Task1

1, The functions of OS are making it more efficient for user to use the computer. The OS can help manage the memory, I/O, it is more convenience for user then the efficient use of OS.

2, The concept of Abstraction is hiding the complex of computer architecture and provide the friendly interface for user. The virtualization in system means the OS illusion the access the resource and hardware but we don’t have exclusive access. Example:

Process: The CPU called by system. The container reference the all code needed to run a program.

Address: the address is the number correspond with the memory space allocated of the program.

Task 2

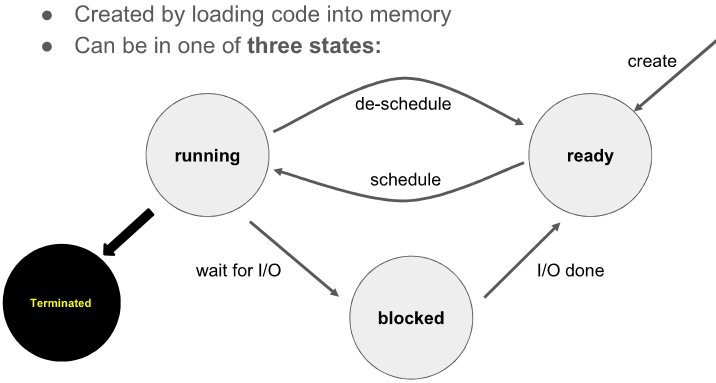
1, A program is a group of instructions to carry out specific tasks it resides in secondary memory (hard drive) whereas process is an instance of executing a program and is loaded into the main memory.

2, The context switch can restored the current process, the process state consists of the contents of all registers that can be accessed through the code. A context switch also sets the CPU mode, switching from user mode to kernel mode or back.

3, Ready state – runnable, but temporarily stopped to allow other process to run on the CPU, it is the first state after loading the code from memory in the OS.

Running state - using CPU at that time and running when the OS schedule the program and switches to it, there are only one process in running state.

Blocked state - unable to run until system call to waiting the I/O, the OS will schedule other ready processes in this time, after I/O read, the blocked state turn to ready state.



Task 3

1, Kernel Mode (High priority) - Code without any restrictions

User Mode (Low priority) - Limited subsets of instructions are allowed.

The user mode involves all the user applications/processes and also it is restricted from accessing the hardware directly, whereas kernel mode has direct access to hardware and maintains control over all resources and the system itself, the OS runs in kernel mode.

2, System call is call from app (from user mode) into the operating system (running in kernel mode), and app cannot execute CPU instructions to access the hardware.

3, Cooperative and Pre-emptive

Cooperative-(non-preemptive) is switch into kernel when user code makes a system call or while the hardware interrupts the process, in the cooperative scheduling, the process cannot be scheduled.

Pre-emptive: wait for a timer interrupt Pre-emptive scheduling can be Pre-emptive; the processes can eb scheduled.

The operating system will initiate a context switch from a running process to another process.

4， time reader

>>C:\Users\aud>time

The current time is: 18:47:39.51

Enter the new time:

Open file

>> C:\Users\aud>cd Desktop

C:\Users\aud\Desktop>dir FIT1047

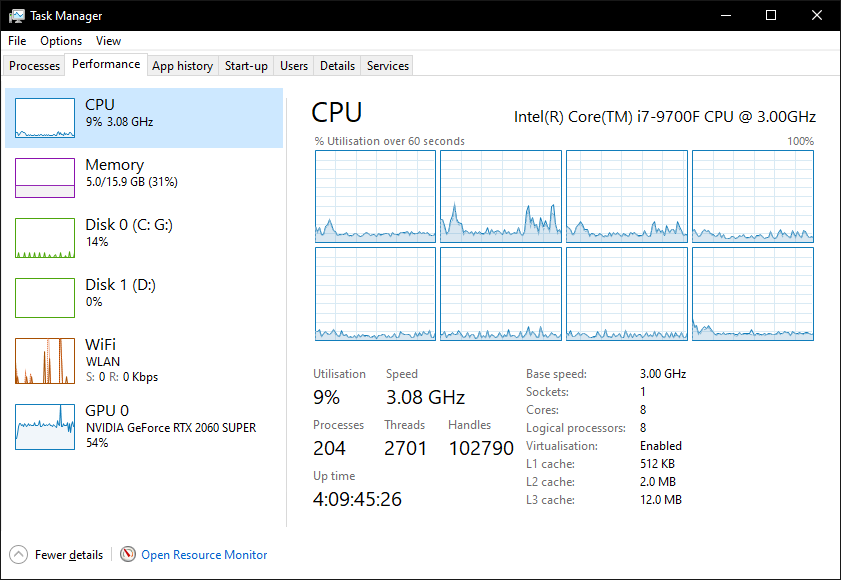
Create file

>> C:\Users\aud\Desktop\FIT1047>mkdir FITLab7

Delete

>> C:\Users\aud\Desktop\FIT1047>rmdir FITLab7

Task 4



1&2, 204 processes running with 8 cores

3, 8 are used, by the chrome.

4, most part of time spend in kernel mode