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 .

or

echo "Enter your filename"
read filename
sed -E "s/(\S+)\s+(\S+)/\2\1/" $filename
or
sed -E "s/(\S+)\s+(\S+)/\2\ \1/" $filename

{ai khane space ashbe}

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.
echo "Enter your filename"
read filename
if test -f $filename
    then echo "This is file!"
elif test -d $filename
    then echo "This is directory!"
else
    echo "File doesn't exits!"
fi



10.
echo -n "Enter your filename"
read filename
if [ ! -f $filename]
    then echo "File doesn't exits"
exit 1
fi
command
tr '[a-z]' '[A-Z]' < $filename


11.
echo "Enter your user Name ?"
read user
last $user


12.
echo -n "Enter your filename"
read filename
echo "Enter the starting line?"
read starting
echo "Enter the ending line?"
read ending
sed -n $starting,$ending\p $filename
sed -n "$starting, $ending p" $filename
or
sed -n $starting,$ending\p $filename | cat > newfile
more newfile

13.
echo -n "Enter your filename"
read filename
echo "Enter your pattern"
read pattern
```

```
sed -i "/$pattern/d" $filename

or
sed -i "/$pattern/d" $filename | cat > newfile
more newfile

{-i kaj na krle baad; mane hocce case-insensitive}

14.a

echo "Enter a String"
read string
echo "Enter sub-string starting point to cut?"
read spoint
echo "Enter sub-string length?"
read length
echo {string:spoint:length}

14.b
echo "Enter a String"
read string
echo ${#string}


THE END
```