Chris Banci
CS433
Prof. Zhang
February 9, 2017

Homework 1

1.13) The issue of resource utilization shows up in different forms in different types of operating systems. List what resources must be managed carefully in the following settings:

a. Mainframe or minicomputer systems

- memory, cpu, storage, i/o operations, network

b. Workstations connected to servers

- memory, storage, cpu, network

c. Mobile computers

- cpu, memory, power consumption, storage.

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?

- An interrupt is a signal to the CPU to take note of a flow of change within the system and decide what action to take next. An Interrupt is hardware-generated from an I/O device, while a trap or exception is a software-generated interrupt. Traps can be generated intentionally by a user program to make system calls by switching to kernel mode.

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.

- Because an SMP system uses multiple processors, it is possible for CPU's to hold different values. For example, CPU0 reads data with a value of 3 from the main memory and stores it to its own cache. CPU1 reads the same data from main memory and stores it to its own cache. Now CPU0 then updates the data in its cache to 6. Since CPU0 and CPU1 have separate caches, CPU1 did does not hold the same value. CPU0 = 6, while CPU1 = 3.

1.25) Describe a mechanism for enforcing memory protection in order to prevent a program from modifying the memory associated with other programs.

- A simple mechanism to solve this issue is to keep track of the memory location of each program, then limit their access from writing outside of their given memory location. This way, the programs memory is protected from outside processes. The operating system has mechanisms such as address translation and dual mode operation to achieve this.

2.13) Describe three general methods for passing parameters to the operating system.

- 1. Pass parameters to system using registers.
- 2. Store parameters in memory, then pass memory address to system using registers.
- 3. Push parameters to the stack and let the system pop it off the stack.

2.14) Describe how you could obtain a statistical profile of the amount of time spent by a program executing different sections of its code. Discuss the importance of obtaining such a statistical profile.

- To obtain a statistical profile, one could issue timer interrupts in intervals when when executing a program. Statistical profiling takes periodic samples of a program during the execution state and this information allows a programmer to optimize the performance of a program by looking at sections of code that spends the largest portion of its cycles.