Administrivia

- Correction to last time's notes — I said "RPC (remote procedure call)", should have said "RMI (remote method invocation)".

- OpenMP homework coming soon. Don't try installation right now — script is broken. (I was sure I had tested it, but apparently not!)

Slide 1

Creating Threads in Java, Revisited

- Java 1.5 has many new features for working with threads, including interfaces that make it easier to create threads and assign them work: `Hello3.java` and `Hello4.java` on sample programs page.

Slide 2

Synchronization in Java, Continued

- Recall — `synchronized` methods/blocks can be used to ensure that only one thread at a time accesses some shared variable.

- For more complex synchronization problems, can use `wait` and `notify` (or `notifyAll`):

  `wait` suspends executing thread (adds to "wait set").

  `notify` wakes up one thread from the wait set. `notifyAll` wakes up all threads. Waked-up thread(s) then compete to reacquire lock and continue execution.

  Can only be done from within synchronized method/block.

  Typical idiom — loop to check condition, `wait`.

- Example — bounded buffer class (`BoundedBuffer.java` on sample programs page).

Slide 3

Controlling Threads in Java

- Preferred method of controlling one thread from another uses "interrupted" status. (Early version of Java provided other methods, e.g., `stop` — now deprecated.

- Set status with `interrupt` (instance method).

- Check status with `isInterrupted` (instance method) or `interrupted` (static method), or by catching `InterruptedException` thrown by `wait`, `sleep`, `join`, etc.

- Example — bounded buffer test program (`TestBoundedBuffer.java` on sample programs page).

Slide 4

Minute Essay

- Have you downloaded a copy of the Intel compiler? Tried to install?