Jeremy Holloway
CPSC-3220-001
5/22/2019

Pre-class Assignment #6

1. Explain the difference between mechanism and policy.
   A policy is a method for decision making while mechanisms are the implementations of performing those decisions.

2. Name at least three events that can trigger a thread context switch.
   1. Thread_yield()
   2. Thread_join()
   3. Interrupt
   4. Exception

3. Why would you disable interrupts while a thread context switch occurs?
   To prevent a system from making two context switches at the same time.

4. What are green threads?
   A thread system implemented entirely at user-level without any reliance on operating system kernel services, other than those designed for single threaded processes.

5. Why would you use green threads?
   If you wanted a multi-threaded process to appear as if it were a normal single-threaded process.

6. What are scheduler activations?
   A multiprocessor scheduling policy where each application is informed of how many processors it has been assigned and whenever the assignment changes.

7. Why would you use scheduler activations?
   Allows a process to run multiple threads in parallel on a multiprocessor, and they allow a process to hide I/O latency by running other threads when one thread blocks for I/O.

8. What are the three ways listed in the textbook in which an operating system can return results after a call to asynchronous I/O?
   1. calling a signal handler
   2. placing the result in a queue in the processes memory
   3. storing the result in kernel memory until the process makes another system call to retrieve it

9. Give at least one argument in favor of event-driven programming as compared to threads.
   The space for and context switch overheads of this approach would be lower, or some past operating systems had inefficient or unscalable implementations of their thread systems.

10. Give at least one argument in favor of threads as compared to event-driven programming.
    An event driven approach doesn’t exploit multiple processors