Part 1: True False (Answer + Explain)

1. If there is very high CPU utilization, then it is likely that *thrashing* is occurring.

False, thrashing is when the CPU is underutilized, but pages are being swapped out of memory aggressively.

2. If a particular IO device implements a *blocking interface*, then you will need multiple threads to have concurrent operations which use that device.

True, only with non-blocking IO can you have concurrency without multiple threads.

3. Reading 100Mb in random 1kb chunks will take slightly longer than reading 100Mb in a single sequential scan.

False (ish) - it will take WAY longer, random access on disks is much, much slower than sequential access due to seek time.

4. For IO devices which receive new data very frequently, it is more efficient to interrupt the CPU than to have the CPU poll the device.

False, it is more efficient to poll, since the CPU will get overwhelmed with interrupts.

5. Nachos is an easy-to-understand, fun way to learn about Operating Systems - or - I am really going to miss Nachos when we finish the second assignment.

False - several answers were accepted.

Short answer:
6. Disk requests come into the disk driver for cylinders: 10, 22, 20, 2, 40, 6, and 38, in that order. The disk head is currently positioned over cylinder 20. A seek takes 6 milliseconds per cylinder moved. What is the sequence of reads and total seek time using each of the following algorithms?

i) (4 points) First-come, first-served:
10, 22, 20, 2, 40, 6, 38
10 + 12 + 2 + 18 + 38 + 34 +32 = 146 cylinders = 876 milliseconds.
We subtracted 1 point for missing the first reference (to 20), and 1 point for not writing out the sequence of accesses.

ii) (4 points) Shortest Seek Time First:
20, 22, 10, 6, 2, 38, 40
0 + 2 + 12 + 4 + 4 + 36 + 2 = 60 cylinders = 360 milliseconds.

iii) (4 points) SCAN (initially moving upwards):
20, 22, 38, 40, 10, 6, 2
0 + 2 + 16 + 2 + 30 + 4 + 4 = 58 cylinders = 348 milliseconds.

7. Two-level Page Tables:
i) Give a two to three sentence description of a two-level page table.

ii) Briefly (2 sentences) state one advantage AND one disadvantage of two-level page tables.

8. What is the difference between Mesa and Hoare scheduling for monitors? Include passing of locks between signaler and signalee, scheduling of CPU resources, and impact on programmer.

For Mesa scheduling, the signaler keeps the lock and CPU, while the signaled thread is simply put on the ready queue and will run at a later time. Further, a programmer with Mesa scheduled monitors must recheck the condition after being awoken from a Wait() operation [i.e. they need a while loop around the execution of Wait(). For Hoare scheduling, the signaler gives the lock and CPU to the signaled thread which begins running until it releases the lock, at which point the signaler regains the lock and CPU. A programmer with Hoare scheduled monitors does not need to recheck the condition after being awoken, since they know that the code after the Wait() is executed immediately after the Signal() [i.e. they do not need a while loop around the execution of Wait()].

What is priority donation? What sort of information must the OS track to allow it to perform priority donation?

Priority donation is the process of avoiding priority inversion by giving ("donating") priority from a high-priority blocked thread to a lower-priority thread holding a lock needed by the high-priority thread. The OS must keep trace of lock acquisition and release operations and associate them with threads in order to perform this optimization.

9. We looked at disabling CPU interrupts as a simple way to create a critical section in the kernel. Name two drawbacks of this approach. One of them should be a problem that is exacerbated by trends in modern computer hardware.

One issue is that you can't receive interrupts from devices or timers within a critical section, which may be desired. For instance, what if you accidentally have an infinite loop in the kernel critical section? A second issue is that it is difficult to disable interrupts on multiple cores.