Computer Comm & Networks - ITCS 8166 (Assignment – 1)

Abdullah Al Raqibul Islam (UNCC ID# **801151189**)

Software Details

1. Host Operating System and version: macOS (Version: 10.15.7)
2. Virtualization tool name and version: VirtualBox (Version: 6.1.16)

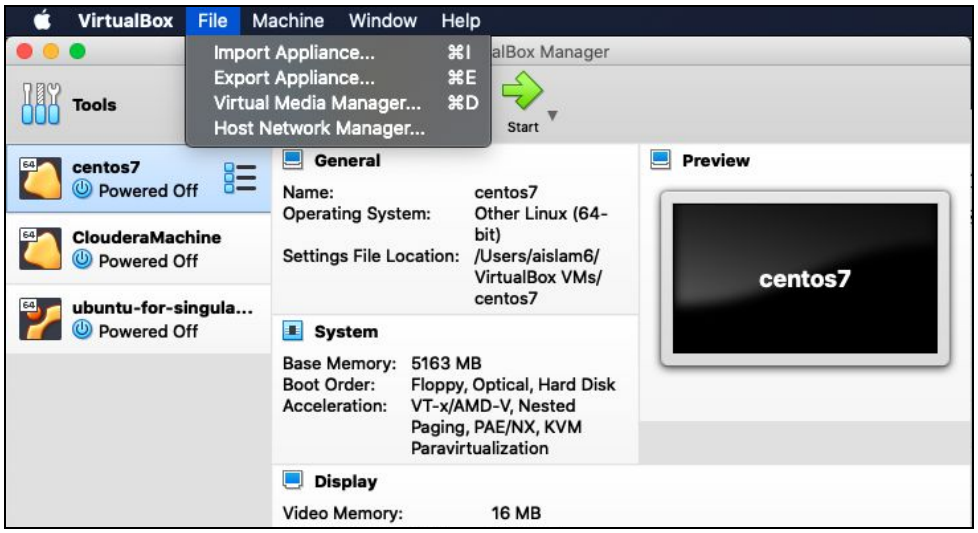
Part 1

Installation Process:

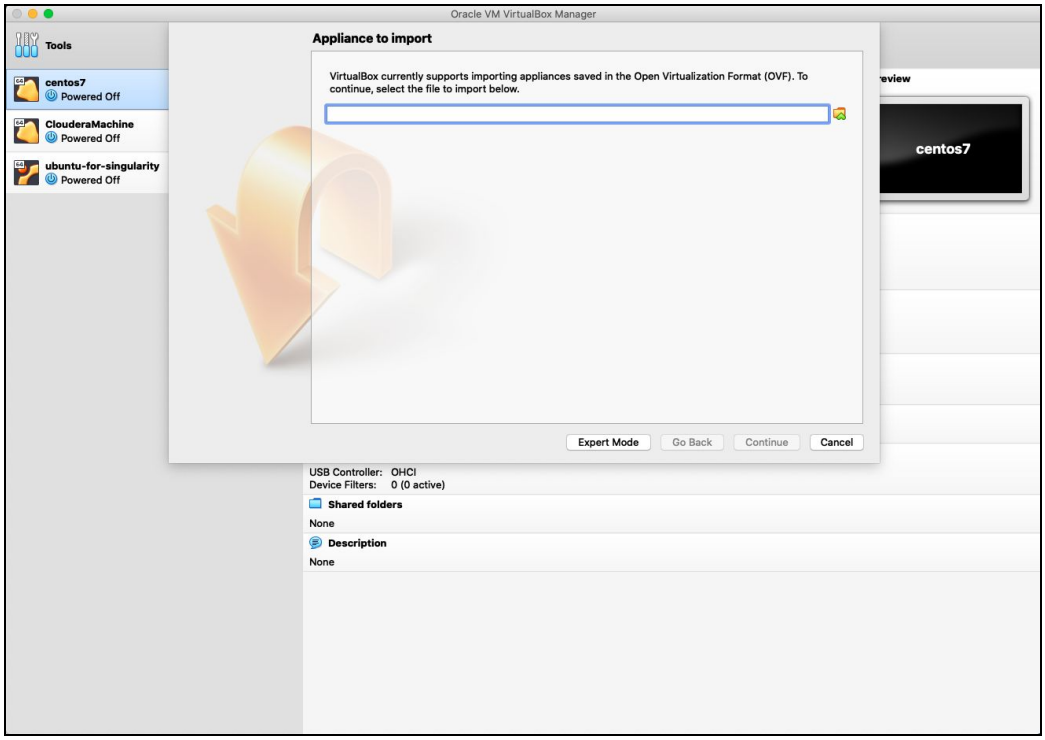
I already have a VirtualBox copy installed on my host computer. The details of the host and virtualization tool has been listed above.

Step 1:

Go to virtual box and in file menu, select “Import Appliance”

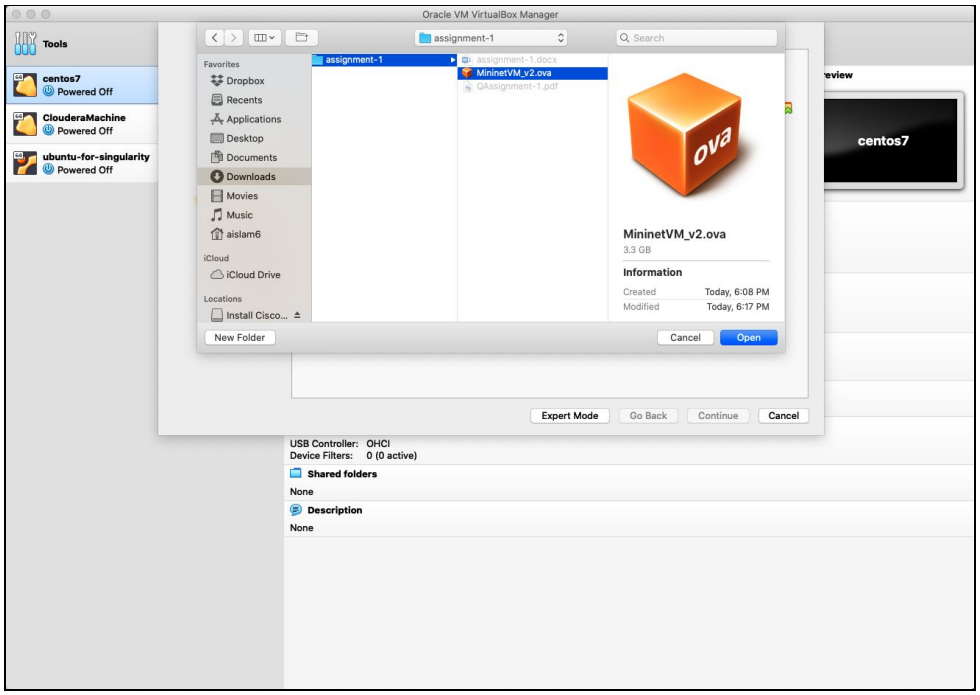


Step 2:

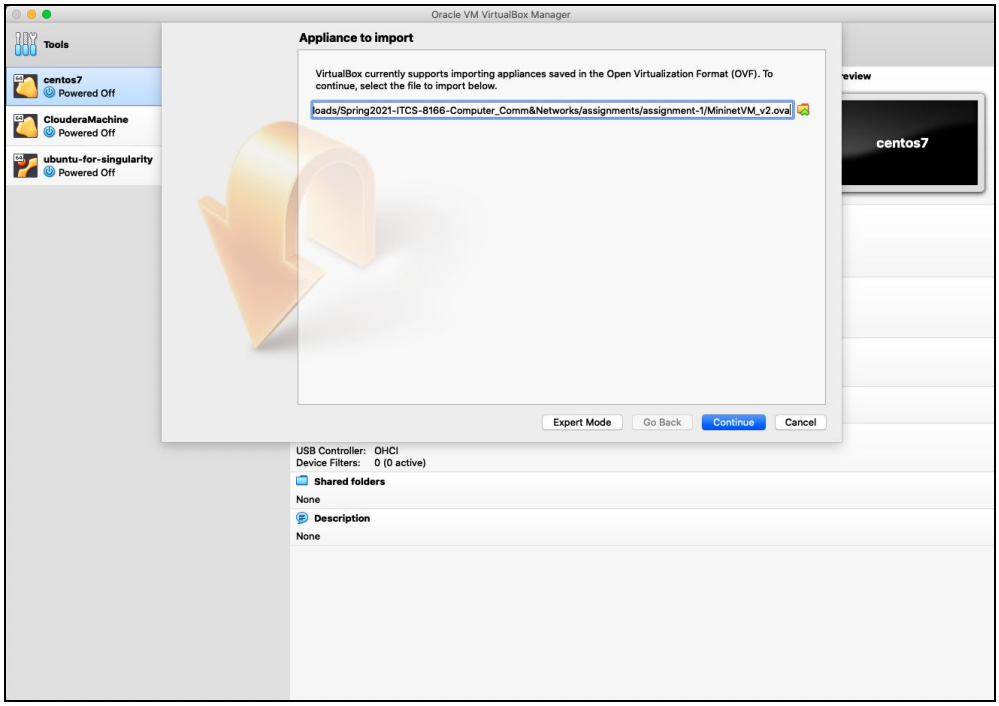


Step 3:

Browse the downloaded VM image as shown below

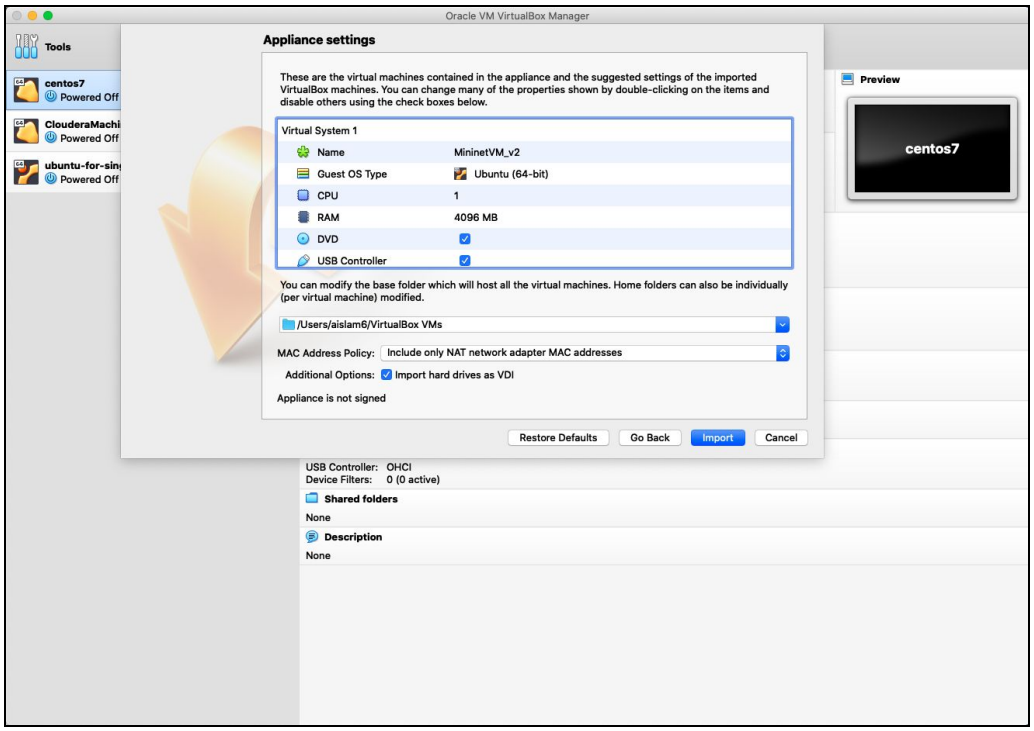


Step 4:

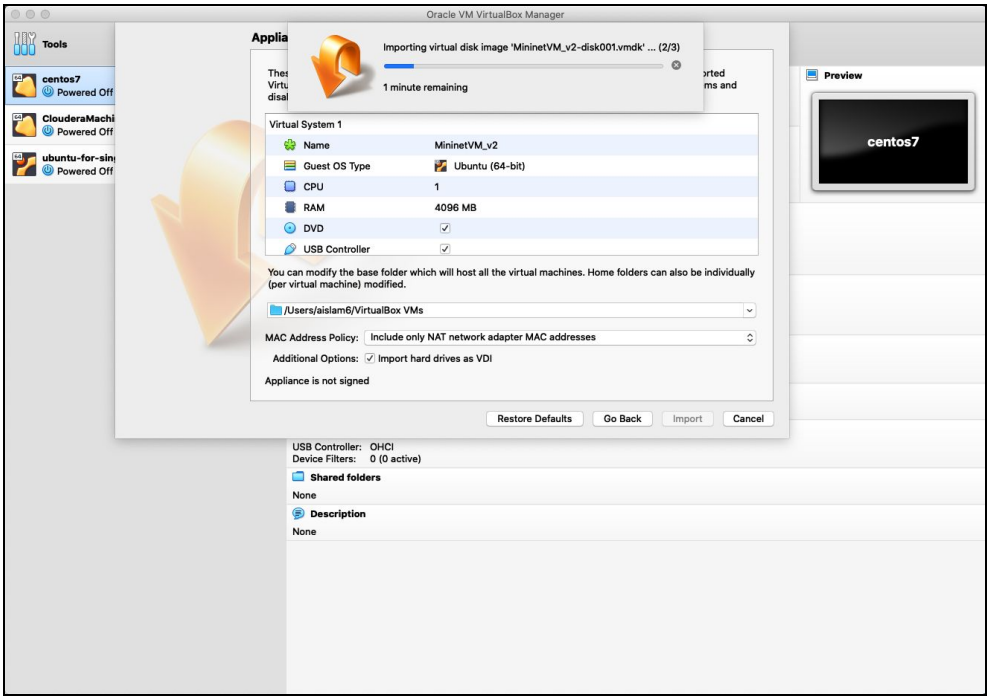


Step 5:

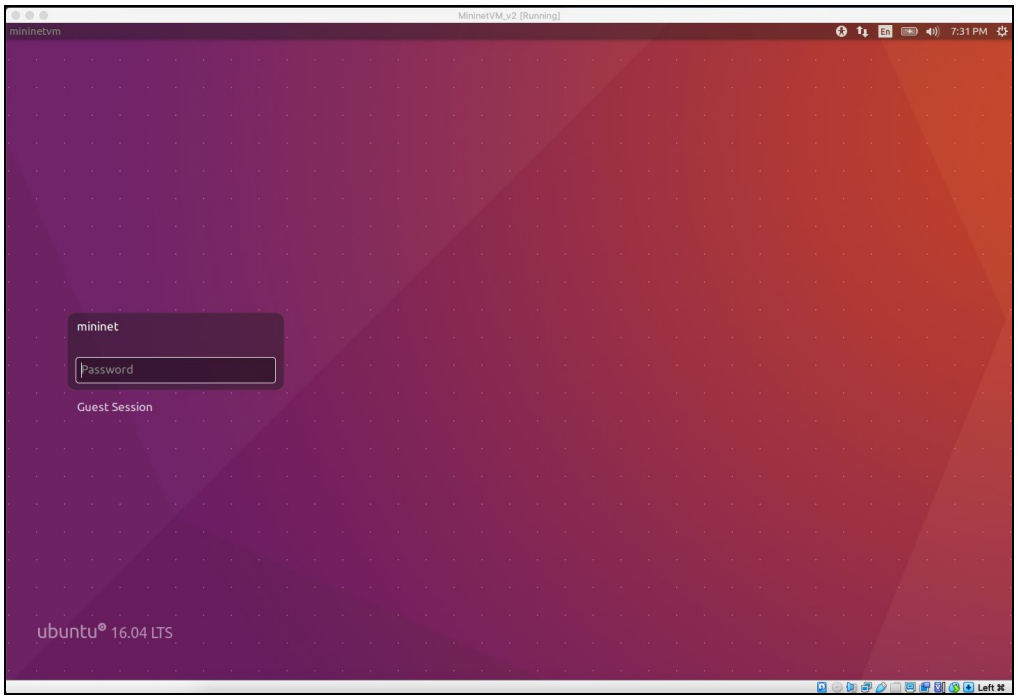
Once clicked on next, it will ask to “import’. Click on “import”



Step 6:

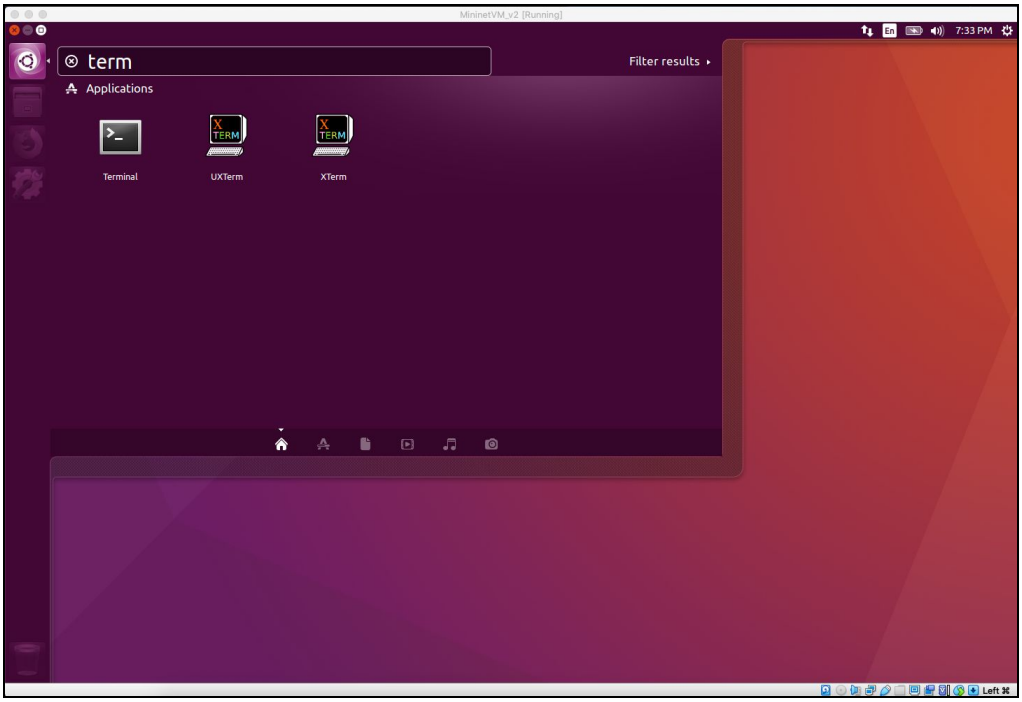


Step 7:



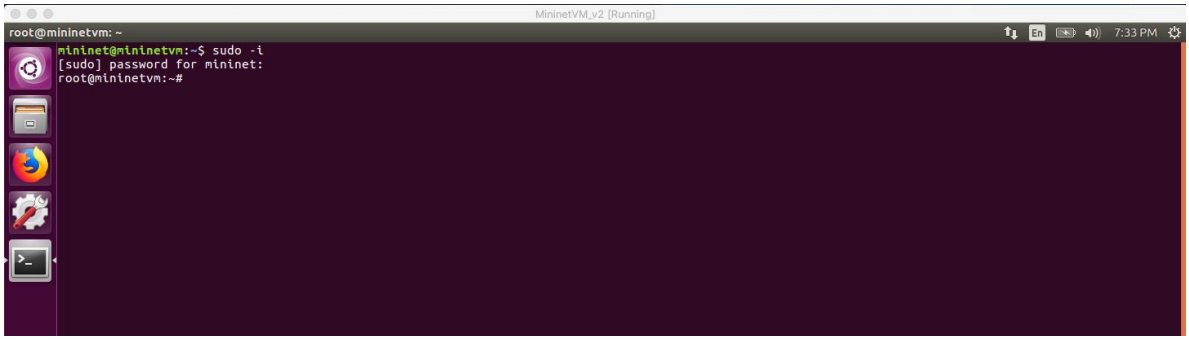
Step 8:

Open the terminal in VM



Step 9:

Switch to root by executing the following command.



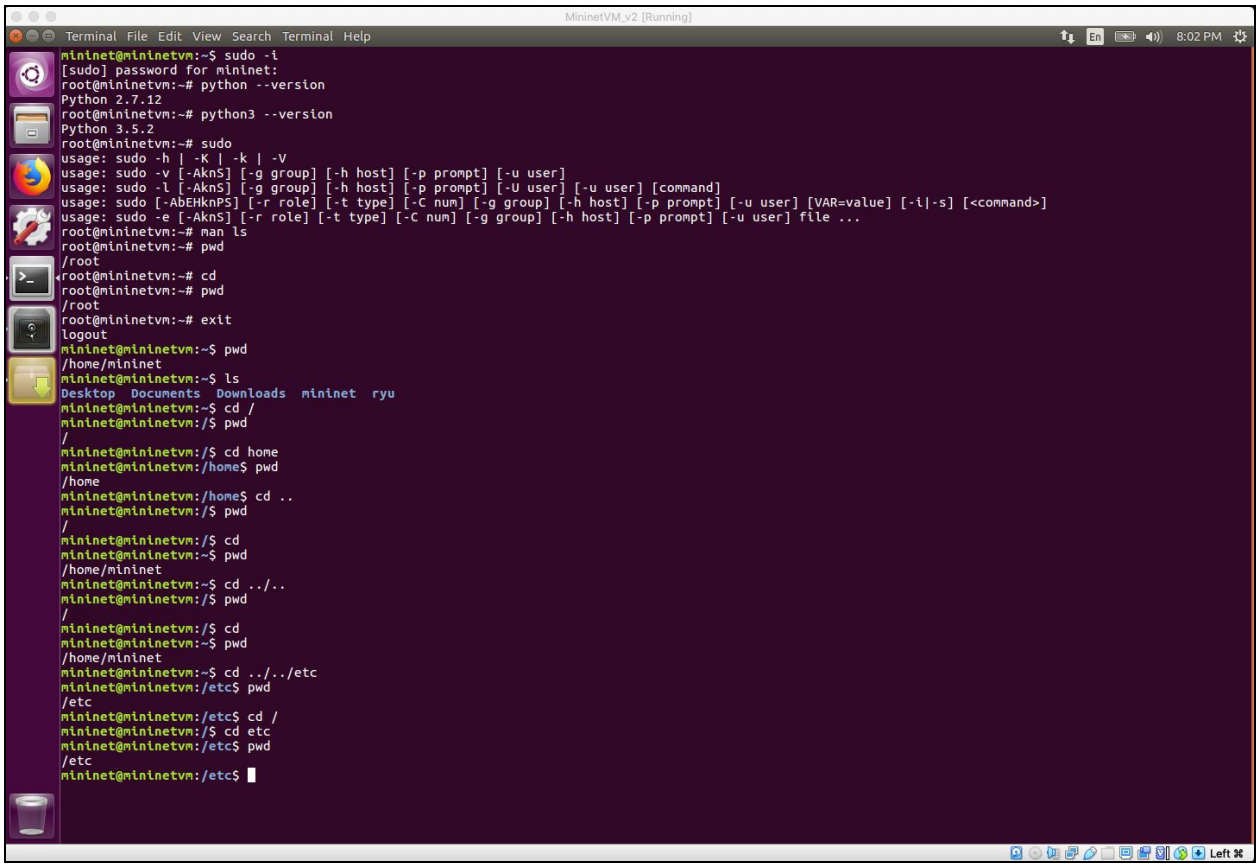
Part 2

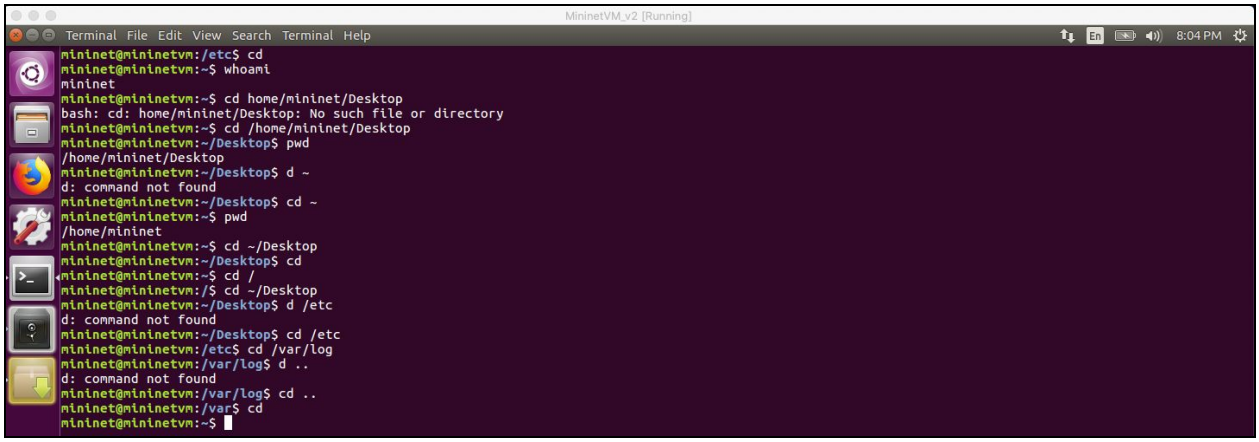
1:

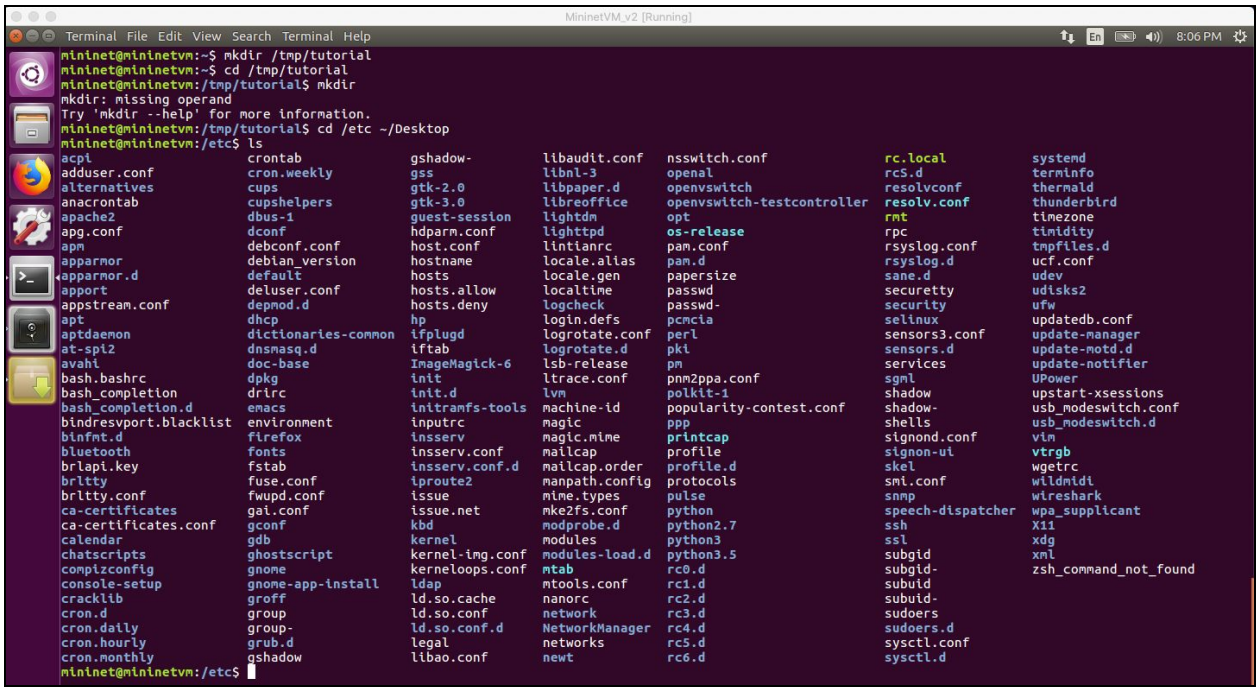
Read and try out below commands (with options) in the given terminal. You must identify the purpose of the commands and get familiar with it. (10 Points)

```
sudo
man ls
cd
cd.. cd
~
mkdir cp
rm
mv
grep
cat
```

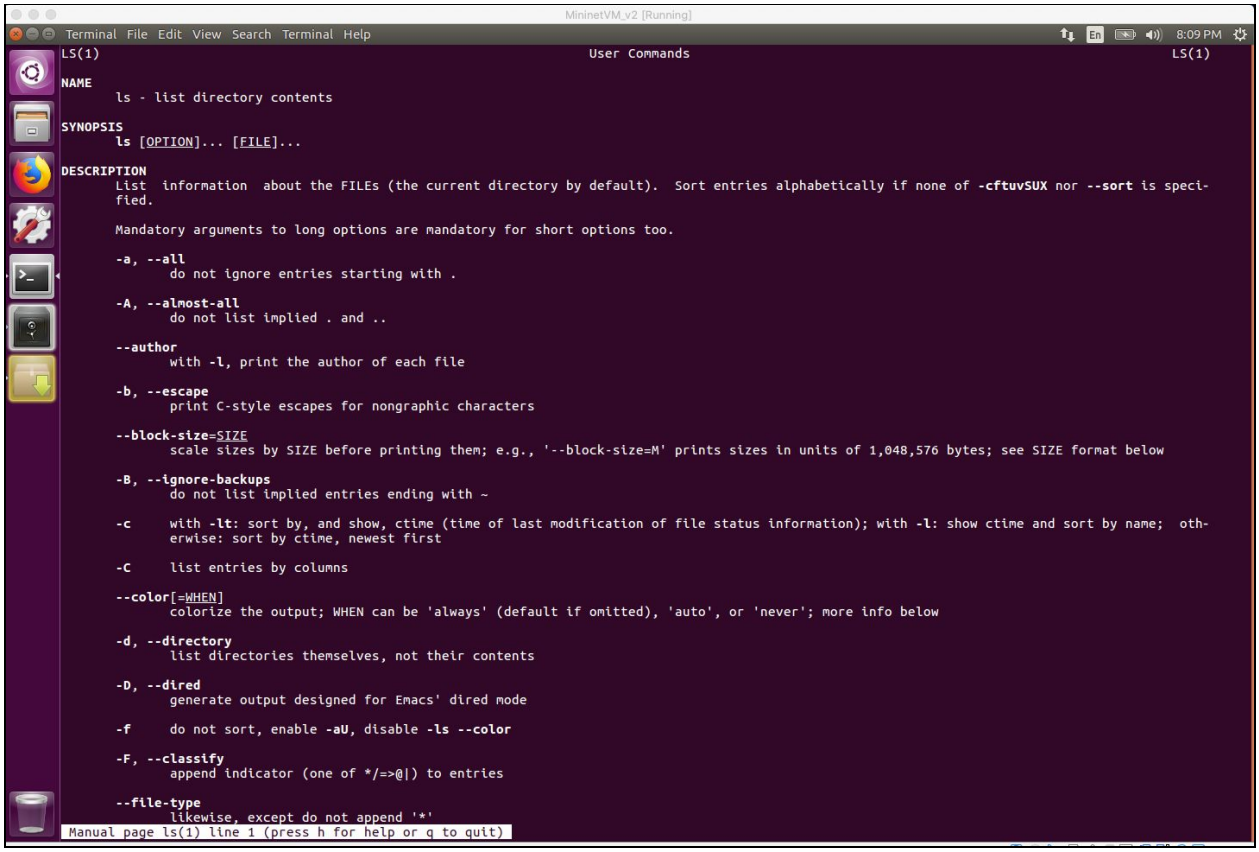
and many other Linux commands.

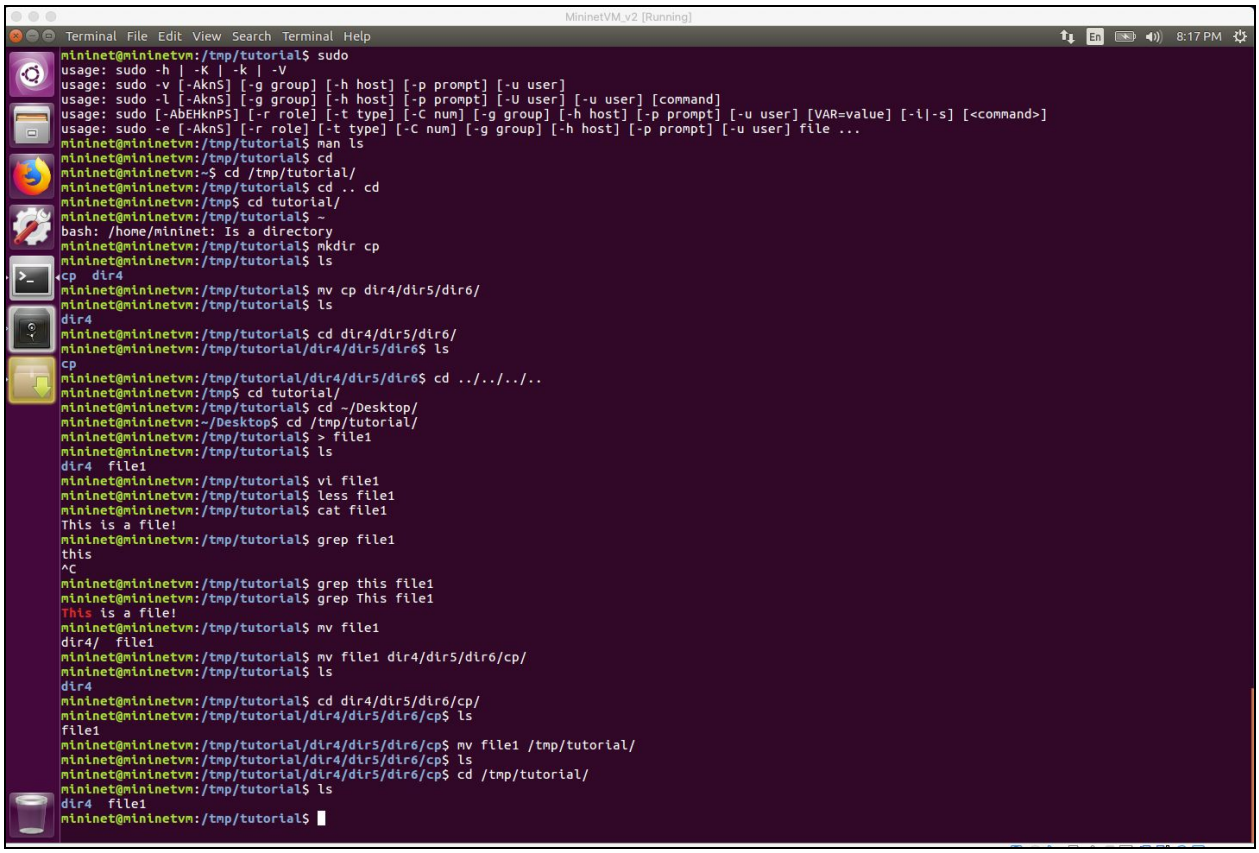



MininetVM_v2 [Running]
Terminal File Edit View Search Terminal Help
8:04 PM
mininet@mininetvm:/etc$ cd
mininet@mininetvm:~$ whoami
mininet
mininet@mininetvm:~$ cd home/mininet/Desktop
bash: cd: home/mininet/Desktop: No such file or directory
mininet@mininetvm:~$ cd /home/mininet/Desktop
mininet@mininetvm:~/Desktop$ pwd
/home/mininet/Desktop
mininet@mininetvm:~/Desktop$ d ~
d: command not found
mininet@mininetvm:~/Desktop$ cd ~
mininet@mininetvm:~$ pwd
/home/mininet
mininet@mininetvm:~$ cd ~/Desktop
mininet@mininetvm:~/Desktop$ cd
mininet@mininetvm:~$ cd /
mininet@mininetvm:/$ cd ~/Desktop
mininet@mininetvm:~/Desktop$ d /etc
d: command not found
mininet@mininetvm:~/Desktop$ cd /etc
mininet@mininetvm:/etc$ cd /var/log
mininet@mininetvm:/var/log$ d ..
d: command not found
mininet@mininetvm:/var/log$ cd ..
mininet@mininetvm:/var$ cd
mininet@mininetvm:~$


MininetVM_v2 [Running]
Terminal File Edit View Search Terminal Help
8:06 PM
mininet@mininetvm:~$ mkdir /tmp/tutorial
mininet@mininetvm:~$ cd /tmp/tutorial
mininet@mininetvm:/tmp/tutorial$ mkdir
mkdir: missing operand
Try 'mkdir --help' for more information.
mininet@mininetvm:/tmp/tutorial$ cd /etc ~/Desktop
mininet@mininetvm:/etc$ ls
acpi adduser.conf alternatives anacrontab apache2 apg.conf apm apparmor apparmor.d apport appstream.conf apt aptdaemon at-spi2 avahi bash.bashrc bash_completion bash_completion.d bindresvport.blacklist binfmt.d bluetooth brlapi.key brltty brltty.conf ca-certificates ca-certificates.conf calendar chatscripts compizconfig console-setup cracklib cron.d cron.daily cron.hourly cron.monthly
crontab cron.weekly cups cupshelpers dbus-1 dconf debconf.conf debian_version default deluser.conf depmod.d dhcp dictionaries-common dnsmasq.d doc-base dpkg drirc emacs environment firefox fonts fstab fuse.conf fwupd.conf gai.conf gconf gdb ghostscript gnome gnome-app-install groff group group- grub.d gshadow
gshadow- gss gtk-2.0 gtk-3.0 guest-session hdparm.conf host.conf hostname hosts hosts.allow hosts.deny hp ifplugd iftab ImageMagick-6 init init.d initramfs-tools inputrc insserv insserv.conf insserv.conf.d iproute2 issue issue.net kbd kernel kernel-img.conf kerneloops.conf ldap ld.so.cache ld.so.conf ld.so.conf.d legal libao.conf
libaudit.conf libnl-3 libpaper.d libreoffice lightdm lighttpd lintianrc locale.alias locale.gen localtime logcheck login.defs logrotate.conf logrotate.d lsb-release ltrace.conf lvm machine-id magic magic.mime mailcap mailcap.order manpath.config mime.types mke2fs.conf modprobe.d modules modules-load.d mtab mtools.conf nanorc network NetworkManager networks newt
nsswitch.conf openal openvswitch openvswitch-testcontroller opt os-release pam.conf pam.d papersize passwd passwd- pcmcia perl pki pm pnm2ppa.conf polkit-1 popularity-contest.conf ppp printcap profile profile.d protocols pulse python python2.7 python3 python3.5 rc0.d rc1.d rc2.d rc3.d rc4.d rc5.d rc6.d
rc.local rcS.d resolvconf resolv.conf rmt rpc rsyslog.conf rsyslog.d sane.d securetty security selinux sensors3.conf sensors.d services sgml shadow shadow- shells signond.conf signon-ui skel smi.conf snmp speech-dispatcher ssh ssl subgid subgid- subuid subuid- sudoers sudoers.d sysctl.conf sysctl.d
systemd terminfo thermald thunderbird timezone timidity tmpfiles.d ucf.conf udev udisks2 ufw updatedb.conf update-manager update-motd.d update-notifier UPower upstart-xsessions usb_modeswitch.conf usb_modeswitch.d vim vtrgb wgetrc wildmidi wireshark wpa_supplicant X11 xdg xml zsh_command_not_found
mininet@mininetvm:/etc$


MininetVM_v2 [Running]
Terminal File Edit View Search Terminal Help
8:09 PM
mininet@mininetvm:/tmp/tutorial$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-u user] file ...
mininet@mininetvm:/tmp/tutorial$


MininetVM_v2 [Running]
Terminal File Edit View Search Terminal Help
8:09 PM
LS(1) User Commands LS(1)
NAME
ls - list directory contents
SYNOPSIS
ls [OPTION]... [FILE]...
DESCRIPTION
List information about the FILEs (the current directory by default). Sort entries alphabetically if none of -cftuvSUX nor --sort is specified.
Mandatory arguments to long options are mandatory for short options too.
-a, --all
do not ignore entries starting with .
-A, --almost-all
do not list implied . and ..
--author
with -l, print the author of each file
-b, --escape
print C-style escapes for nongraphic characters
--block-size=SIZE
scale sizes by SIZE before printing them; e.g., '--block-size=M' prints sizes in units of 1,048,576 bytes; see SIZE format below
-B, --ignore-backups
do not list implied entries ending with ~
-c with -lt: sort by, and show, ctime (time of last modification of file status information); with -l: show ctime and sort by name; otherwise: sort by ctime, newest first
-C list entries by columns
--color[=WHEN]
colorize the output; WHEN can be 'always' (default if omitted), 'auto', or 'never'; more info below
-d, --directory
list directories themselves, not their contents
-D, --dired
generate output designed for Emacs' dired mode
-f do not sort, enable -aU, disable -ls --color
-F, --classify
append indicator (one of */=>@|) to entries
--file-type
likewise, except do not append '*'
Manual page ls(1) line 1 (press h for help or q to quit)

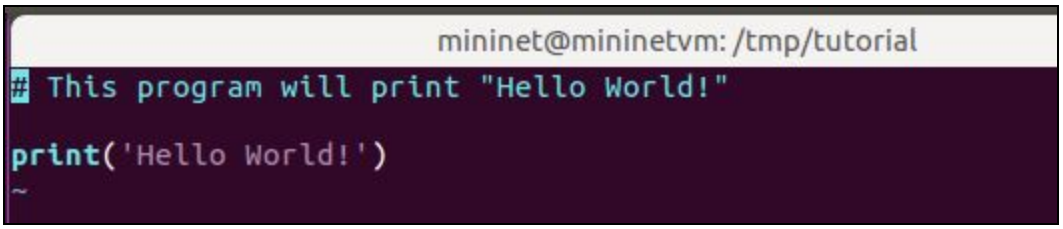

MininetVM_v2 [Running]
Terminal File Edit View Search Terminal Help
8:17 PM
mininet@mininetvm:/tmp/tutorial$ sudo
usage: sudo -h | -K | -k | -V
usage: sudo -v [-AknS] [-g group] [-h host] [-p prompt] [-u user]
usage: sudo -l [-AknS] [-g group] [-h host] [-p prompt] [-U user] [-u user] [command]
usage: sudo [-AbEHknPS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-u user] [VAR=value] [-i|-s] [<command>]
usage: sudo -e [-AknS] [-r role] [-t type] [-C num] [-g group] [-h host] [-p prompt] [-u user] file ...
mininet@mininetvm:/tmp/tutorial$ man ls
mininet@mininetvm:/tmp/tutorial$ cd
mininet@mininetvm:~$ cd /tmp/tutorial/
mininet@mininetvm:/tmp/tutorial$ cd .. cd
mininet@mininetvm:/tmp$ cd tutorial/
mininet@mininetvm:/tmp/tutorial$ ~
bash: /home/mininet: Is a directory
mininet@mininetvm:/tmp/tutorial$ mkdir cp
mininet@mininetvm:/tmp/tutorial$ ls
cp dir4
mininet@mininetvm:/tmp/tutorial$ mv cp dir4/dir5/dir6/
mininet@mininetvm:/tmp/tutorial$ ls
dir4
mininet@mininetvm:/tmp/tutorial$ cd dir4/dir5/dir6/
mininet@mininetvm:/tmp/tutorial/dir4/dir5/dir6$ ls
cp
mininet@mininetvm:/tmp/tutorial/dir4/dir5/dir6$ cd ../../../..
mininet@mininetvm:/tmp$ cd tutorial/
mininet@mininetvm:/tmp/tutorial$ cd ~/Desktop/
mininet@mininetvm:~/Desktop$ cd /tmp/tutorial/
mininet@mininetvm:/tmp/tutorial$ > file1
mininet@mininetvm:/tmp/tutorial$ ls
dir4 file1
mininet@mininetvm:/tmp/tutorial$ vi file1
mininet@mininetvm:/tmp/tutorial$ less file1
mininet@mininetvm:/tmp/tutorial$ cat file1
This is a file!
mininet@mininetvm:/tmp/tutorial$ grep file1
this
^C
mininet@mininetvm:/tmp/tutorial$ grep this file1
mininet@mininetvm:/tmp/tutorial$ grep This file1
This is a file!
mininet@mininetvm:/tmp/tutorial$ mv file1
dir4/ file1
mininet@mininetvm:/tmp/tutorial$ mv file1 dir4/dir5/dir6/cp/
mininet@mininetvm:/tmp/tutorial$ ls
dir4
mininet@mininetvm:/tmp/tutorial$ cd dir4/dir5/dir6/cp/
mininet@mininetvm:/tmp/tutorial/dir4/dir5/dir6/cp$ ls
file1
mininet@mininetvm:/tmp/tutorial/dir4/dir5/dir6/cp$ mv file1 /tmp/tutorial/
mininet@mininetvm:/tmp/tutorial/dir4/dir5/dir6/cp$ ls
mininet@mininetvm:/tmp/tutorial/dir4/dir5/dir6/cp$ cd /tmp/tutorial/
mininet@mininetvm:/tmp/tutorial$ ls
dir4 file1
mininet@mininetvm:/tmp/tutorial$

2:

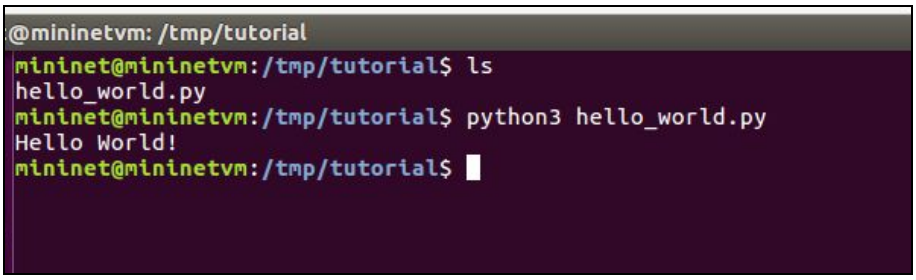
Write a python program which will print “hello world”. You must write this program in Linux terminal usinganyeditorandthenexecuteitonthesame.(10points)

hint: Use“sudo python <file-name.py>”to run any python file from the terminal

Code:



Programming Output:



3:

Write a program that asks the user to enter their “name” and their “age”. Print a message addressed to them that tells them if he/she is eligible to vote if the user’s age is 18 or more or else print a message saying the user is not eligible to vote. (hint: Use if/else conditions to code the logic) (10 points)

Ref:

1. https://www.tutorialspoint.com/python/python_basic_syntax.htm
2. https://www.tutorialspoint.com/python/python_variable_types.htm

Code:

```
mininet@mininetvm: /tmp/tutorial
print("Hi! I would like to know your name and age!")
name = input("Name: ")
age_txt = input("Age: ")

age = int(age_txt)

if age >= 18:
        print("Hello {}! Congratulation! You are eligible to vote!".format(name))
else:
        print("Hello {}! Sorry you are eligible to vote!".format(name))
~
~
```

Programming Output:

```
mininet@mininetvm:/tmp/tutorial$ ls
hello_world.py
mininet@mininetvm:/tmp/tutorial$ > vote.py
mininet@mininetvm:/tmp/tutorial$ vim vote.py
mininet@mininetvm:/tmp/tutorial$ python3 vote.py
Hi! I would like to know your name and age!
Name: Raqib
Age: 18
Hello Raqib! Congratulation! You are eligible to vote!
mininet@mininetvm:/tmp/tutorial$ python3 vote.py
Hi! I would like to know your name and age!
Name: Biqar
Age: 17
Hello Biqar! Sorry you are eligible to vote!
mininet@mininetvm:/tmp/tutorial$
```

4:

Write a python program in which bandwidth between 4 hosts (given as below) into a dictionary and a function to return the bandwidth of a link, when link is passed as a parameter. (10 points)

Links have p as following {(1,2):10Mbps, (2,3):5Mbps, (3,4):10Mbps}

For example, Input: Enter the link: (2,3)
Output should be: Band width for link between h2 and h3 is 5Mbps

Hint: Use dictionary, tuples, tuple unpacking in python

Ref:

1. https://realpython.com/python-lists-tuples/
2. https://realpython.com/python-dicts/

Code:

```
mininet@mininetvm: /tmp/tutorial
import re

d = { (1, 2) : "10 Mbps", (2, 3) : "5 Mbps", (3, 4) : "10 Mbps" }

def func(h1, h2):
        if (h1, h2) in d:
                return d[(h1, h2)]
        else:
                return "Not available!"

link = input("Enter the link: ")

link = link.replace('(', '').replace(')', '')
#link.strip('(\)\')
print(link)

hosts = link.split(',')
host1 = int(hosts[0])
host2 = int(hosts[1])

print("Bandwidth for link between h{} and h{} is: {}".format(host1, host2, func(host1, host2)))
```

Programming Output:

```
mininet@mininetvm: /tmp/tutorial
mininet@mininetvm:/tmp/tutorial$ > bandwidth.py
mininet@mininetvm:/tmp/tutorial$ ls
bandwidth.py  hello_world.py  vote.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ vim bandwidth.py
mininet@mininetvm:/tmp/tutorial$ python3 bandwidth.py
Enter the link: (1,2)
1,2
Bandwidth for link between h1 and h2 is: 10 Mbps
mininet@mininetvm:/tmp/tutorial$ python3 bandwidth.py
Enter the link: (2,3)
2,3
Bandwidth for link between h2 and h3 is: 5 Mbps
mininet@mininetvm:/tmp/tutorial$ python3 bandwidth.py
Enter the link: (3,4)
3,4
Bandwidth for link between h3 and h4 is: 10 Mbps
mininet@mininetvm:/tmp/tutorial$
```

5:

Write a decorator function that takes in 3 strings as arguments and returns them in reverse order. (10 points)

Ref:

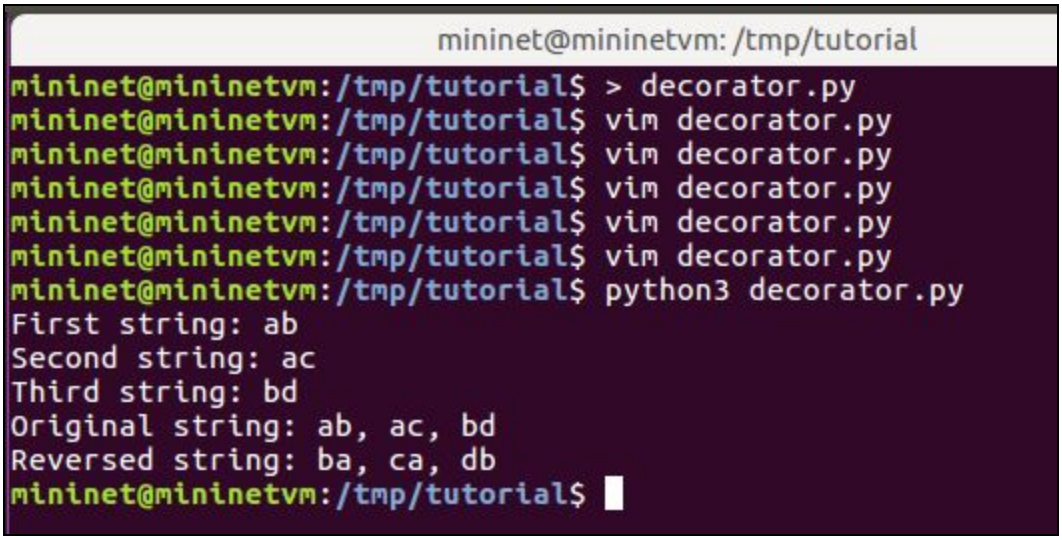
1. https://realpython.com/primer-on-python-decorators/
2. https://www.youtube.com/watch?v=FsAPt_9Bf3U

Code:

mininet@mininetvm: /tmp/tutorial

```
# Decorator function

def decorate_it(func):
        def wrapper(*args, **kwargs):
                return func(*args, **kwargs)
        return wrapper

def str_reverse(str1, str2, str3):
        rev_str1 = ''.join(reversed(str1))
        rev_str2 = ''.join(reversed(str2))
        rev_str3 = ''.join(reversed(str3))

        return rev_str1, rev_str2, rev_str3


s1 = input("First string: ")
s2 = input("Second string: ")
s3 = input("Third string: ")

print("Original string: {}, {}, {}".format(s1, s2, s3))

my_func = decorate_it(str_reverse)
r1, r2, r3 = my_func(s1, s2, s3)

print("Reversed string: {}, {}, {}".format(r1, r2, r3))
~
~
```

Programming Output:


mininet@mininetvm: /tmp/tutorial
mininet@mininetvm:/tmp/tutorial$ > decorator.py
mininet@mininetvm:/tmp/tutorial$ vim decorator.py
mininet@mininetvm:/tmp/tutorial$ vim decorator.py
mininet@mininetvm:/tmp/tutorial$ vim decorator.py
mininet@mininetvm:/tmp/tutorial$ vim decorator.py
mininet@mininetvm:/tmp/tutorial$ vim decorator.py
mininet@mininetvm:/tmp/tutorial$ python3 decorator.py
First string: ab
Second string: ac
Third string: bd
Original string: ab, ac, bd
Reversed string: ba, ca, db
mininet@mininetvm:/tmp/tutorial$