1. Explain interrupts to me. Include quick explanations of what they are, how they're handled, examples of what they can be used for (and why they're useful), and how IRQ numbers are handled (including when used with multiple PICs).

2. Explain user mode and kernel mode for operations. How can user applications make use of code that requires kernel mode?

3. Explain the components within the PCB.

4. Tell me some differences between a thread and a process.

5. Why are thread control blocks necessary as separate entities from the PCB?

6. If a ULT thread blocks for a system IO event, what is the state of that thread? What is the state of the process?

7. Why is thread synchronization important? Give an example.

8. Describe what would happen if a process exited while a thread of that process is currently running.

9. What's the principle of locality, and why does it matter?

10. Suppose I write a number-crunching program. Briefly explain two completely different ways I can improve my throughput on a multicore system.

Submission: **Type out** all of your answers and print them out. **ONLY TYPED SUBMISSIONS WILL BE ACCEPTED AND MARKED!** Attach a departmental cover page, staple everything together, and drop it into the appropriate dropbox in J-block. Envelope not required. Work alone. Don't cheat.