# Operating System
# LAB - 01

**5.**

who > myfile1 | less myfile1
or
who > myfile1 | more myfile1

**6.**

who > myfile2 | (date ; more myfile2)

**7.**

sed -E 's/(\S+)\s+(\S+)/\2 \1/' myfile
where **(\S+)** \ s+ **(\S+)** . Means either first **(\S+)** and one character plus (s+) **(\S+)** will be replaced \2 to \1 .

**8.**

- cat > myfile.sh
  echo “\HELLO WORLD”

  sh myfile.sh
  which prints \HELLO WORLD.

- cat > myfile.c

  #include<stdio.h>
  int main(){
  printf(“HELLO WORLD!\n”);
  return 0;
  }

  gcc myfile.c -o myfile
  ./myfile
  Prints

HELLO WORLD!

(time)

@ time gcc myfile.c -o myfile

real 0m0.051s
user 0m0.034s
sys 0m0.018s

@ time sh script.sh
or time ./myfile

both just prints. Running time almost same.

@ time sh script.sh

real 0m0.002s
user 0m0.001s
sys 0m0.002s

@ time ./myfile

real 0m0.002s
user 0m0.000s
sys 0m0.002s

**9.**

touch script.sh | gedit script.sh
Then edit that with

```
echo "Enter a file name: "
read string
if test -f $string
then echo "It is a file"
elif test -d $string
then echo "It is a directory"
else
echo "Something else or not exists"
fi
```

**10.**

```
echo -n "Enter the file: "
read fileName
if [ ! -f$fileName ]
then
echo "File not exists"
exit 1
fi
tr '[a-z]' '[A-Z]'<$fileName
```

**11.**

```
echo "enter the user name : \c"
read usr
tuser=`who | tr -s " " | head -1 | cut -d " " -f1`
if [ "$tuser" = "$usr" ]
then
tm=`who | tr -s " " | head -1 | cut -d " " -f4`
uhr=`echo $tm | cut -d ":" -f1`
umin=`echo $tm | cut -d ":" -f2`
shr=`date "+%H"`
smin=`date "+%M"`
if [ $smin -lt $umin ]
then
shr=`expr $shr - 1`
smin=`expr $smin + 60`
fi
h=`expr $shr - $uhr`
m=`expr $smin - $umin`
echo "user name : $usr"
echo "login period : $h : $m"
else
echo "Invalid User"
fi
```

**12.**

```
echo "enter the filename: "
read filename
echo "enter the starting line number: "
read s
echo "enter the ending line number: "
read n
sed -n $s,$n\p $filename | cat > newline
cat newline
```

**13.**

```
echo "Enter the file: "
read fileName
echo "The file is:\n "
cat $fileName
echo "Enter the word: "
read word
sed -ie /$word/d $fileName
echo "After deletation the file is:\n"
cat $fileName
```

**14.**

Extract a sub-string from a given string :

```
echo "Enter a string: "
read str
echo "Enter a position you want a substring starts: "
read start
echo "Enter the substring length: "
read length
echo ${str:start:length}
```

Lenght of a given string :

```
echo "Enter a string: "
read string
echo ${#string}
```