

CSc 360: Operating Systems (Summer 2006)

Self-test Questions 2

This list, by no means thorough, is compiled from “This lecture” on the lecture slides.

1. How do `fork()` and `exec()` work? Why `wait()`?
2. Compare the ways to create, synchronize and terminate processes and threads.
3. Compare the ways to communicate between processes and between threads.
4. What information do PCB and TCB keep? How to schedule processes or threads? On uniprocessor or multiprocessor?
5. How to use shared memory or message passing to solve the producer-consumer problem with processes?
6. How to use mutex and/or condition variables to solve the producer-consumer problem with threads?
7. How does RPC work? Why marshall parameters?
8. Compare user and kernel threads and the mappings between them.
9. The goals and criteria of scheduling.
10. Preemptive and non-preemptive scheduling algorithms.