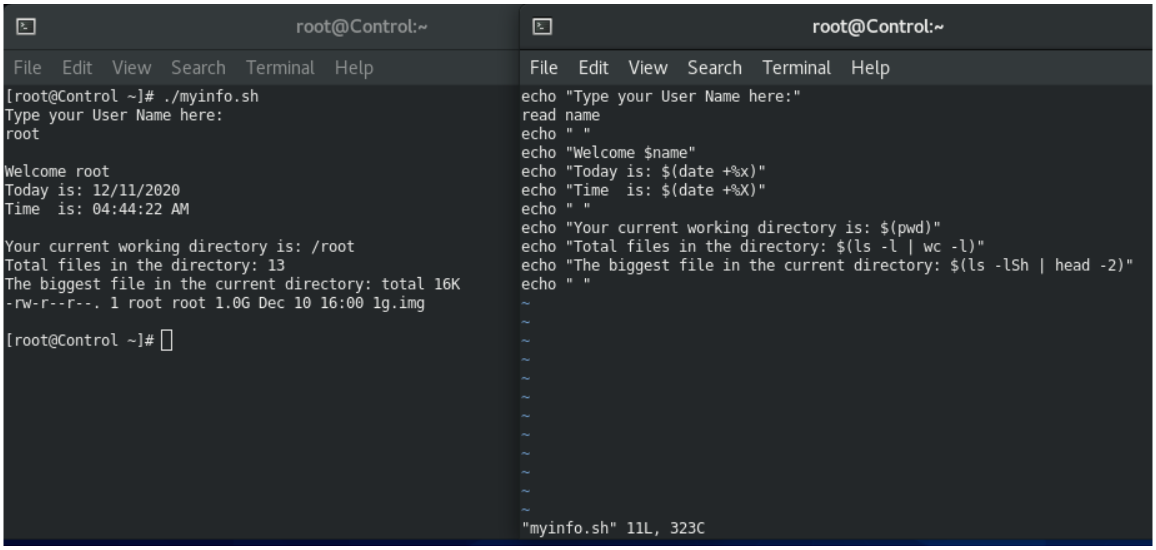
Assignment 0

1. **Create a simple shell script to tell the user about their session – they need to know:**

- What their username is
- What the current date is
- What the time is
- What their current working directory is
- How many files they have in that directory
- What is the biggest file in their current directory

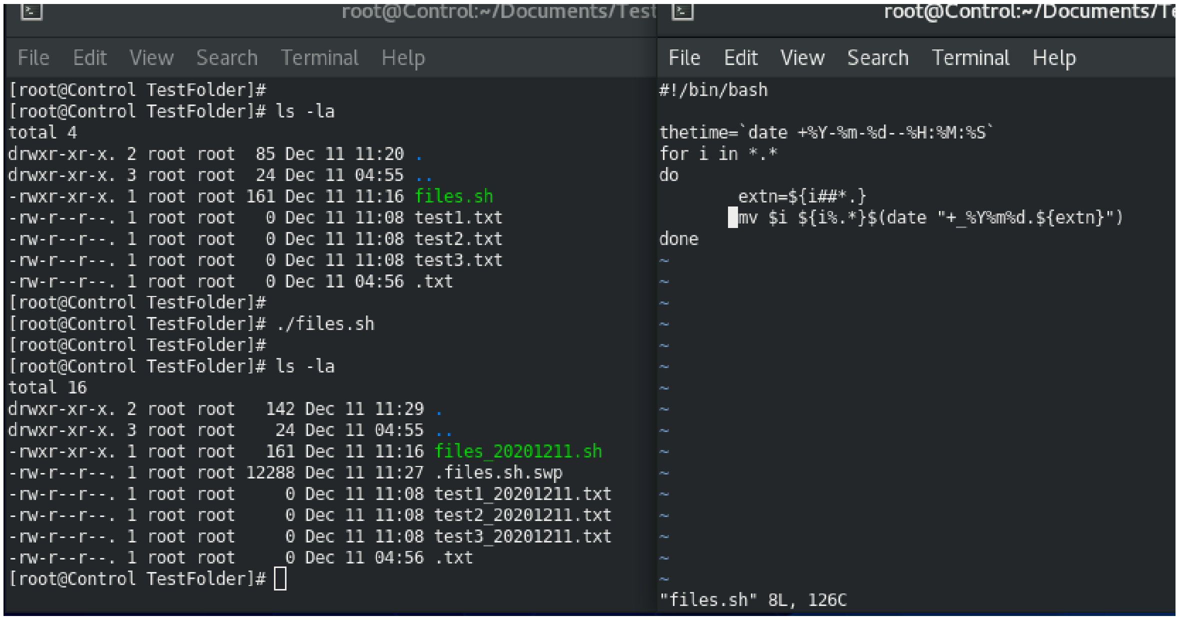

root@Control:~
File Edit View Search Terminal Help
root@Control:~
File Edit View Search Terminal Help

```
[root@Control ~]# ./myinfo.sh
Type your User Name here:
root

Welcome root
Today is: 12/11/2020
Time  is: 04:44:22 AM

Your current working directory is: /root
Total files in the directory: 13
The biggest file in the current directory: total 16K
-rw-r--r--. 1 root root 1.0G Dec 10 16:00 1g.img

[root@Control ~]# 
```

```
echo "Type your User Name here:"
read name
echo " "
echo "Welcome $name"
echo "Today is: $(date +%x)"
echo "Time  is: $(date +%X)"
echo " "
echo "Your current working directory is: $(pwd)"
echo "Total files in the directory: $(ls -l | wc -l)"
echo "The biggest file in the current directory: $(ls -lSh | head -2)"
echo " "
~
~
~
~
~
~
~
~
~
~
~
~
"myinfo.sh" 11L, 323C
```

Assignment 1

Create a directory with a few test files in it (the files can be empty). Now write a script that for every file in that directory you rename it to have an extension of today's date in YYYYMMDD format.

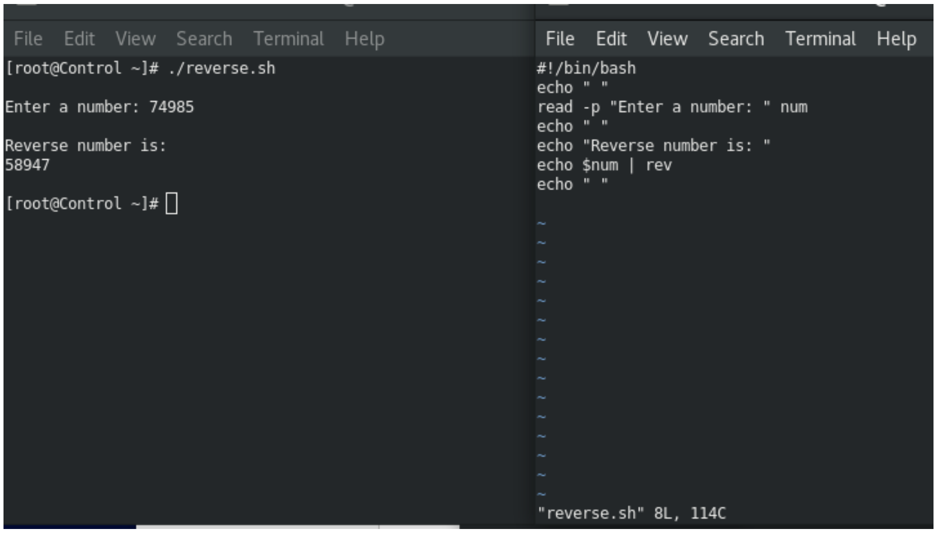

root@Control:~/Documents/Test
File Edit View Search Terminal Help
root@Control:~/Documents/Te
File Edit View Search Terminal Help

```
[root@Control TestFolder]#
[root@Control TestFolder]# ls -la
total 4
drwxr-xr-x. 2 root root  85 Dec 11 11:20 .
drwxr-xr-x. 3 root root  24 Dec 11 04:55 ..
-rwxr-xr-x. 1 root root 161 Dec 11 11:16 files.sh
-rw-r--r--. 1 root root   0 Dec 11 11:08 test1.txt
-rw-r--r--. 1 root root   0 Dec 11 11:08 test2.txt
-rw-r--r--. 1 root root   0 Dec 11 11:08 test3.txt
-rw-r--r--. 1 root root   0 Dec 11 04:56 .txt
[root@Control TestFolder]#
[root@Control TestFolder]# ./files.sh
[root@Control TestFolder]#
[root@Control TestFolder]# ls -la
total 16
drwxr-xr-x. 2 root root   142 Dec 11 11:29 .
drwxr-xr-x. 3 root root    24 Dec 11 04:55 ..
-rwxr-xr-x. 1 root root   161 Dec 11 11:16 files_20201211.sh
-rw-r--r--. 1 root root 12288 Dec 11 11:27 .files.sh.swp
-rw-r--r--. 1 root root     0 Dec 11 11:08 test1_20201211.txt
-rw-r--r--. 1 root root     0 Dec 11 11:08 test2_20201211.txt
-rw-r--r--. 1 root root     0 Dec 11 11:08 test3_20201211.txt
-rw-r--r--. 1 root root     0 Dec 11 04:56 .txt
[root@Control TestFolder]# 
```

```
#!/bin/bash

thetime=`date +%Y-%m-%d--%H:%M:%S`
for i in *.*
do
        extn=${i##*.}
        mv $i ${i%.*}$(date "+_%Y%m%d.${extn}")
done
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"files.sh" 8L, 126C
```

Assignment 2

Write a script that takes a number as an input and reverses it out to the user. For example, if the original number is 74985, the output should be 58947.


File Edit View Search Terminal Help
File Edit View Search Terminal Help

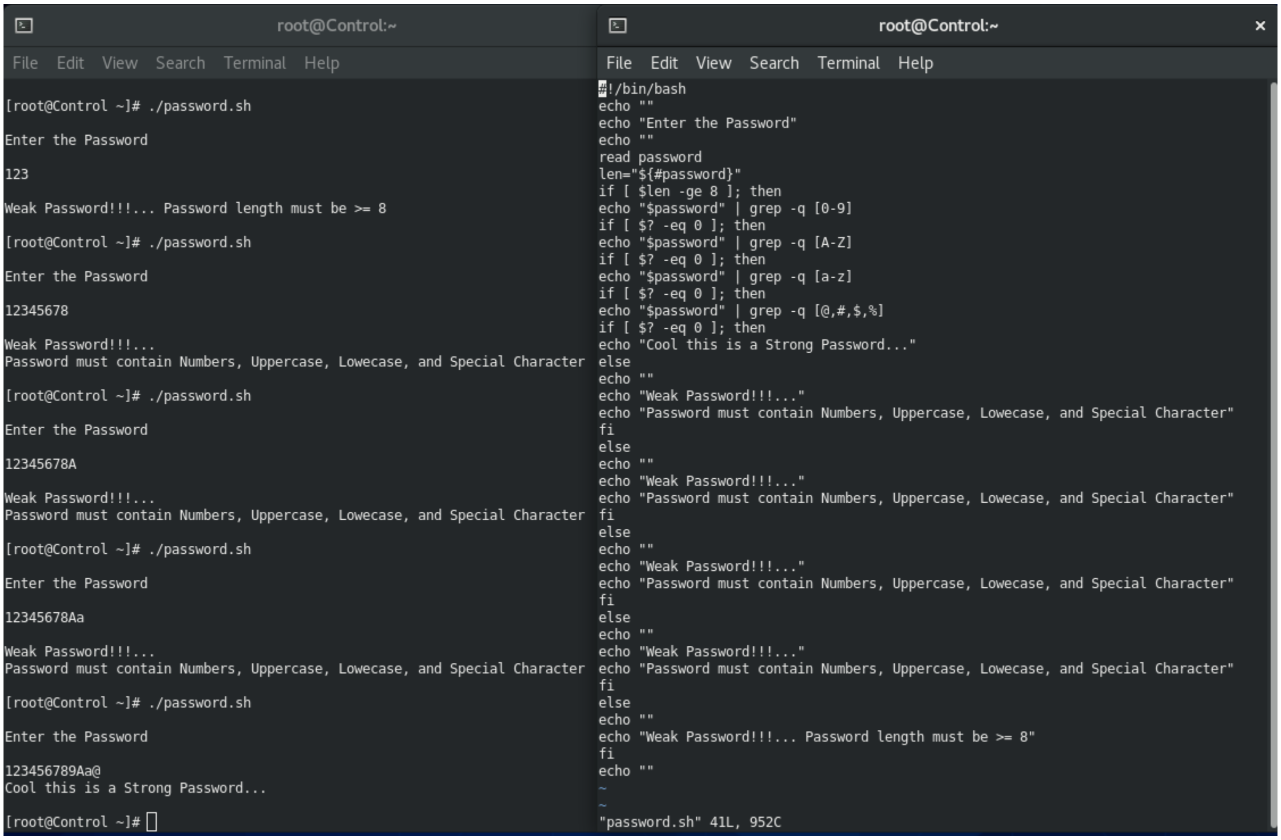
```
[root@Control ~]# ./reverse.sh

Enter a number: 74985

Reverse number is:
58947

[root@Control ~]#
```

```
#!/bin/bash
echo " "
read -p "Enter a number: " num
echo " "
echo "Reverse number is: "
echo $num | rev
echo " "

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"reverse.sh" 8L, 114C
```

Assignment 3

Write a script to validate how secure someone's password is. Things you would care about:

- Length should be 8 or more characters
- The password should contain numbers and letters
- There should be both uppercase and lowercase letters


root@Control:~
File Edit View Search Terminal Help
root@Control:~
File Edit View Search Terminal Help

```
[root@Control ~]# ./password.sh

Enter the Password

123

Weak Password!!!... Password length must be >= 8

[root@Control ~]# ./password.sh

Enter the Password

12345678

Weak Password!!!...
Password must contain Numbers, Uppercase, Lowecase, and Special Character

[root@Control ~]# ./password.sh

Enter the Password

12345678A

Weak Password!!!...
Password must contain Numbers, Uppercase, Lowecase, and Special Character

[root@Control ~]# ./password.sh

Enter the Password

12345678Aa

Weak Password!!!...
Password must contain Numbers, Uppercase, Lowecase, and Special Character

[root@Control ~]# ./password.sh

Enter the Password

123456789Aa@
Cool this is a Strong Password...

[root@Control ~]#
```

```
#!/bin/bash
echo ""
echo "Enter the Password"
echo ""
read password
len="${#password}"
if [ $len -ge 8 ]; then
echo "$password" | grep -q [0-9]
if [ $? -eq 0 ]; then
echo "$password" | grep -q [A-Z]
if [ $? -eq 0 ]; then
echo "$password" | grep -q [a-z]
if [ $? -eq 0 ]; then
echo "$password" | grep -q [@,#,$,%]
if [ $? -eq 0 ]; then
echo "Cool this is a Strong Password..."
else
echo ""
echo "Weak Password!!!..."
echo "Password must contain Numbers, Uppercase, Lowecase, and Special Character"
fi
else
echo ""
echo "Weak Password!!!..."
echo "Password must contain Numbers, Uppercase, Lowecase, and Special Character"
fi
else
echo ""
echo "Weak Password!!!..."
echo "Password must contain Numbers, Uppercase, Lowecase, and Special Character"
fi
else
echo ""
echo "Weak Password!!!..."
echo "Password must contain Numbers, Uppercase, Lowecase, and Special Character"
fi
else
echo ""
echo "Weak Password!!!... Password length must be >= 8"
fi
echo ""
~
~
"password.sh" 41L, 952C
```