CSE 542 Home work assignment 1

Assigned: Thurs, Sep 16

Due: Thurs, Sep 30, 11:00AM

Late submissions will not be accepted

Individual effort

At this point in the semester, you are expected to know (at least) about the following terms. Make sure that you fully understand the nuances of each term/technique. If any of these terms are unclear, talk to the instructor or the TA immediately.
Interrupt, system call, synchronous I/O, asynchronous I/O, DMA, cache, supervisor mode/user mode, monolithic/microkernel, process, threads, PCB, CPU-bound, IO-bound, context switch, message passing/shared memory, preemptive/cooperative scheduling, utilization, throughput, turnaround time, waiting time, response time, FCFS, SJF, Round robin, convoy effect, realtime, priority inversion, priority-inheritance, race condition, critical section, testandset, atomic operation, semaphore, counting semaphore, binary semaphore, spinlock, deadlock, livelock, starvation, bounded buffer problem, reader-writer problem, dining philosopher problem, monitor, serializability, deadlock conditions (mutual exclusion, hold and wait, no preemption, circular wait), resource allocation graph, bankers algorithm, deadlock detection and avoidance.

1. Precisely describe the sequence of operations required for the context switch of processes, kernel threads and user level threads. Describe the components that make one technique preferable over the other (form of context switching)

2. True or False (you may justify your answer briefly)
   i) For systems with preemptive scheduling, increasing the scheduling quantum is a good way to improve system throughput

   ii) Giving higher internal priority for the CPU to processes that were recently awakened for I/O completions is likely to improve overall system throughput and response time

   iii) Multithreading always provides better performance than a single threaded solution

3. Consider a computer with five individual resources name R1 …. R5. Let five processes P1, …. P5 make requests in order, as follows:
   I. P1 requests R2
   II. P4 requests R3
   III. P3 requests R1
   IV. P2 requests R4
   V. P5 requests R5
   VI. P4 requests R2
   VII. P5 requests R3
   VIII. P3 requests R5
   IX. P1 requests R1
   X. P2 requests R2

b. Assume the resource manager uses the liberal "allocate a requested resource if it is currently free" policy. At the end of the requests, is the allocation safe or unsafe? If unsafe, is there deadlock and if so at what point did it occur and which processes did it involve?

c. Instead of a liberal policy, imagine that processes P1 through P5 make advance claims that each needs all resources. If the Banker's algorithm is applied, how could the resources be allocated at the end of the requests?

4. Briefly discuss the strengths and weaknesses of Round Robin scheduling with Shortest Job First (SJF) scheduling with respect to the usual goals of a CPU scheduler. Why do most modern CPU schedulers combine Round Robin with SJF by giving CPU priority to I/O bound jobs?

5. What is priority inversion and why is it bad? Illustrate with an example. Use your example to illustrate one technique that avoids priority inversion

6. Exercise 4.5

7. Exercise 7.21: Show that the two-phase locking protocol ensures conflict serializability

8. Exercise 8.2