CS 39 – Final exam

Name:_____

You have 3 hours for this exam. Notice that there are 100 points, so budget your time wisely! A good bit of space is provided for your answers. Write clearly and legibly. Be sure to read carefully and answer the question being asked.

1. (7 points) What is the relationship between *processes* and *threads*? Why do threads have much shorter context switching times?

2. (7 points) How can a multilevel queueing scheduler be made to avoid starvation of low-priority tasks?

3. (11 points) *Producer/consumer process pairs* are difficult for most schedulers to handle: The consumer should be scheduled only after the producer has been scheduled to generate what the consumer needs. Why do most schedulers handle these pairs badly? How can lottery scheduling allow these pairs to be scheduled well?

4. (7 points) What were the problems with *overlays* that were solved by introducing *segments* into memory management? What hardware and kernel support was needed to provide segments?

5. (7 points) How, can a kernel avoid using too much main memory to store its page table, while still providing reasonably fast page lookups in the average case? Describe the structure of the page table for your solution in detail.

6. (11 points) Why are *phase changes* so important to page replacement mechanisms? Describe why LRU handles phase changes well, and LFU does not. (Do not simply given an example to show **that** LRU handles phase changes better than LFU; describe **why**!)

7. (7 points) Why is it possible, in the *Baker's algorithm*, that two processes can select the same ticket? How does the algorithm handle this situation to ensure progress?

8. (7 points) *Deadlock* can be handled automatically in a system either by an *deadlock avoidance* policy or a *deadlock breaking policy*. Compare and contrast these two options.

9. (11 points) The semaphore *wait* operation can cause a process either *busy wait* or *sleep*. In order to reduce the processing time wasted beyond what would be optimal, when should a semaphore busy wait, and when should it sleep? Justify your answer.

10. (7 points) In scheduling disk accesses, one possible policy that can be applied is the *shortest* seek time first policy. What property of this policy makes it undesirable? Given an example of a sequence of disk requests that would make reveal this property of this disk scheduler.

11. (9 points) Can a log-structured filesystem ever be in an inconsistent state? Justify your answer.

12. (9 points) What is *false sharing*, and how does it arise in a *serially consistent distributed shared memory*? Given an example of two processes, sharing memory, that can cause this problem to occur.