

CSCI 3753

Problem Set 1

Kyle Ward
Spring 2018, CU-Boulder

January 29, 2018

1. *What are the three high level features that are provided by an Operating System? (describe each*

Hardware Abstraction:

This part of the OS deals with I/O and other hardware needed to run a computer. It doesn't matter what CPU, keyboard, mouse, screen, printer etc. you have. The OS abstracts away all the specific information needed for each device and allows it to be run in any situation/configuration wanted.

Security:

This part of the OS will protect the OS and kernel from being overwritten or attacked. It is the only thing that has permission to directly interact with the hardware. If the OS crashes due to error or being overwritten then the whole computer will crash.

Resource Management:

This part of the OS makes everything run smoothly. It divides resources such as memory and CPU time between processes.

2. *Why does an Operating System need protection from user code?*

Being protected from users is very important for the OS. The OS is in charge of making the computer function and usable. If user code accidentally or maliciously changes the OS, Data could be lost/stolen or worse, the machine stops working correctly/at all.

3. *What are the four major components of an Operating System? (describe each)*

File System:

This manages the memory on our hard drive or other mass storage device

Memory Manager:

This manages the cache and pages tables, computing where to place/take things from with respect to the hard drive and also making sure that we have enough memory for the task needed.

Scheduler:

This manages CPU usage between different processes. It prioritizes which process gets the CPU and for how much time.

Device Manager:

The Device Manager handles all of the I/O for hardware such as the keyboard, mouse, printer, and screen.

4. *How does a user application access system code within the Operating System? (list the steps, describe how CPU knows what code is running)*

The application will make a system call. The the OS catches this in a trap. The OS switches the mode bit to 0(which is kernel mode) and the OS executes the system call. Then the OS switches the mode bit back to 1(user mode) and returns from the system call back to the application.

5. How are parameters passed to and results returned from a System Call? (describe possible mechanisms)

The application makes a system call with parameters. The OS catches it in the trap and determines which type of system call it is.

6. What are the steps taken to start up a computer?
7. What is a Virtual Machine and how does it differ from a physical machine?
8. Describe the differences between multi-programming and multi-tasking?
9. What is a context switch? (describe the mechanism, the reasons for a switch, and what is switched)
10. Provide a timing chart for the processes listed below using multi-programming and another when using multi-tasking. The values are all in number of ticks and you can assume no switching overhead. The multi-tasking timeline should assume a 40 tick time block. The IO column indicates when within the process an IO request is made and how long the wait for IO completion will take.

Process	Length	Arrival Time	IO (time, wait)
P1	40	0	(20, 30)
P2	20	10	none
P3	25	30	none

