Report of Project-1 Linux

Zhan Wang 17307110421

April 14, 2021

1 Introduction

The overall goal of this project is to master Linux basic commands and compile shell script.
Environment: Ubuntu 18.04 + python 3.8

2 Question

E1. Use the command uname to simultaneously show the name and version of the system.

```
uname -s -v
```

E2. Use the command echo to show the full path of your home directory.

```
echo ~
```

E3. Use the command whoami to show the current login user.

```
whoami
```

E4. Use the command chmod to make the file 1.sh in the current directory executable.

```
chmod +x 1.sh
```

E5. Show the disk usage of the current directory, with depth 1, and human readable output.

```
du -h --max-depth=1
```

E6. Use ssh to connect to 10.19.248.12, with user oyel.

```
ssh oyel@10.19.248.12
```

E7. Use apt-get to install jq.

```
sudo apt-get install jq
```

E8. Display the absolute path of the current directory.

```
pwd
```

E9. Make a new file named 1.sh, with content: echo “hello world”.

```
vim 1.sh
```

Then write the following codes in 1.sh

```
#!/bin/bash
echo \hello world"
```

E10. Change the name of file 1.sh to 2.sh.

```
mv 1.sh 2.sh
```

E11. Make a new folder in current directory, named folder1.

```
mkdir folder1
```

E12. Copy 3.sh in current directory to folder1, still named 3.sh.

```
cp -i 3.sh ./folder1/3.sh
```

E13. Use the command cat to obtain the content of 2.sh and redirect the output to 3.sh.

```
cat 2.sh > 3.sh
```

E14. Create two new txt files, add line numbers to the contents of file 1, and then input file 1 to file 2.

```
cat > file1.txt # write content and exit with "ctrl+d"
cat -n file1.txt > file2.txt
```

E15. Show the name of all files and folders, including the hidden ones, in current directory.

```
ls -a
```

E16. Randomly select a file directory to list the files modified in the current directory and its subdirectories in the last 10 days.

```
find . -ctime -10
```

E17. Make the files and subdirectories of directory “dir” readable, writable, and executable to all users, use mode.

```
chmod 777 dir/
```

E18. Run a python program, query the pid number, and kill it with the kill command.

```
python test.py
ps | grep python
kill pid # pid: the pid number of the running python program
```

N1. Show line 6-10 of file /etc/hosts.

```
head -10 /etc/hosts | tail -5
```

N2. Copy the file 1.sh in the current directory to your home directory, the modification time should be preserved.

```
cp -p 1.sh ~/
```

N3. Check whether the files and subdirectories of directory "dir" are writable. If not, output "no x mode".

```
[ -w dir/ ] || echo "no x mode"
```

N4. Use grep to find all matches of pattern "url" in file "commits.json", print 1 line of leading and trailing context surrounding each match.

```
grep url commits.json -B 1 -A 1
```

N5. Sort all the files and subdirectories in the current folder according to size in descending order and print the largest three with filename and size. Use pipeline.

```
du -a  | sort -n -r | head -n 3
```

H1. Echo every 2 letter combination of a, b, c, and d including doubles. Do not write out all combinations manually, and do not use loops.

```
echo {a,b,c}{a,b,c}
```

H2. Use awk to show the first and the second column of command "ls -l", the header should not be included(grep -v).

```
ls -l | awk  'NR !=1 { print $1, $2 }'
```

H3. Transform all uppercase letters to lowercase in the file "input.txt", and output the result to "output.txt".

```
tr '[:upper:]' '[:lower:]' < input.txt > output.txt
```

H4. Write a script to sum from 1 to 100.

```
#!/bin/bash
sum=0
for i in `seq 1 100`
do
    sum=$[$i+$sum]
done
echo $sum
```

H5. Write a script to show the number of lines of each file in the current directory.

```
#!/bin/bash
for f in ./*
do
    [ -f $f ] && echo ‘wc -l $f‘
done
```

H6. Write a script. Count the number of users currently logged in to the system, and judge whether it is more than three. If it is, display the actual number and give a warning message. Otherwise, list the account name and terminal of the logged-in user.

```
#!/bin/bash
i=$( users | wc -w )
if [ $i -ge 3 ]; then
    echo "Warning! $i users!"
else
    echo ‘who | awk ’{ print $1, $2 }’‘
fi
```