CS 677: Distributed and Operating Systems
Homework 3

Due one week from posting date as listed on the course web page

1. When might a process pool model be better than using a thread pool model?
2. Why is it important to respect cache affinities of processes and threads in multiprocessor scheduling?
3. Since a static thread pool will cause requests to block when all threads are busy processing requests, why should the server not create a static thread pool with a very large number of threads? (Explain the disadvantage of such an approach).
4. For a multiprocess scheduler with distributed queue (one queue per core), why can imbalance occur even if you start with perfectly load balanced queues, each with equal number of processes?
5. Explain why distributed scheduling in a network of workstations is not effective at very low loads and at very high loads.
6. Why do cluster scheduling of batch jobs support separate queues per user group? (Explain an advantage of such an approach)
7. Why is emulation slower than native hardware virtualization?
8. How does a type 2 hypervisor handle sensitive instructions inside a virtual machine? Also discuss this approach does not require any special hardware support from the processor.