BITS Pilani - Hyderabad Campus
Operating Systems (CS F372)

Tutorial 1

The objective of this tutorial is to learn some Linux OS concepts and commands.

1. Run the command ps with various options and explain the output (important fields)

   ```
   $ ps                    ( to read manual page for any command, run man <command>
   $ ps -e
   ```

   You will see that each process has a unique id (called ***pid***). Use pipe (|) to send output to 'more' to view output one screen at a time. What are Unix pipes?

   ```
   $ ps -ef
   $ pstree
   ```

   (to see process tree. Look at the tree and find the bash process. Go up the tree by using grep to look at parents recursively. You can see that init process is at the root of this tree.)

   ```
   $ ps -ef | grep [parent pid]..
   ```

2. Run a program and use kill *pid* to terminate it. E.g. open vi in the background (tell them about background processes), and terminate it by its pid.

   ```
   $ sleep 100&            (command to sleep for 100 seconds)
   [1] 12175  shows pid
   $ kill 12175            (use kill -9 <pid> if this doesn't work)
   ```

3. Create hard link to a file. What is a hard link? Need for hard links?

   Assume a file 1.c is present with some text content; run the following commands

   ```
   $ ln 1.c 2.c
   $ ls -li
   ```

   After running the above command, you will see two files with same size and **inode** number (what is inode?)

   Now, delete 2.c, and again run "`ls -li`"

   Repeat the above process but now delete the original file instead of the new one. Nothing changes except the file name. Directories have at least 2 links. Why?

4. Create symbolic link to a file. What is a symbolic link *aka* soft link *aka* sym link?

   ```
   $ ln -s 1.c 2.c
   $ ls -li
   ```

   The output is different this time, 2.c is shown as a symbolic link to 1.c (look at the first letter in the permission field, and also look at the end of the output line corresponding to 2.c). The inode is also different. Attempting to read content of 2.c will result in content of 1.c being read instead. The symbolic link file just contains the name of the file it is pointing to; you can see this by running:

```
$ readlink 2.c
```

it will show 1.c as its content

Copying the symbolic link to another file copies the content of the original file the symbolic link is pointing to.

```
$ cp 2.c 3.c
```

`$ ls -li` to see the inodes

Now delete the symbolic link.

```
$ rm 2.c
$ ls -li
```

Now delete all **.c** files (use ***rm*** <***filename***>)

Repeat the above process of creating symbolic link, but now delete the original file instead of the symbolic link.

```
$ ln -s 1.c 2.c
$ ls -li
$ rm 1.c
```

The symbolic link remains but the file it points to has become invalid. Try copying 2.c to another file….will result in an error.

```
$ cp 2.c 3.c
cp: cannot stat ‘2.c’: No such file or directory
```

Why are symbolic links needed?

5. What is sudo? Use sudo for running privileged commands e.g .

   ```
   $ sudo apt-get install vim
   ```

   [You may need to enter password]

   It will download and install vim from the repository on the Internet. To uninstall, run

   ```
   $ sudo apt-get remove vim
   ```

6. Boot process: Linux booting comprises the following steps:
   - BIOS (from ROM)
   - Boot Loader
     - MBR (master boot record – in the first sector, used to load GRUB)
     - GRUB (Grand Unified Boot Loader)
   - Kernel
   - Init
   - Runlevel scripts

Read more at : https://en.wikipedia.org/wiki/Linux_startup_process

Tutorial 2

The objective of this tutorial is to learn the concepts related to booting an OS.

1. GRUB (Grand Unified Boot Loader):

GRUB is the boot loader that Linux systems use to boot the operating system of your choice.

By default, in a single boot system the GRUB menu won't be displayed. So press Shift immediately after POST to make GRUB menu visible. You can see the list of OS installed in your system and can select one of them

You can change various options in the Grub in Linux.

```
$ sudo vi /etc/default/grub

<<snip>>
GRUB_DEFAULT=0
GRUB_HIDDEN_TIMEOUT=0
GRUB_HIDDEN_TIMEOUT_QUIET=true
GRUB_TIMEOUT=10
<<end snip>>
```

To make the GRUB changes above effective, you need to run :

```
$ sudo update-grub
```

Now, during next booting any changes made in /etc/default/grub will be effective.

Some of the options you can modify are:

```
GRUB_DEFAULT  OS you want to boot, numbered 0 onwards. 0 by default.
GRUB_HIDDEN_TIMEOUT  timeout to show GRUB (in sec)
GRUB_HIDDEN_TIMEOUT_QUIET=true  to display the time for the above
GRUB_TIMEOUT  time to load the default OS (in sec)
```

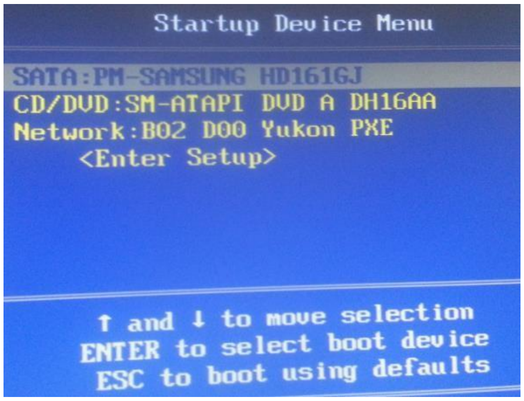
Make changes to GRUB_TIMEOUT; setting it to 0 would timeout immediately or you can increase the time to say 60. In this case, the menu will be displayed for 60 seconds.

2. BIOS

You can change options in ROM-BIOS which will affect the whole system.

Find out the interrupt key. It will likely be one of F8, F12, or Del. To find out the interrupt key, restart your system. When the system powers, press the interrupt key to enter the BIOS menu.

(NOTE: DO NOT CHANGE THE PASSWORD IN BIOS)

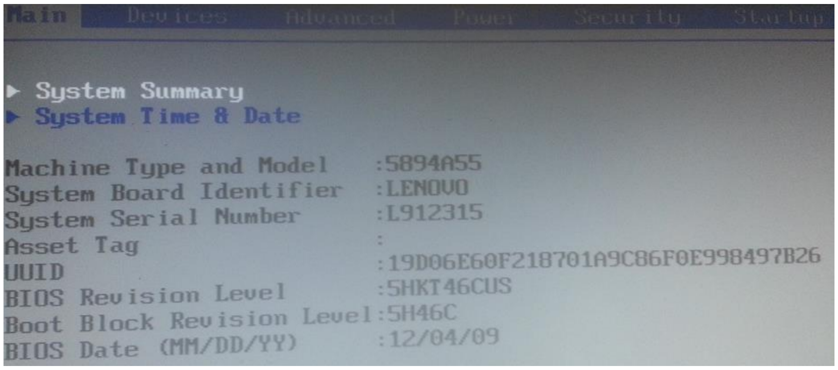


```
         Startup Device Menu

SATA:PM-SAMSUNG HD161GJ
CD/DVD:SM-ATAPI DVD A DH16AA
Network:B02 D00 Yukon PXE
    <Enter Setup>

     ↑ and ↓ to move selection
    ENTER to select boot device
     ESC to boot using defaults
```

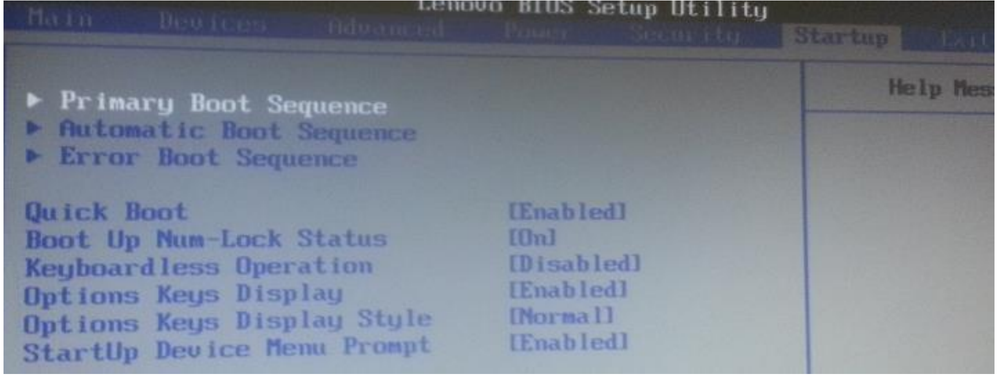
Select “Enter Setup”. You will get screen like below. You can view various system related stuff or make changes to the system parameters here.



```
▶ System Summary
▶ System Time & Date

Machine Type and Model      :5894A55
System Board Identifier     :LENOVO
System Serial Number        :L912315
Asset Tag                   :
UUID                        :19D06E60F218701A9C86F0E998497B26
BIOS Revision Level         :5HKT46CUS
Boot Block Revision Level   :5H46C
BIOS Date (MM/DD/YY)        :12/04/09
```

You can make changes to the boot order of various devices from Primary Boot Sequence.



```
▶ Primary Boot Sequence
▶ Automatic Boot Sequence
▶ Error Boot Sequence

Quick Boot                        [Enabled]
Boot Up Num-Lock Status           [On]
Keyboardless Operation            [Disabled]
Options Keys Display              [Enabled]
Options Keys Display Style        [Normal]
StartUp Device Menu Prompt        [Enabled]
```

Homework: Install Damn Small Linux in a thumb drive, and boot the OS from it. See instructions on how to do at:
http://www.pendrivelinux.com/all-in-one-usb-dsl/