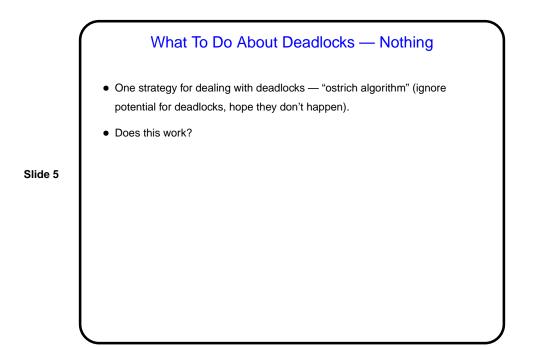
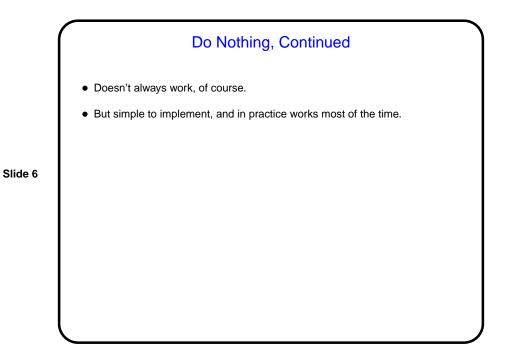


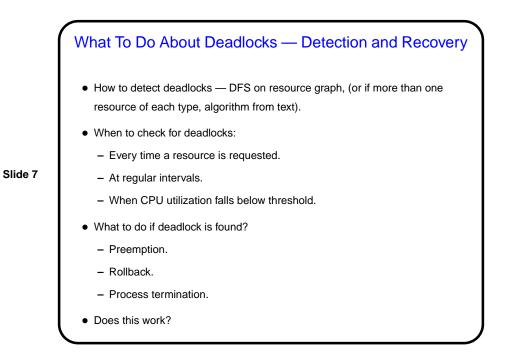
	Resources
	<ul> <li>"Resource" is anything that should be used by only one process at a time — hardware device, piece of information (e.g., database record), etc.</li> <li>Can be unique (e.g, particular database record) or non-unique (e.g., one block of a fixed-size disk area such as swap space).</li> </ul>
Slide 3	<ul> <li>Preemptible versus non-preemptible — preemptible resources can be taken away from current owner without causing something to fail (e.g., memory); non-preemptible resources can't (e.g., hardware device).</li> </ul>
	<ul> <li>Normal sequence for using a resource — request it, use it, release it. If not available when requested, block or busy-wait.</li> <li>Can easily implement this using semaphores, but then deadlock is possible if</li> </ul>
	processes aren't disciplined.

## Deadlocks — Definitions and Conditions

- Definition set of processes is "deadlocked" if each process in set is waiting for an event that only another process in set can cause.
- Necessary conditions:
  - Mutual exclusion resources can be used by at most one process at a time.
  - Hold and wait process holding one resource can request another.
  - No preemption resources cannot be taken away but must be released.
  - Circular wait circular chain of processes exists in which each process is waiting for resource held by next.
- Modeling deadlock "resource graphs".
- What do about them? Various approaches.



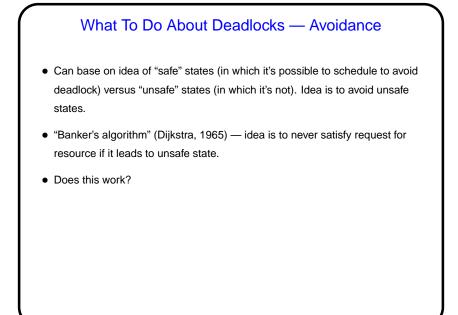




## Detection and Recovery, Continued

• Does work.

• But potentially time-consuming, and "what to do" choices aren't very attractive!



Does work.
But not much used because it assumes a fixed number of processes, resource requirements known in advance.

Slide 9

