CSCE 410/611 Operating Systems Spring 2023

Homework for Week 11

(Due Date: Check Canvas)

1. [1] Typically, at the completion of a device I/O, a single interrupt is raised and appropriately handled by the host processor. In certain settings, however, the code that is to be executed at the completion of the I/O can be broken into two separate pieces. The first piece executes immediately after the I/O completes and schedules a second piece of code (sometimes called a “Deferred Procedure Call”) to be executed at a later time. What is the purpose of using this strategy in the design of interrupt handlers?

The purpose is to do the time sensitive work that can be done quickly, immediately so the IO operation can be handled the less time sensitive work that likely requires more system resources later.

2. Device drivers for USB busses are relying, to a surprising amount, on polling (as opposed to interrupts) to interact with USB devices. Speculate (wildly if necessary) about why designers went with polling.

USB devices most commonly are going to be mouses and keyboards that require a constant stream of information, and constantly update. The other most common type is USB storage drivers and the protocol for these devices require a rapid sending of a \_CLK and CLK signal to ensure data transfer alignment from the USB to PC. This requires precise polling to align with the USB to communicate a begin transfer, end transfer, and actually transfer the information itself.

3. [1] Why is rotational latency usually not considered in disk scheduling?

Most disks do not export they’re position to the host and by the time the information reached the system the disk has likely changed position thus making trying to export the information irrelevant.

References

[1] A. Silberschatz, P. Galvin, and G. Gagne, *Applied Operating Systems Concepts*, John Wiley & Sons, Inc., New York, NY, 2000.

[2] Deitel, Deitel, and Choffnes, *Operating Systems*, Pearson / Prentice Hall, 2004. [3] A. S. Tanenbaum, *Modern Operating Systems*, Pearson / Prentice Hall, 2008. [4] L. F. Bic, A. C. Shaw, *Operating Systems Principles*, Prentice Hall 2003. [5] C. Crowley, *Operating Systems, A Design-Oriented Approach*, Irwin 1997. [6] M. Herlihy, N. Shavit, *The Art of Multiprocessor Programming*, Elsevier, 2008

1