OS LAB ASSIGNMENT 1

Yashica Patodia: 19CS10067
Ishan Goel: 19CS30052

Question 1A

In Assgn1_1a_56_19CS10067_19CS10052.sh we are using the standard method to find the prime divisors of a number We are iterating from 2 to $1 in the for loop and in the while loop we keep dividing the number until the remainder is a non-zero number.
The **set** command is used to set the variable.

Word count:11

```
for((i=2;i<=$1;i++));do while [ $(($1%$i)) == 0 ];do printf "$i "&&set $(($1/$i));done;done
```

Question 1B

In Assgn1_1b_56_19CS10067_19CS10052.sh first we are creating a folder using **mkdir** command then in the for loop iterating through all the sub-directories and files in 1.b.files using the ls command.In the for loop we are sorting the contents using the **sort** command with numeric data **-n** then writing it to a folder 1.b.files.out.In the second part we are using the **cat** command to read all .txt files in that folder and sorting it in increasing order then using the **uniq** command with **-c** flag which tell how many times a line is repeated.The **awk** command is used to print according to the format <num> <freq> which is then written in the 1.b.out.txt using the greater than operator.

Word count: 16

```
mkdir 1.b.files.out;for file in `ls 1.b.files`;do sort -n 1.b.files/$file -o "1.b.files.out/$file";done;cat 1.b.files.out/*.txt|sort -n|uniq -c|awk '{print $2"\t"$1}'>1.b.out.txt
```

Question 1C

In Assgn1_1c_56_19CS10067_19CS10052.sh (Without slash) first we are finding the folders which are at a depth one and which needs to be deleted later. Then we are iterating through all the files inside the folder provided as the command-line argument and storing it in the filename.txt.Next we are separating all the files which are without extension using **grep**.And using **xargs** to move all those files to nil folder.Then finally moving all the files to their respective directory according to the extension.

Word Count=46

```
folders_to_del=$(find $1 -mindepth 1 -maxdepth 1 -type d); for filename in
$(find $1 -type f);do echo $filename>>filename.txt;ext="Nil";if [[ $filename =~
"." ]];then ext=${filename#${filename%.*}.};fi;echo
$ext>>extensions.txt;done;mkdir $1/Nil;grep -v ".*\..*" filename.txt|xargs mv
-t $1/Nil;ext_list=$(sort extensions.txt|uniq);for extension in $ext_list;do
mkdir -p $1/$extension;grep -e "\.$extension$" filename.txt|xargs mv -t
$1/$extension;done;rm -r $folders_to_del
```

Question 1D

In Assgn1_1d_56_19CS10067_19CS10052.sh we are creating a folder files_mod using **mkdir** and iterating through all the files in $1 which is given as input.For each of the files we are using the **sed** command to **subsitute all the spaces with a comma globally** then **counting** the line number using **nl commnd and adding ,** after that and finally writing it to the files_mode/$file.

Word count=9

```
mkdir files_mod;for file in `ls $1`;do sed "s/\s/,/g"<$1$file|nl
-s,>"files_mod/$file";done
```

Question 1E

In Assgn1_1e_56_19CS10067_19CS10052.sh we are using **export** to set the environment variables and using curl to send a **GET** request to example.com and storing the contents in .html file. Then we use {} to incorporate both the urls for ip and headers.For the third part we iterate through the for loop after splitting the string and use jq to print the keys.

Word count:30

```
VERBOSE=$2
function logerrr()
{
    [ "$VERBOSE" == "-v"] && printf "\n$@\n"
}
export REQ_HEADERS=Accept,Host;logerr "REQ_HEADERS environment variable
set";curl example.com -o example.html;logerr "Fetched webpage and created
example.html";curl {ip,headers}.jsontest.com;logerr "Received IP address and
the headers \n";arr=(${REQ_HEADERS//,/ });for i in ${arr[*]};do echo `curl -s
headers.jsontest.com|jq ".$i"`;done;logerr "Parsed the JSON response and
displayed the headers in REQ_HEADERS separated by comma \n";for filename in
$(ls $1);do if jq empty $1$filename;then echo $filename>>valid.txt;else echo
$filename>>invalid.txt;fi;done;logerr "Checked for valid and invalid json
scripts \n"
```

Question 1F

In Assgn1_1f_56_19CS10067_19CS10052.sh this script we use **awk** to take the second column $2 then convert it into lower case then to count the unique words we use **uniq -c** then sort it in reverse order using **-nr** and then use awk to print according to the format given.

Word count:12

```
awk "{print \$$2}" $1|awk '{print tolower($0)}'|sort|uniq -c|sort -nr|awk
"{print \$2, \$1 >\"1f_output_$2_column.freq\"}"
```

Question 1G

In Assgn1_1g_56_19CS10067_19CS10052.sh we use **shuf** command to print the numbers from 1-1500 then use **sed** to replace all the \n by ",". The sed command works line by line hence it doesnt know how to see new line and **-z** command resolves that**.-F** is used for separator.Then we use grep command to check **-q** whether $3 is yes or no.

Word count: 18

```
shuf -i 1-1500|sed -z 's/\n/,/g'|sed 's/,/\n/10;P;D' > $1;awk -F"," "{print
\$$2}" $1|grep -q $3&&{ echo "YES";}||{ echo "NO";}
```