

Homework 1

Misha Ware

1.19 – What is the purpose of interrupts? How does an interrupt differ from a trap? Can traps be generated intentionally by a user program? If so, for what purpose?

Interrupts are important as they stop what the CPU is doing and immediately transfers the execution power towards a fixed location which is usually the location from where the interrupt came from. This allows the operating system to switch what it is working on dynamically based on events that occur.

Although interrupts and traps are similar as they occur when events occur, traps are different from interrupts as they are caused by an error in software (an example in the book is a divide by 0). Yes, these traps can be intentionally being caused (since the book notes maliciousness). Traps are also caused by lack of memory so programs that are designed to crash a system could cause a trap to occur. The purpose of causing a trap would be to crash the system or prevent correct operation of many other processes in the computer.

1.23 – Consider an SMP system similar to the one shown in Figure 1.6. Illustrate with an example how data residing in memory could in fact have a different value in each of the local caches.

An example is when CPU core₀ reads a number, say 72, and saves it into its local cache. Then CPU core₁ could read the same number and save it into its local cache. If Processor core₀ changed that value to 50, then the value would be different than what memory might have since the calculation occurs only in CPU core₀ processor.

Choose one System from the slides in Lecture One (Eniac, System 360, Multics, Android, etc..) and describe in detail (one paragraph) its importance to the evolution of Operating Systems

ENIAC was a very influential computer as it was one of the earliest electronic general-purpose computer. Although the problems that it worked on included artillery calculations, it saved hundreds of man hours. ENIAC was a huge advancement not only because of being electronic but also due to the fact that it could be programmable. Developers could start to write different programs that the computer would be able to calculate unlike prior computers which had set functions when they started up. Operations that the programs could run included loops, branches, and subroutines. Although the computer lacked stored programs like modern computers, the ability to program the computer was a huge advancement at the time even if it was a cumbersome process. Despite not having an operating system, future computers would build upon ENIAC as electrical programmable computers would start to take off.