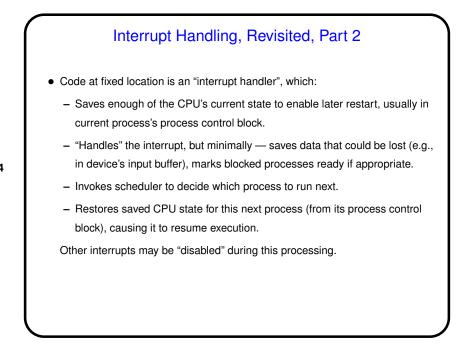
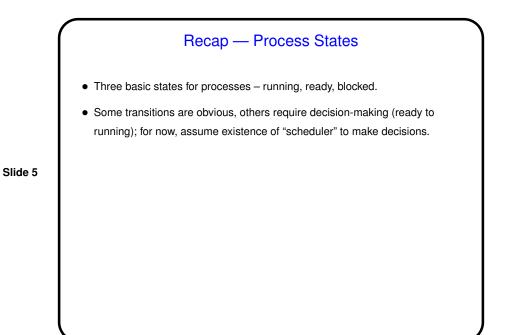
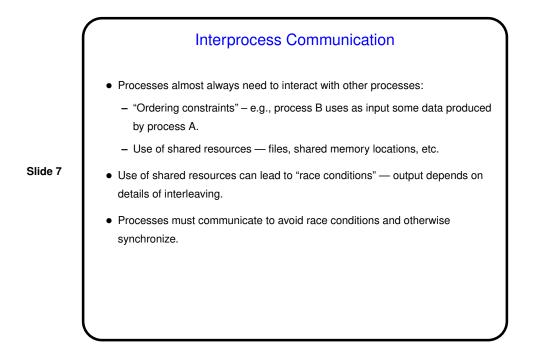


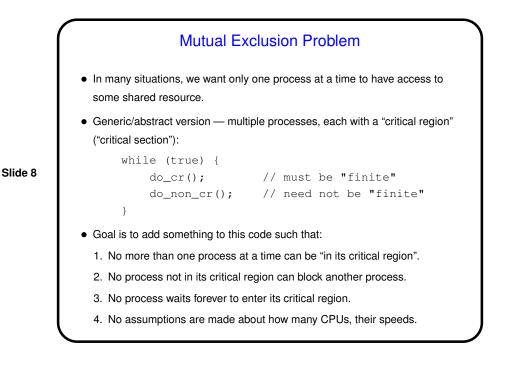
Slide 3

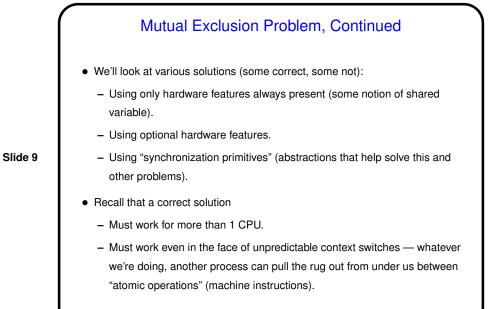


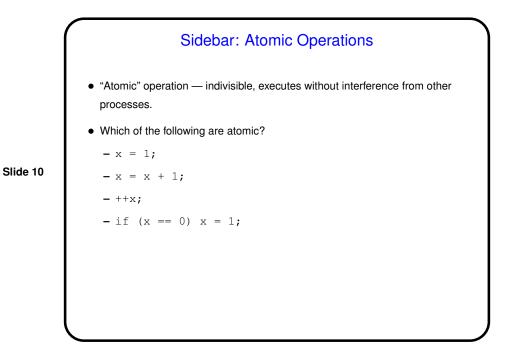


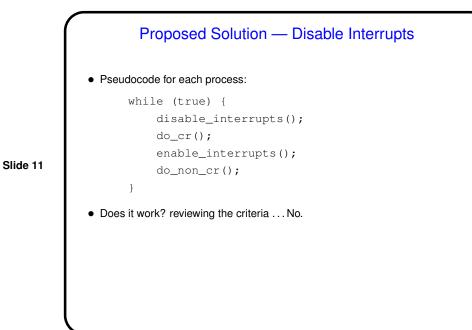
Processes versus threads:
Process implements "program on virtual CPU" abstraction, has its own group of resources.
Thread implements "program on virtual CPU" abstraction, shares group of resources with (some) other threads.
Threads are in a way "processes within processes".
Compare context switching between processes with context switching between threads within process.
Two basic approaches to implementing threads — "in user space" and "in kernel space".











Proposed Solution — Simple Lock Variable

• Shared variables:
 int lock = 0;
Pseudocode for each process:
 while (true) {
 while (lock != 0);
 lock = 1;
 do_cr();
 lock = 0;
 do_non_cr();
 }
• Does it work? reviewing the criteria ... No.

