

SHELL SCRIPTING PRACTICE QUESTION

- ## 1. Script for addition of two real numbers

[illegible]

```
rohan@rohan-virtual-machine: ~/shell
```

```
rohan@rohan-virtual-machine:~/shell$ bash real.sh
enter number 1
4.28
enter number 2
1.21
addition of two real number is : 5.49
rohan@rohan-virtual-machine:~/shell$
```

- ## 2. Script for arithmetic calculator using command line arguments

4 5 6

7 8 9 10

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh

echo "ENTER number of lines you want in patter"
read n
i=1
j=1
while [ $j -le $n ]
do
    for (( k=1; k<=j; k++ ))
    do
        echo -n "$i "
        i=$((i+1))
    done

    echo ""
    j=$((j+1))
done

~
~
~
~
~
~
"pat2.sh" 17 lines, 196 bytes
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ vi pat2.sh
rohan@rohan-virtual-machine:~/shell$ bash pat2.sh
ENTER number of lines you want in patter
6
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
16 17 18 19 20 21
rohan@rohan-virtual-machine:~/shell$
```

5. Script to compare larger integer values from a 'n' number of arguments using command line arguments

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
max=0

for i in "$@"
do
    if [ "$i" -gt "$max" ]
    then
        max=$i
    fi
done
echo "MAX Element is: $max"
```

"largest.sh" 13 lines, 134 bytes

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ vi largest.sh
rohan@rohan-virtual-machine:~/shell$ bash largest.sh 1 3 8 6 5 7 9 2
Max Element is: 9
rohan@rohan-virtual-machine:~/shell$
```

- ### 6. Script to print a given number in reverse order

7. script to read 'n' and generate Fibonacci numbers $\leq n$


```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
for i in $(seq 1 8)
do
    for j in $(seq 1 8)
    do
        S=$((i+j)%2)
        if [ $S -eq 0 ]
        then
            echo -n "0 "
        else
            echo -n "1 "
        fi
    done
    echo ""
done
"chess.sh" 17 lines 173 bytes
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ vi vhess.sh
rohan@rohan-virtual-machine:~/shell$ vi chess.sh
rohan@rohan-virtual-machine:~/shell$ bash chess.sh
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
rohan@rohan-virtual-machine:~/shell$
```

9. Script called say hello, which will print greetings based on time and to provide date information.


```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh

hour=`date +%H`
now=$(date)
if [ $hour -lt 12 ]
then
echo "GOOD MORNING ROHAN"
echo "Current date is $now"
elif [ $hour -le 16 ]
then
echo "GOOD AFTERNOON ROHAN"
echo "Current date is $now"
elif [ $hour -le 20 ]
then
echo "GOOD EVENING ROHAN"
echo "Current date is $now"
else
echo "GOOD NIGHT ROHAN"
echo "Current date is $now"
fi
~
~
~
'greet.sh' 20 lines, 349 bytes
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ vi greet.sh
rohan@rohan-virtual-machine:~/shell$ bash greet.sh
GOOD MORNING ROHAN
Current date is Thursday 20 April 2023 11:22:38 AM IST
rohan@rohan-virtual-machine:~/shell$
```

10. Script to locks file permissions for a particular directory for groups and others



rohan@rohan-virtual-machine: ~/shell

```
# /bin/bash/sh
```

```
file=$1
```

```
ls -la "${file}"
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

```
~
```

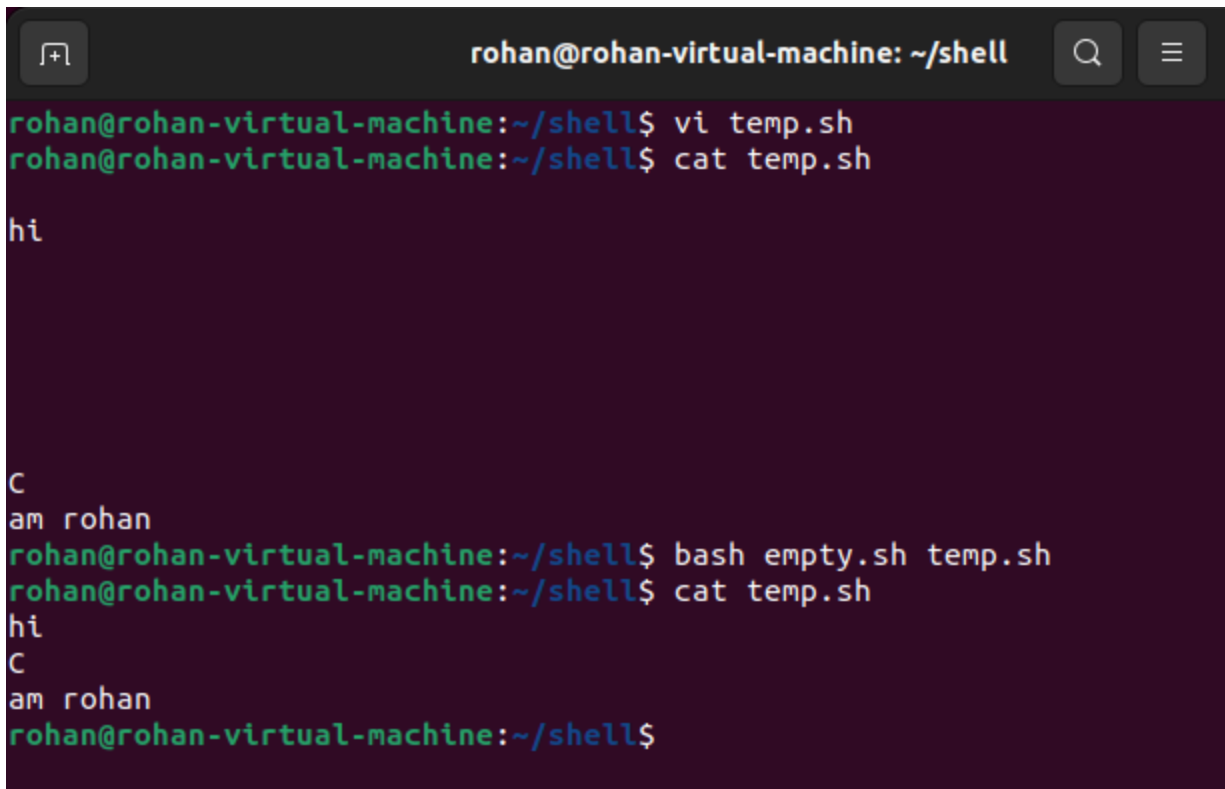
```
~
```

```
~
```

```
~
```

```
~
```

```
"per.sh" 4 lines, 41 bytes
```


A terminal window titled "rohan@rohan-virtual-machine: ~/shell" with search and menu icons. The terminal shows a user editing a file named "temp.sh" in vi mode, saving it with "C", and then running it with "bash empty.sh temp.sh". The script prints "hi" twice. The user then exits vi with "C" and "am rohan".

```
rohan@rohan-virtual-machine:~/shell$ vi temp.sh
rohan@rohan-virtual-machine:~/shell$ cat temp.sh

hi

C
am rohan
rohan@rohan-virtual-machine:~/shell$ bash empty.sh temp.sh
rohan@rohan-virtual-machine:~/shell$ cat temp.sh
hi
C
am rohan
rohan@rohan-virtual-machine:~/shell$
```

12. Script to perform arithmetic operation on digits of a given number depending upon the operator.

```
rohan@rohan-virtual-machine: ~/shell
#!/bin/bash

if [ $# -eq 0 ];
then
    echo "Usage: $0 string"
exit 1
fi

number=${1%[-+*/]}
operator=${1:${#number}:1}

if [[ ! $operator =~ [-+*/] ]];
then
    echo "Error: Invalid operator."
exit 1
fi

result=0
for ((i=0; i<${#number}; i++));
do
    digit=${number:$i:1}
    case $operator in
        +) result=$((result + digit)) ;;
        -) result=$((result - digit)) ;;
        \*) result=$((result * digit)) ;;
        /) result=$((result / digit)) ;;
    esac
done
echo "The $operator of the digits in $number is $result"
~
~
~
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ vi Q8.sh
rohan@rohan-virtual-machine:~/shell$
rohan@rohan-virtual-machine:~/shell$ bash Q8.sh 12345+
The + of the digits in 12345 is 5
rohan@rohan-virtual-machine:~/shell$ bash Q8.sh 12345
Error: Invalid operator.
rohan@rohan-virtual-machine:~/shell$ bash Q8.sh 12345-
The - of the digits in 12345 is -5
rohan@rohan-virtual-machine:~/shell$
```

Q13. Script to print the length of each and every string using arrays

```
rohan@rohan-virtual-machine: ~/shell
```

```
rohan@rohan-virtual-machine:~/shell$ touch Q10.sh
rohan@rohan-virtual-machine:~/shell$ bash Q10.sh hello, i am rohan
String lenght=
hello, = -6
i = -1
am = -2
rohan = -5
rohan@rohan-virtual-machine:~/shell$
```

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
str_arr=($*)
echo "String lenght="
for i in "${str_arr[@]}"
do
    echo "$i = -${#i}"
done
```

"Q10.sh" 9 lines, 107 bytes

Q14. Script to sort a given number in ascending or descending order.

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh

while getopts "a:d:" opt;
do
    case ${opt} in
        a) order="ascending";;
        d) order="descending";;
        esac
    done
    shift $((OPTIND -1))
    args=("$@")
    if [[ "$order" == "ascending" ]];
    then
        sorted_args=($(printf '%s\n' "${args[@]}" | sort -n))
    elif [[ "$order" == "descending" ]];
    then sorted_args=($(printf "%s\n" "${args[@]}" | sort -nr))
    fi
    echo "The $order order of the array ts: "
    echo "${sorted_args[@]}"
    ~
    ~
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ bash Q12.sh -a 96 12 85 36 25 1 9 14 14 11
The ascending order of the array ts:
1 9 11 12 14 14 25 36 85
rohan@rohan-virtual-machine:~/shell$ bash Q12.sh -d 96 12 85 36 25 1 9 14 14 11
The descending order of the array ts:
85 36 25 14 14 12 11 9 1
rohan@rohan-virtual-machine:~/shell$
```

Q15. Script to print the following:

- Currently logged users
- Your shell directory
- Home directory
- OS name & version
- Current working directory
- Number of users logged in

- Show all available shells in your system
- Hard disk information
- CPU information.
- Memory information.
- File system information.
- Currently running process

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
echo "System information "

echo "1. Currently logged users : $who"
echo "2. Your shell directory $SHELL"
echo "Your home directory is $HOME"
echo "OS name & version: $(uname -a) "
echo "Current working directory: $pwd"
echo "Number of users logged in: $( who | wc -l ) "
echo "7. Show all available shells in your system $(cat /etc/shells)"
echo "8. Hard disk information $(df -h)"
echo "9. CPU information $lscpu"
echo "10. Memory information $(free -h)"
echo "11. File system information $(df -T)"
echo "12. Currently running process $ps"

~
~
~
```



```

rohan@rohan-virtual-machine:~/shell$ vi Q13.sh
rohan@rohan-virtual-machine:~/shell$ bash Q13.SH
bash: Q13.SH: No such file or directory
rohan@rohan-virtual-machine:~/shell$ bash Q13.sh
System information
1. Currently logged users :
2. Your shell directory /bin/bash
Your home directory is /home/rohan
OS name & version: Linux rohan-virtual-machine 5.19.0-40-generic #41-22.04.1-Ubuntu SMP PREEMPT_DYNAMIC Fri Mar 31 16:00:14 UTC 2 x86_64 x86_64 x86_64 GNU/Linux
Current working directory:
Q13.sh: line 10: wc-l: command not found
Number of users logged in:
7. Show all available shells in your system # /etc/shells: valid login shells
/bin/sh
/bin/bash
/usr/bin/bash
/bin/rbash
/usr/bin/rbash
/usr/bin/sh
/bin/dash
/usr/bin/dash
8. Hard disk information


| Filesystem | Size | Used | Avail | Use% | Mounted on                            |
|------------|------|------|-------|------|---------------------------------------|
| tmpfs      | 195M | 1.7M | 193M  | 1%   | /run                                  |
| /dev/sda3  | 20G  | 13G  | 6.0G  | 68%  | /                                     |
| tmpfs      | 971M | 0    | 971M  | 0%   | /dev/shm                              |
| tmpfs      | 5.0M | 4.0K | 5.0M  | 1%   | /run/lock                             |
| /dev/sda2  | 512M | 6.1M | 506M  | 2%   | /boot/efi                             |
| tmpfs      | 195M | 100K | 194M  | 1%   | /run/user/1000                        |
| /dev/sr0   | 145M | 145M | 0     | 100% | /media/rohan/CDROM                    |
| /dev/sr1   | 4.6G | 4.6G | 0     | 100% | /media/rohan/Ubuntu 22.04.2 LTS amd64 |


9. CPU information
10. Memory information


|       | total | used  | free  | shared | buff/cache | available |
|-------|-------|-------|-------|--------|------------|-----------|
| Mem:  | 1.9Gi | 77Mi  | 15Mi  | 311Mi  | 221Mi      |           |
| Swap: | 2.1Gi | 254Mi | 1.8Gi |        |            |           |


11. File system information


| Filesystem | Type    | 1K-blocks | Used     | Available | Use% | Mounted on                            |
|------------|---------|-----------|----------|-----------|------|---------------------------------------|
| tmpfs      | tmpfs   | 198720    | 1728     | 196992    | 1%   | /run                                  |
| /dev/sda3  | ext4    | 19946096  | 12682256 | 6225300   | 68%  | /                                     |
| tmpfs      | tmpfs   | 993588    | 0        | 993588    | 0%   | /dev/shm                              |
| tmpfs      | tmpfs   | 5120      | 4        | 5116      | 1%   | /run/lock                             |
| /dev/sda2  | vfat    | 524252    | 6216     | 518036    | 2%   | /boot/efi                             |
| tmpfs      | tmpfs   | 198716    | 100      | 198616    | 1%   | /run/user/1000                        |
| /dev/sr0   | iso9660 | 148004    | 148004   | 0         | 100% | /media/rohan/CDROM                    |
| /dev/sr1   | iso9660 | 4812096   | 4812096  | 0         | 100% | /media/rohan/Ubuntu 22.04.2 LTS amd64 |


12. Currently running process
rohan@rohan-virtual-machine:~/shell$

```

Q16. Script to rename a file/directory replaced by lower/upper case letters.



rohan@rohan-virtual-machine: ~/shell

```
#!/bin/bash

echo "Enter the directory path:"
read directory

if [[ "$file" == "${file^^}" ]];
then
    for file in "$directory"/*;
    do
        if [[ -f "$file" ]];
        then
            mv "$file" "${file,,}"
        elif [[ -d "$file" ]];
        then
            mv "$file" "${file,,}"
        fi
    done
elif [[ "$file" == "${file,,}" ]];
then
for file in "$directory"/*;
do
    if [[ -f "$file" ]];
    then
        mv "$file" "${file^^}"
    elif [[ -d "$file" ]];
    then
        mv "$file" "${file^^}"
    fi
done
fi
echo "After running the script:"
ls "$directory"

~
~
~
```

```

rohan@rohan-virtual-machine:~$ cd shell
rohan@rohan-virtual-machine:~/shell$ vi Q14.sj
rohan@rohan-virtual-machine:~/shell$ vi Q14.sh
rohan@rohan-virtual-machine:~/shell$ bash Q14.sh
Enter the directory path:
/home/rohan/shell
mv: '/home/rohan/shell/:' and '/home/rohan/shell/:' are the same file
mv: '/home/rohan/shell/@' and '/home/rohan/shell/@' are the same file
mv: '/home/rohan/shell/1' and '/home/rohan/shell/1' are the same file
mv: '/home/rohan/shell/addition.sh' and '/home/rohan/shell/addition.sh' are the same file
mv: '/home/rohan/shell/arr2.sh' and '/home/rohan/shell/arr2.sh' are the same file
mv: '/home/rohan/shell/arr3.sh' and '/home/rohan/shell/arr3.sh' are the same file
mv: '/home/rohan/shell/arr.sh' and '/home/rohan/shell/arr.sh' are the same file
mv: '/home/rohan/shell/cal.sh' and '/home/rohan/shell/cal.sh' are the same file
mv: '/home/rohan/shell/chess.sh' and '/home/rohan/shell/chess.sh' are the same file
mv: '/home/rohan/shell/content.sh' and '/home/rohan/shell/content.sh' are the same file
mv: '/home/rohan/shell/div.sh' and '/home/rohan/shell/div.sh' are the same file
mv: '/home/rohan/shell/empty.sh' and '/home/rohan/shell/empty.sh' are the same file
mv: '/home/rohan/shell/exe.sh' and '/home/rohan/shell/exe.sh' are the same file
mv: '/home/rohan/shell/fib.sh' and '/home/rohan/shell/fib.sh' are the same file
mv: '/home/rohan/shell/for2.sh' and '/home/rohan/shell/for2.sh' are the same file
mv: '/home/rohan/shell/for.sh' and '/home/rohan/shell/for.sh' are the same file
mv: '/home/rohan/shell/greet.sh' and '/home/rohan/shell/greet.sh' are the same file
mv: '/home/rohan/shell/if_else.sh' and '/home/rohan/shell/if_else.sh' are the same file
mv: '/home/rohan/shell/if.sh' and '/home/rohan/shell/if.sh' are the same file
mv: '/home/rohan/shell/largest.sh' and '/home/rohan/shell/largest.sh' are the same file
mv: '/home/rohan/shell/leap.sh' and '/home/rohan/shell/leap.sh' are the same file
mv: '/home/rohan/shell/less.sh' and '/home/rohan/shell/less.sh' are the same file
mv: '/home/rohan/shell/mod.sh' and '/home/rohan/shell/mod.sh' are the same file
mv: '/home/rohan/shell/mul.sh' and '/home/rohan/shell/mul.sh' are the same file
mv: '/home/rohan/shell/no_file.sh' and '/home/rohan/shell/no_file.sh' are the same file
mv: '/home/rohan/shell/pat1.sh' and '/home/rohan/shell/pat1.sh' are the same file
mv: '/home/rohan/shell/pat2.sh' and '/home/rohan/shell/pat2.sh' are the same file
mv: '/home/rohan/shell/per.sh' and '/home/rohan/shell/per.sh' are the same file
mv: '/home/rohan/shell/prime.sh' and '/home/rohan/shell/prime.sh' are the same file
mv: '/home/rohan/shell/q10.sh' and '/home/rohan/shell/q10.sh' are the same file
mv: '/home/rohan/shell/q12.sh' and '/home/rohan/shell/q12.sh' are the same file
mv: '/home/rohan/shell/q13.sh' and '/home/rohan/shell/q13.sh' are the same file
mv: '/home/rohan/shell/q14.sh' and '/home/rohan/shell/q14.sh' are the same file
mv: '/home/rohan/shell/q15' and '/home/rohan/shell/q15' are the same file
mv: '/home/rohan/shell/q8.sh' and '/home/rohan/shell/q8.sh' are the same file
mv: '/home/rohan/shell/real.sh' and '/home/rohan/shell/real.sh' are the same file
mv: '/home/rohan/shell/rev.sh' and '/home/rohan/shell/rev.sh' are the same file
mv: '/home/rohan/shell/sub.sh' and '/home/rohan/shell/sub.sh' are the same file
mv: '/home/rohan/shell/switch1.sh' and '/home/rohan/shell/switch1.sh' are the same file
mv: '/home/rohan/shell/switch2.sh' and '/home/rohan/shell/switch2.sh' are the same file
mv: '/home/rohan/shell/switch.sh' and '/home/rohan/shell/switch.sh' are the same file
mv: '/home/rohan/shell/temp.sh' and '/home/rohan/shell/temp.sh' are the same file
mv: '/home/rohan/shell/until1.sh' and '/home/rohan/shell/until1.sh' are the same file

```

Q17. Script to rename current working directory with given name

```
rohan@rohan-virtual-machine: ~/shell

#!/bin/bash
current_dir=$(basename "$PWD")
new_name=$1
echo "$current_dir"
mv "$pwd" "$new_name"

rohan@rohan-virtual-machine:~/shell$ bash Q15.sh SHELL
shell
shell SHELL
rohan@rohan-virtual-machine:~/shell$
```

Q18. cript to rename all .jpg files by replacing prefix which is given by user

```
rohan@rohan-virtual-machine: ~/shell

#!/bin/bash/sh

echo "Enter prefix which u want"
read n

for i in *.jpg;
do
    mv "$i" "${n}_${file#*_}"
done

~
~
~
~
~
```

Q19. Script to print contents of file from given line number to next given number of lines.


```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh

Longest=$(awk -F: '{print length($1) " " $1}' /etc/passwd | sort -n | tail -n 1 | awk '{print $2}')

shortest=$(awk -F: '{print length($1) " " $1}' /etc/passwd | sort -n | head -n 1 | awk '{print $2}')

echo "The user with longest name is $Longest"
echo "The user with shortest name is $shortest"
~
~
~
~
~
~
```

```
rohan@rohan-virtual-machine:~/shell$ bash Q18.sh
The user with longest name is gnome-initial-setup
The user with shortest name is lp
rohan@rohan-virtual-machine:~/shell$
```

Q21. Script to delete all the .swp files found in your system or directory.

```
rohan@rohan-vir:~$ # /bin/bash/sh
find . -type f -name '*.swp' -print
"Q19.sh" 3 lines, 51 bytes
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ vi Q19.sh
rohan@rohan-virtual-machine:~/shell$ bash Q19.sh
find: unknown predicate '-name*.swp'
rohan@rohan-virtual-machine:~/shell$
```

Q22. script for generating random 8-character passwords including alpha numeric characters.

```
rohan@rohan-virtual-machine: ~/shell
#!/bin/bash
function generate_password() {
local length=$1
local chars='abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'
for i in $(seq 1 $length);
do
echo -n "${chars: RANDOM%${#chars}:1}"
done
echo
}
echo "The generated 8 random passwords are:"
for i in $(seq 1 8);
do
generate_password 8
done
~
```

```
rohan@rohan-virtual-machine:~/shell$ vi Q20.sh
rohan@rohan-virtual-machine:~/shell$ bash Q20.sh
The generated 8 random passwords are:
mcXw8Rpn
2vr4VmVw
LPkEqcH0
MacTESye
zuYWhJsP
tBD4pPcN
khwwIEVI
d6BnUTcY
rohan@rohan-virtual-machine:~/shell$
```

Q23. Script to convert content of file lower to uppercase and upper to lowercase.



rohan@rohan-virtual-machine: ~/shell

```
# /bin/bash/sh

echo -n "Enter the file name"
read file_name

echo 1 = to upper case
echo 2 = to lower case

echo -n "to upper case or lower case? enter your choice"
read choice

case $choice in
    1) tr '[:lower:]' '[:upper:]' < $file_name ;;
    2) tr '[:upper:]' '[:lower:]' < $file_name ;;
    *) echo "INDia Choice"
       exit 1 ;;
esac

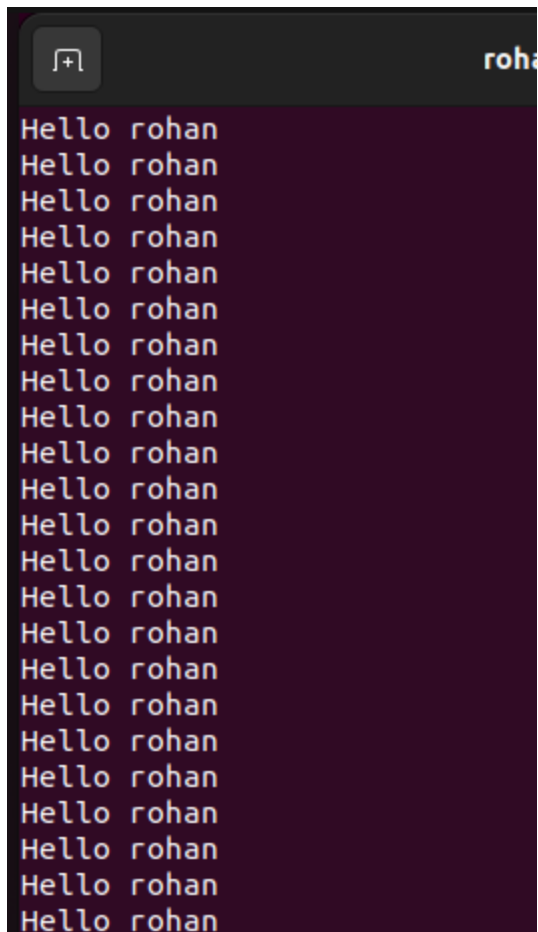
~
~
~
~
~

"Q22.sh" 18 lines, 326 bytes
```

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ bash Q22.sh
Enter the file name Q20.sh
1 = to upper case
2 = to lower case
to upper case or lower case? enter your choice 2
#!/bin/bash
function generate_wassword() {
local length=$1
local chars='abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789'
for i in $(seq 1 $length);
do
echo -n "${chars: RANDOM%${#chars}:1}"
done
echo
}
echo "The generated 8 random wasswords are:"
for i in $(seq 1 8);
do
generate_wassword 8
done
rohan@rohan-virtual-machine:~/shell$
```

Q24. Script to use pipes or redirection to create an infinite feedback loop.

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
while true;
do
    echo "Hello rohan"
done
~
~
~
~
~
```

A terminal window with a dark background. The title bar at the top shows a window icon and the text "rohan". The terminal contains 20 lines of text, each reading "Hello rohan".

```
rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan  
Hello rohan
```

Q26. Script to use a recursive function to print each argument passed to the function.

[illegible]

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ touch Q26.sh
rohan@rohan-virtual-machine:~/shell$ bash Q26.sh 1 2 5 9 4 s jk l 7 5
1
2
5
9
4
s
jk
l
7
5
```

Q27. Script to determine whether a given file system or mount point is mounted.

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh

if [ $# -eq 0 ];
then
    echo "Enter file"
    exit 1
fi

if mountpoint -q -- "$1";
then
    echo "$1 is mounter"
else
    echo "$1 is not mounter"
fi
~
~
```

```
rohan@rohan-virtual-machine:~/shell$ bash Q27.sh /dev/sda8
/dev/sda8 is not mounter
rohan@rohan-virtual-machine:~/shell$
```

Q28. Script that takes any number of directories as command-line arguments and then lists the contents of each of the directories.

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
dir=$1
ls -l "$dir"
~
~
~
~
~
~
~
```

```

rohan@rohan-virtual-machine:~/shell$ bash Q28.sh
ls: cannot access '': No such file or directory
rohan@rohan-virtual-machine:~/shell$ bash Q28.sh /home/rohan/shell
total 224
-rw-rw-r-- 1 rohan rohan  0 Apr 21 16:58 :
-rw-rw-r-- 1 rohan rohan 107 Apr 21 16:10 @
-rw-rw-r-- 1 rohan rohan  93 Apr 19 16:12 1
-rw-rw-r-- 1 rohan rohan  61 Apr 15 18:53 addition.sh
-rw-rw-r-- 1 rohan rohan 159 Apr 20 16:09 arr2.sh
-rw-rw-r-- 1 rohan rohan 195 Apr 20 16:17 arr3.sh
-rw-rw-r-- 1 rohan rohan 105 Apr 20 16:06 arr.sh
-rw-rw-r-- 1 rohan rohan  66 Apr 19 22:42 cal.sh
-rw-rw-r-- 1 rohan rohan 173 Apr 19 23:27 chess.sh
-rw-rw-r-- 1 rohan rohan  87 Apr 19 16:23 content.sh
-rw-rw-r-- 1 rohan rohan  77 Apr 15 19:27 div.sh
-rw-rw-r-- 1 rohan rohan  34 Apr 21 15:49 empty.sh
-rw-rw-r-- 1 rohan rohan 144 Apr 19 16:00 exe.sh
-rw-rw-r-- 1 rohan rohan 139 Apr 19 23:21 fib.sh
-rw-rw-r-- 1 rohan rohan  80 Apr 15 19:40 for2.sh
-rw-rw-r-- 1 rohan rohan  71 Apr 20 15:14 for.sh
-rw-rw-r-- 1 rohan rohan 349 Apr 20 11:33 greet.sh
-rw-rw-r-- 1 rohan rohan 125 Apr 19 12:52 if_else.sh
-rw-rw-r-- 1 rohan rohan 108 Apr 20 15:16 if.sh
-rw-rw-r-- 1 rohan rohan 134 Apr 19 23:02 largest.sh
-rw-rw-r-- 1 rohan rohan 205 Apr 19 12:52 leap.sh
-rw-rw-r-- 1 rohan rohan  93 Apr 19 16:11 less.sh
-rw-rw-r-- 1 rohan rohan  72 Apr 15 19:32 mod.sh
-rw-rw-r-- 1 rohan rohan  67 Apr 15 19:19 mul.sh
-rw-rw-r-- 1 rohan rohan  0 Apr 19 15:48 no_file.sh

```

Q29. Script to locks file permissions for a particular directory for groups and others

```
rohan@rohan-virtual-machine: ~/shell
# /bin/bash/sh
ls -l "$1"
echo "File permissions are"

chmod 443 "$1"
chmod 111 "$1"

echo "Changed permission"

ls -l "$1"
~
~
~
~
~
```

```
rohan@rohan-virtual-machine:~/shell$ bash Q29.sh /home/rohan/shell/Q22.sh
-rw-rw-r-- 1 rohan rohan 326 Apr 21 18:25 /home/rohan/shell/Q22.sh
File permissions are
Changed permission
---x--x--x 1 rohan rohan 326 Apr 21 18:25 /home/rohan/shell/Q22.sh
rohan@rohan-virtual-machine:~/shell$
```

Q30. Script to count the number of users with user IDs between 500 and 10000 on the system

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ touch Q30.sh
rohan@rohan-virtual-machine:~/shell$ bash Q30.sh 200 2500
Number of users with user IDs between 500 and 10000: 0
rohan@rohan-virtual-machine:~/shell$ vi Q30.sh
rohan@rohan-virtual-machine:~/shell$
```

```
rohan@rohan-virtual-machine: ~/shell
#!/bin/bash

count=0

while IFS=':' read -r user_id rest
do
    if [[ $user_id -ge 500 && $user_id -le 10000 ]]
    then
        count=$((count+1))
    fi
done < /etc/passwd

echo "Number of users with user IDs between 500 and 10000: $count"
~
~
~
~
```

Q31. Script or each directory in the \$PATH variable, to display the number of executable files in that directory.



rohan@rohan-virtual-machine: ~/shell

```
rohan@rohan-virtual-machine:~/shell$ bash Q31.sh
```

```
Directory: /usr/local/sbin
```

```
Number of executable files: 0
```

```
Directory: /usr/local/bin
```

```
Number of executable files: 0
```

```
Directory: /usr/sbin
```

```
Number of executable files: 0
```

```
Directory: /usr/bin
```

```
Number of executable files: 0
```

```
Directory: /sbin
```

```
Number of executable files: 0
```

```
Directory: /bin
```

```
Number of executable files: 0
```

```
Directory: /usr/games
```

```
Number of executable files: 0
```

```
Directory: /usr/local/games
```

```
Number of executable files: 0
```

```
Directory: /snap/bin
```

```
Number of executable files: 0
```

```
Directory: /snap/bin
```

```
Number of executable files: 0
```

```
rohan@rohan-virtual-machine:~/shell$
```

```
rohan@rohan-virtual-machine: ~/shell
#!/bin/bash

IFS=':' read -ra dirs <<< "$PATH"

for dir in "${dirs[@]}";
do

    num_exec=$(find "$dir" -type f -perm +x | wc -l)

    echo "Directory: $dir"
    echo "Number of executable files: $num_exec"
    echo ""
done

~
~
~
~
```

Q34. Script to calculate the BMI.

```
rohan@rohan-virtual-machine: ~/shell
rohan@rohan-virtual-machine:~/shell$ touch Q34.sh
rohan@rohan-virtual-machine:~/shell$ bash Q34.sh
Enter your weight in kilograms:
5
Enter your height in meter:
9
Your BMI is .06
You are underweight.
rohan@rohan-virtual-machine:~/shell$
```

```
# /bin/bash/sh
echo "Enter your weight in kilograms:"
read weight

echo "Enter your height in meter:"
read height
# calculate BMI
bmi=$(echo "scale=2; $weight / ($height * $height)" | bc)

echo "Your BMI is $bmi"

# Determine weight status based on WHO guidelines
if (( $(echo "$bmi < 18.5" | bc -l) )); then
    echo "You are underweight."
elif (( $(echo "$bmi < 25" | bc -l) )); then
    echo "you have a healthy weight."
elif (( $(echo "$bmi < 30" | bc -l) )); then
    echo "You are overweight."
else
    echo "You are obese."
fi
```

"Q34.sh" 21 lines, 534 bytes