Name : Muhammad Ishraf Shafiq Zainuddin
ID : 200342741
Date : 16th May 2018

Notes

1. shellscript command

```
gcc test.c -o test
./test                    //Yo Earth!
```

2. (a)

```
#!/bin/bash        //Tell the computer that we want to use bash
echo “compiling…”
gcc test.c -o test    //in compile.sh
echo “done”
```

(b)

```
chmod +x compile.sh
./compile.sh
```

Operation System - A lot of Abstraction of Hardwares
Virtualization - A process of creating the desirable
GUI - Graphical User Interface
- Visual way of interacting with a computer using items used by most OS
- Allow users to interact with electronic devices

ps - kind of processes that are spawned
Processes - Representation of program (1 process, 1 program)
- You can isolate processes from each other
- Allow for multi-tasking
- Virtual memory (Pool of Memory)
- Allows for monitoring access (avoid chaos)

Processes can spawn another processes and should know about each other.

Process - Portion of a program (represented in Main Memory)

Program - Present in Disk

Layers of Abstraction

User Processes - set of command user can use, Do not have direct access to Hardware

Kernel - core of OS (the abstraction of Hardware), driver

Hardware - based layer, CPU etc….

User Space - “Browser should not have access to Kernel space”

- small sub of memory, only allows safe processes, “it’s fine if its crash”

Kernel Space - have unrestricted access to Hardware

Hardware - Main memory: Bunch of bits (CPU)

- Generally there’s no physical separation of space
- Illusion of space/process (Virtualization separation)
- Big Pool of Memory (Illusion)

Modern OS - Can use different processes at the same time