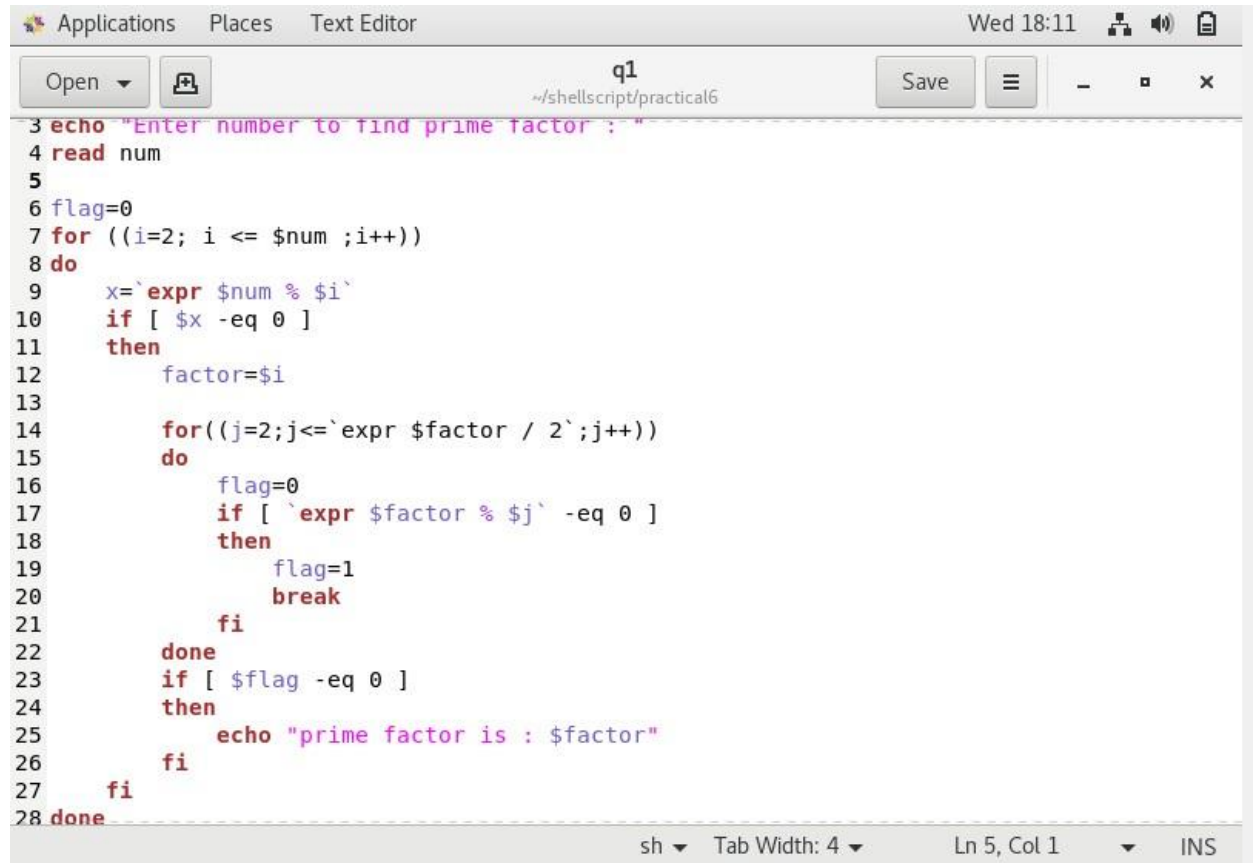


Practical 6

1. Write a shell script, which finds the prime factors of a given number.

Program:



The screenshot shows a text editor window titled 'q1' with the file path '~/shellscript/practical6'. The script is written in a shell language and is designed to find the prime factors of a user-input number. It starts by prompting the user to enter a number, then uses a loop to check for factors from 2 up to the number itself. For each factor found, it prints the factor and then recursively checks for factors of the remaining quotient. The script uses nested loops and conditional statements to ensure that only prime factors are identified.

```
3 echo "Enter number to find prime factor : "
4 read num
5
6 flag=0
7 for ((i=2; i <= $num ;i++))
8 do
9     x=`expr $num % $i`
10    if [ $x -eq 0 ]
11    then
12        factor=$i
13
14        for((j=2;j<=`expr $factor / 2`;j++))
15        do
16            flag=0
17            if [ `expr $factor % $j` -eq 0 ]
18            then
19                flag=1
20                break
21            fi
22        done
23        if [ $flag -eq 0 ]
24        then
25            echo "prime factor is : $factor"
26        fi
27    fi
28 done
```

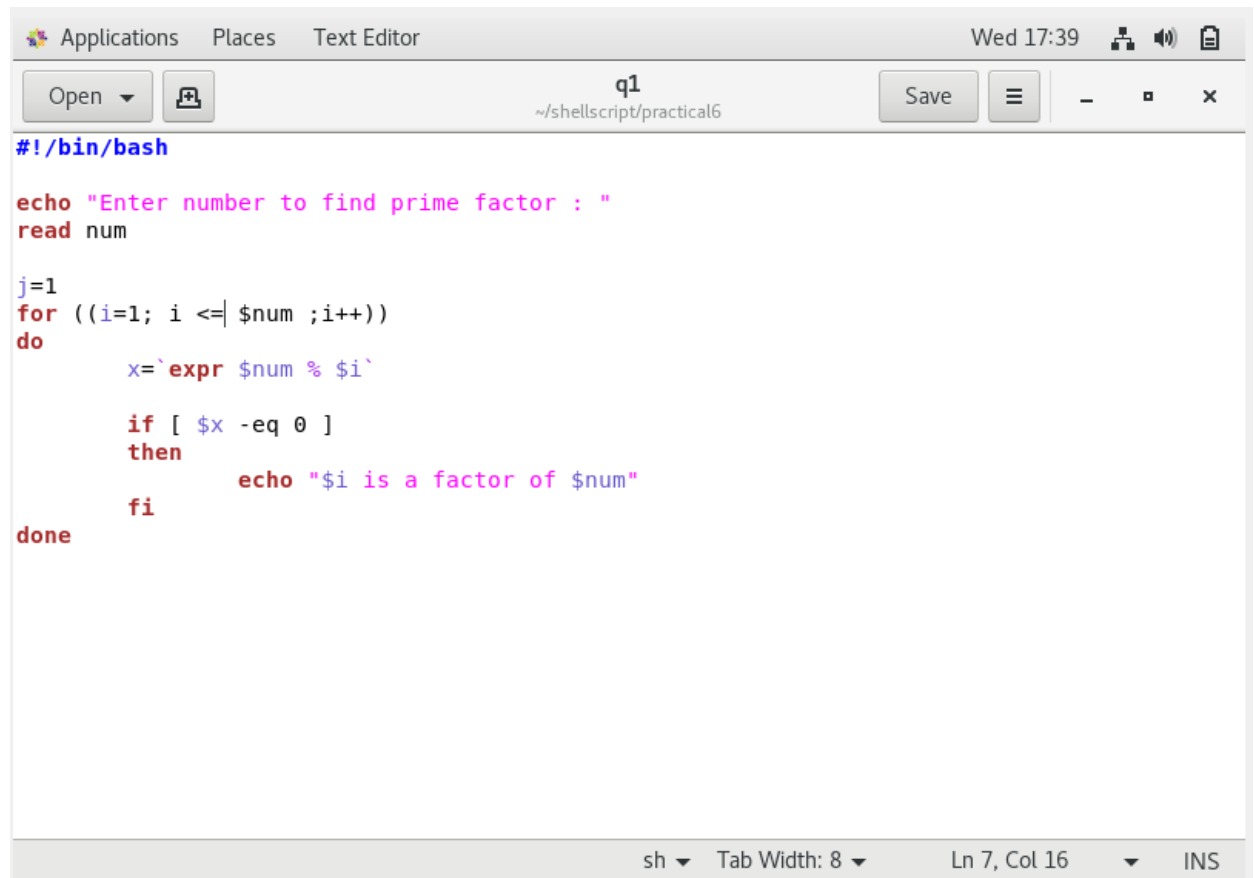
Output:

The screenshot shows a terminal window titled "19it006@19it006:~/p6". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the following sequence of commands and output:

```
[19it006@mylinuxvm practical6]$ gedit q1
[19it006@mylinuxvm practical6]$ sh q1
Enter number to find prime factor :
12
prime factor is : 2
prime factor is : 3
[19it006@mylinuxvm practical6]$
```

2. Write a shell script that accepts a positive integer value from the user, say 34, and prints out all the divisors of 34 as a list: Enter a positive integer: 34 The divisors of 34 are: 1, 2, 17, and 34.

Program:



The screenshot shows a text editor window titled 'q1' with the file path '~/shellscript/practical6'. The script content is as follows:

```
#!/bin/bash

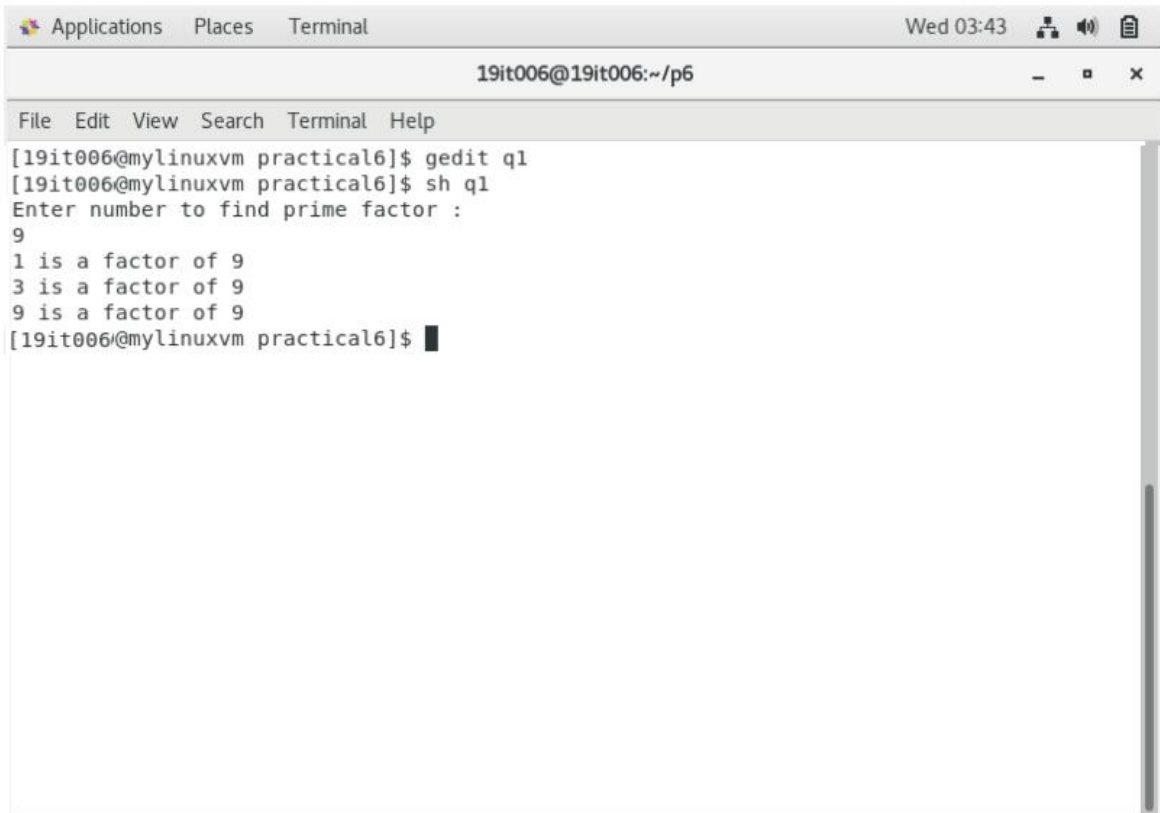
echo "Enter number to find prime factor : "
read num

j=1
for ((i=1; i <= $num ;i++))
do
    x=`expr $num % $i`

    if [ $x -eq 0 ]
    then
        echo "$i is a factor of $num"
    fi
done
```

The status bar at the bottom indicates 'sh', 'Tab Width: 8', 'Ln 7, Col 16', and 'INS'.

Output:

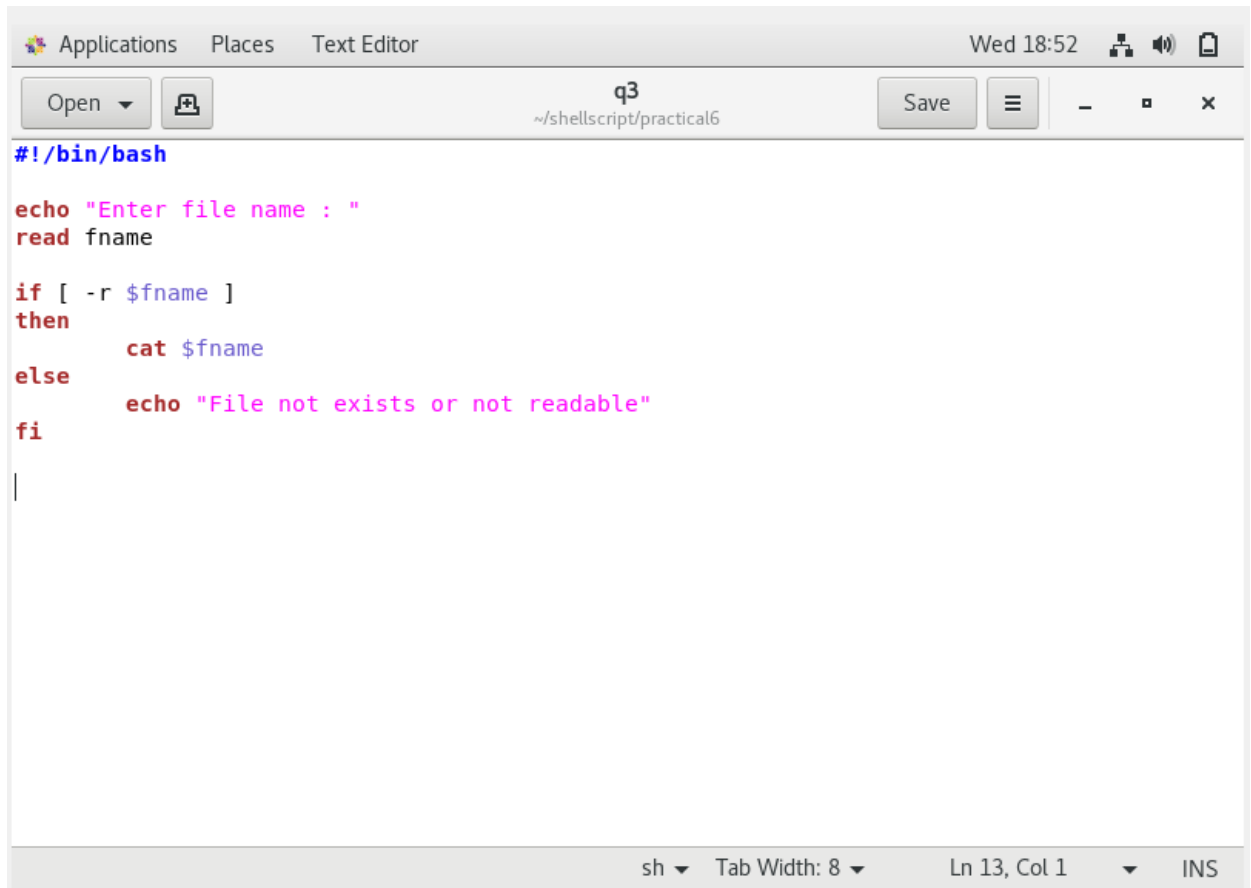


The screenshot shows a terminal window titled "Terminal" with a menu bar containing "File", "Edit", "View", "Search", "Terminal", and "Help". The window title bar also shows "Applications", "Places", and "Terminal". The terminal content shows the user running a shell script named "q1" using the "sh" command. The script prompts the user to enter a number to find its prime factors. The user enters "9", and the script outputs that 1, 3, and 9 are factors of 9. The terminal prompt is "[19it006@mylinuxvm practical6]\$".

```
19it006@19it006:~/p6
File Edit View Search Terminal Help
[19it006@mylinuxvm practical6]$ gedit q1
[19it006@mylinuxvm practical6]$ sh q1
Enter number to find prime factor :
9
1 is a factor of 9
3 is a factor of 9
9 is a factor of 9
[19it006@mylinuxvm practical6]$
```

3. Check whether a given file is readable or not. If it is readable, then display the file contents.

Program:



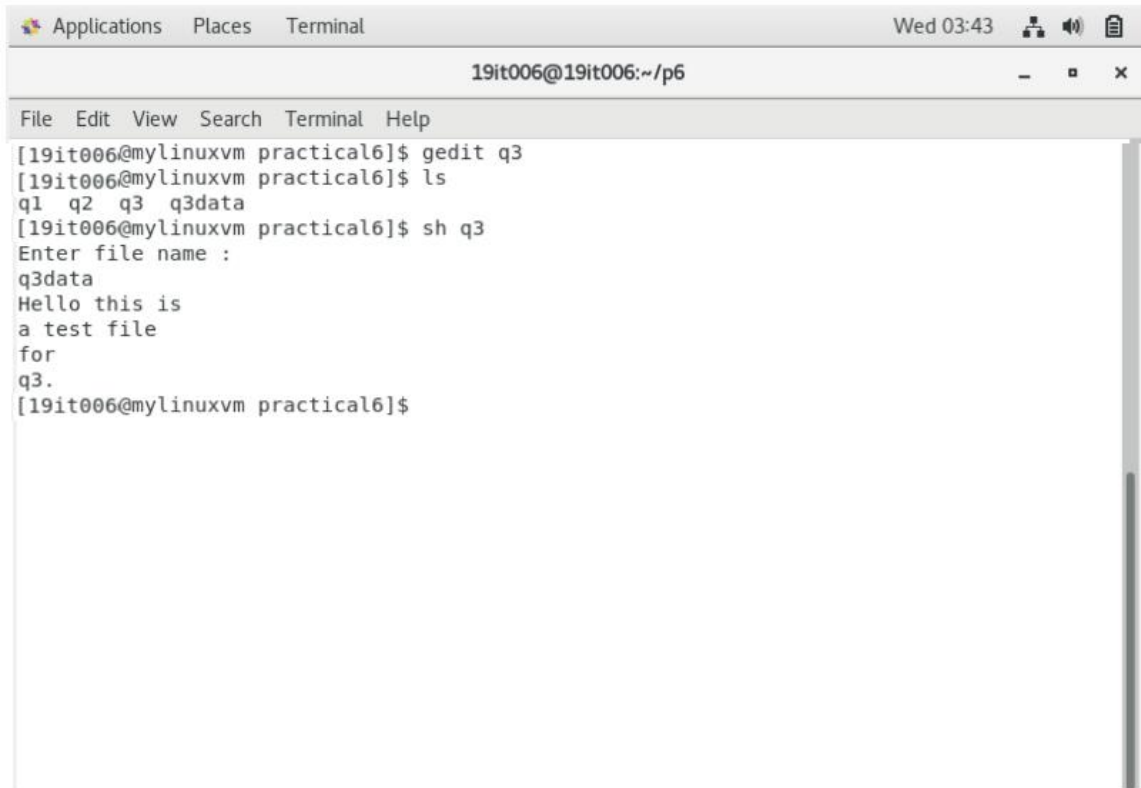
The screenshot shows a Linux desktop environment with a text editor window titled 'q3' at the path '~/shellscript/practical6'. The window has a menu bar with 'Applications', 'Places', and 'Text Editor'. The status bar at the bottom indicates 'sh', 'Tab Width: 8', 'Ln 13, Col 1', and 'INS'. The script content is as follows:

```
#!/bin/bash

echo "Enter file name : "
read fname

if [ -r $fname ]
then
    cat $fname
else
    echo "File not exists or not readable"
fi
```

Output:

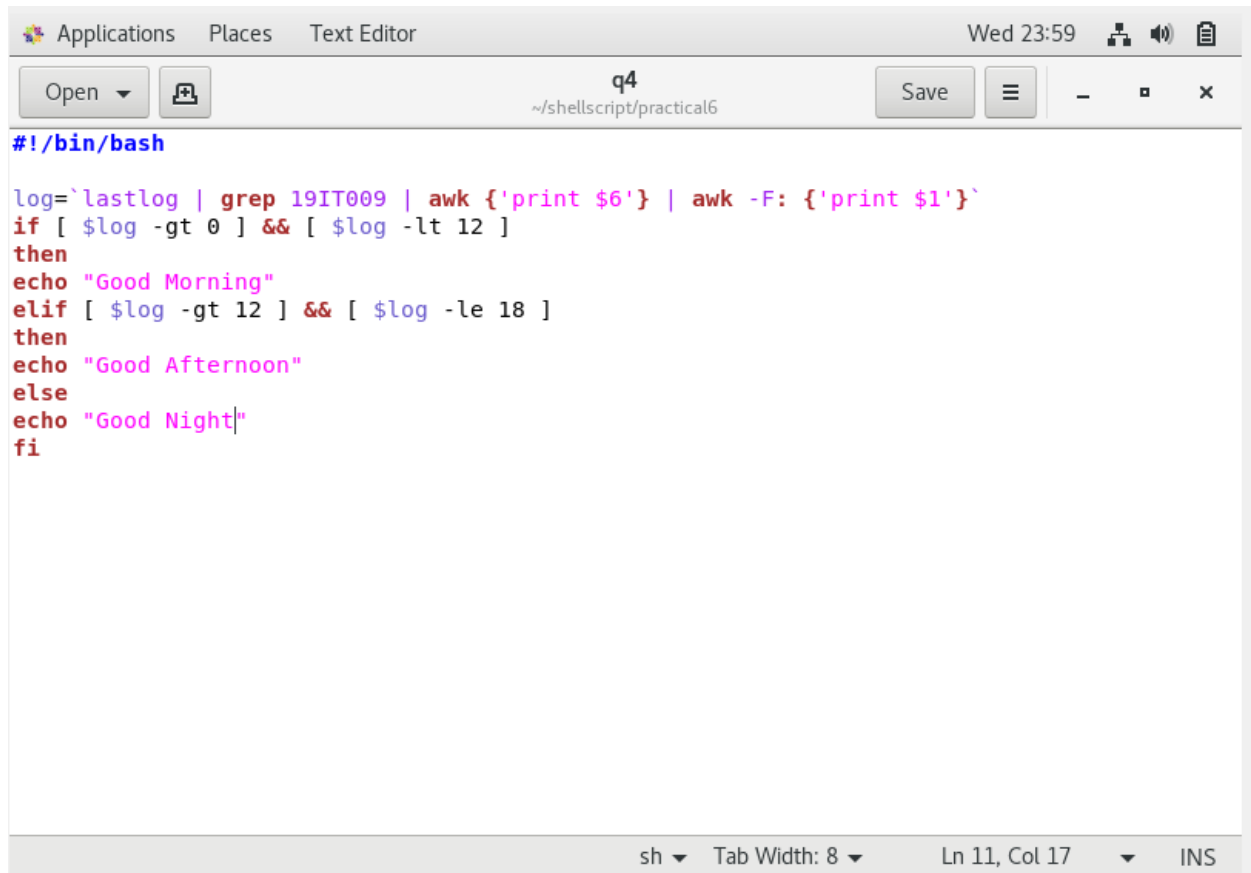


The image shows a terminal window titled "19it006@19it006:~/p6". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal output shows the following sequence of commands and responses:

```
[19it006@mylinuxvm practical6]$ gedit q3
[19it006@mylinuxvm practical6]$ ls
q1 q2 q3 q3data
[19it006@mylinuxvm practical6]$ sh q3
Enter file name :
q3data
Hello this is
a test file
for
q3.
[19it006@mylinuxvm practical6]$
```

4. Display a message “Good Morning” or “Good Afternoon” according to the user login time.

Program:

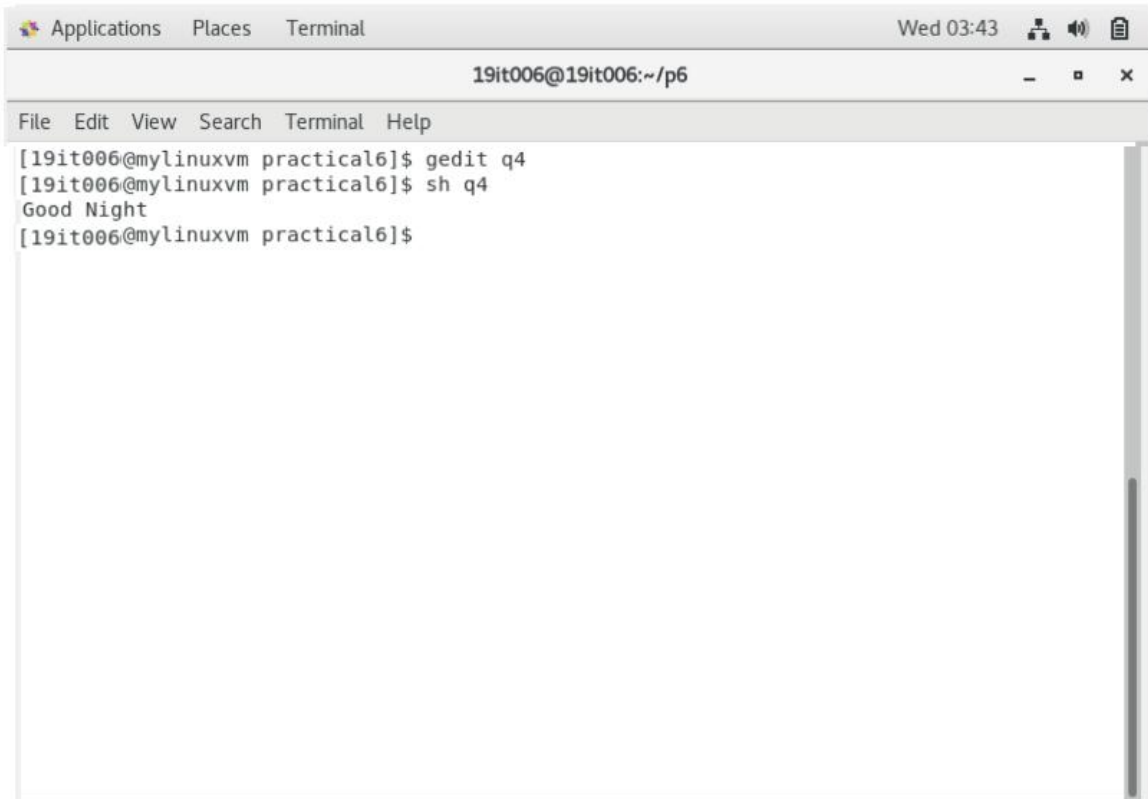


The screenshot shows a text editor window titled 'q4' with the file path '~/shellscript/practical6'. The script content is as follows:

```
#!/bin/bash
log=`lastlog | grep 19IT009 | awk {'print $6'} | awk -F: {'print $1'}`
if [ $log -gt 0 ] && [ $log -lt 12 ]
then
echo "Good Morning"
elif [ $log -gt 12 ] && [ $log -le 18 ]
then
echo "Good Afternoon"
else
echo "Good Night"
fi
```

The status bar at the bottom indicates 'sh', 'Tab Width: 8', 'Ln 11, Col 17', and 'INS'.

Output:

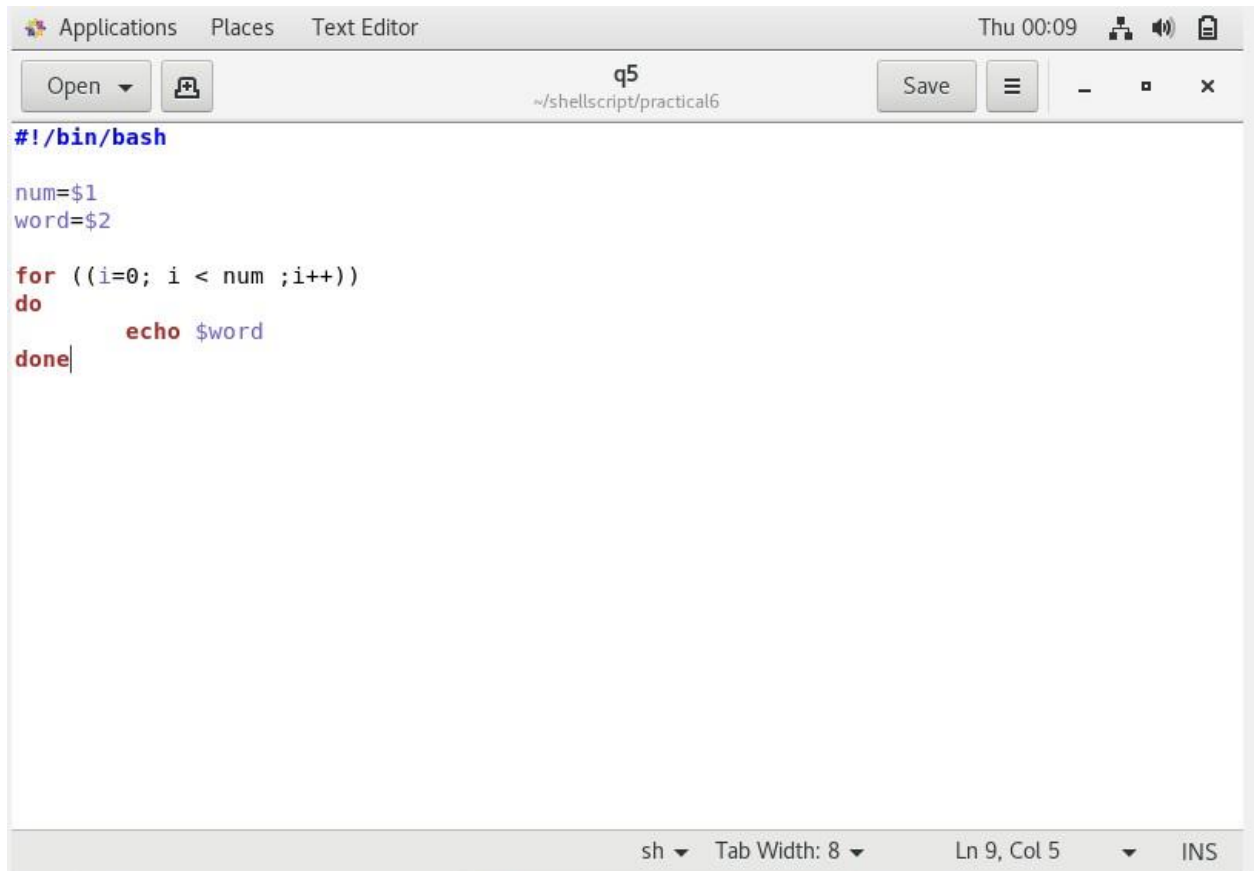


The screenshot shows a terminal window titled "19it006@19it006:~/p6". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal output shows the following commands and responses:

```
[19it006@mylinuxvm practical6]$ gedit q4
[19it006@mylinuxvm practical6]$ sh q4
Good Night
[19it006@mylinuxvm practical6]$
```

5. A shell script, which takes as command line input a number n , and a word. It then prints the word n times, once on each line.

Program:



The screenshot shows a text editor window titled 'q5' with the file path '~/shellscript/practical6'. The editor contains a shell script with the following code:

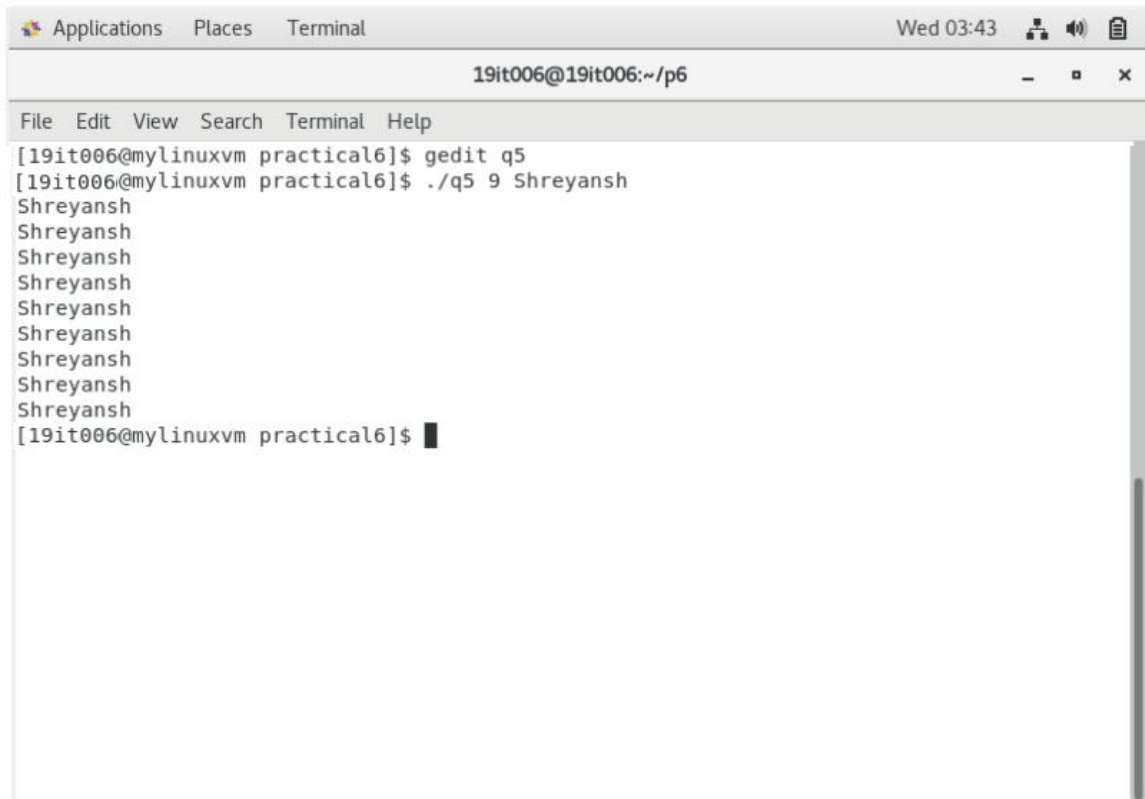
```
#!/bin/bash

num=$1
word=$2

for ((i=0; i < num ;i++))
do
    echo $word
done
```

The status bar at the bottom indicates 'sh', 'Tab Width: 8', 'Ln 9, Col 5', and 'INS'.

Output:



The screenshot shows a terminal window titled '19it006@19it006:~/p6'. The window has a menu bar with 'File', 'Edit', 'View', 'Search', 'Terminal', and 'Help'. The terminal content shows the following commands and output:

```
[19it006@mylinuxvm practical6]$ gedit q5
[19it006@mylinuxvm practical6]$ ./q5 9 Shreyansh
Shreyansh
Shreyansh
Shreyansh
Shreyansh
Shreyansh
Shreyansh
Shreyansh
Shreyansh
Shreyansh
[19it006@mylinuxvm practical6]$
```

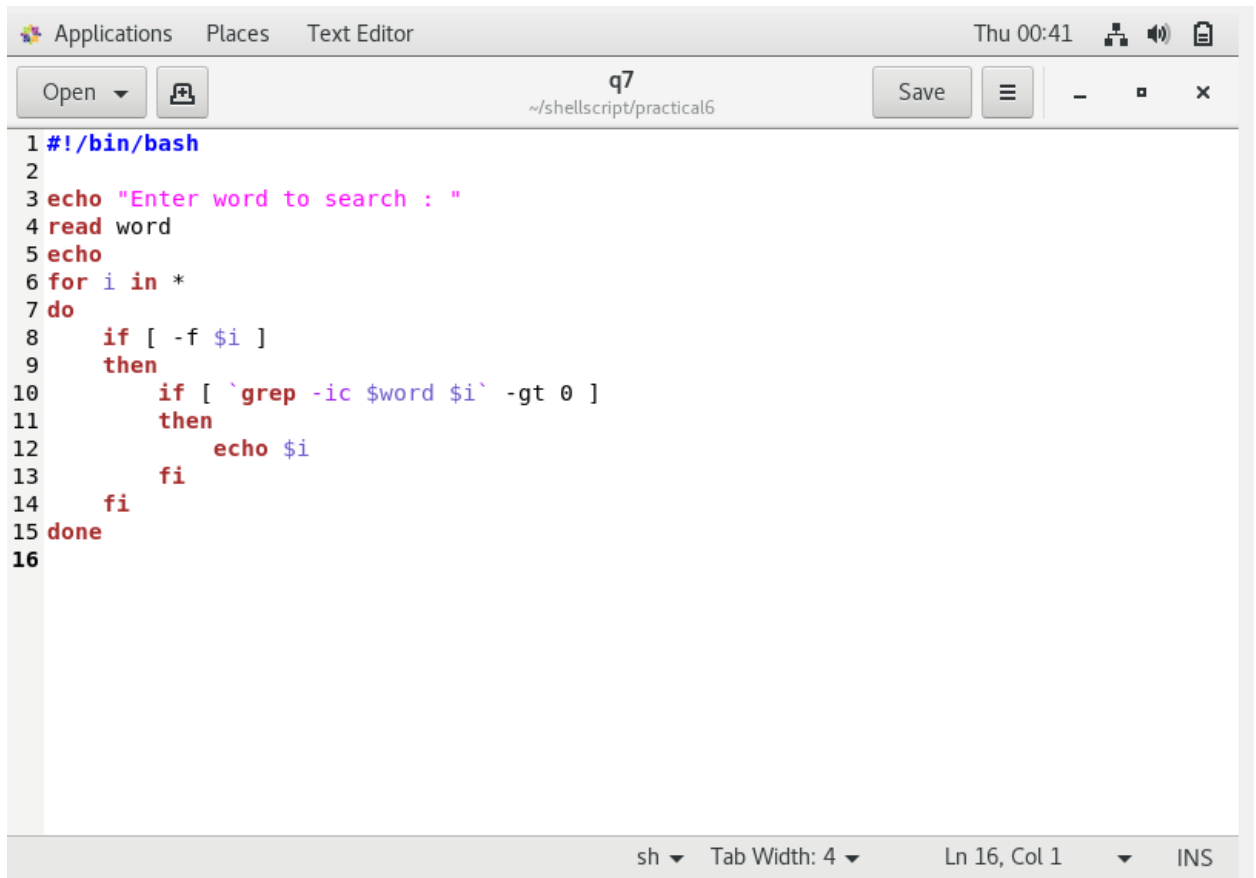
6. A shell script, which reports the names and sizes of all the files in a directory whose size exceeds 1000 bytes, in descending order of their sizes and the total number of such files.

Program:

Output:

7. Write a script that will search for a specific word in all the files in the current directory and then prompt with the file name in which word is found.

Program:



The screenshot shows a text editor window titled 'q7' with the file path '~/.shellscrip/practical6'. The script contains the following code:

```
1 #!/bin/bash
2
3 echo "Enter word to search : "
4 read word
5 echo
6 for i in *
7 do
8     if [ -f $i ]
9     then
10         if [ `grep -ic $word $i` -gt 0 ]
11         then
12             echo $i
13         fi
14     fi
15 done
16
```

The status bar at the bottom indicates 'sh', 'Tab Width: 4', 'Ln 16, Col 1', and 'INS'.

Output:



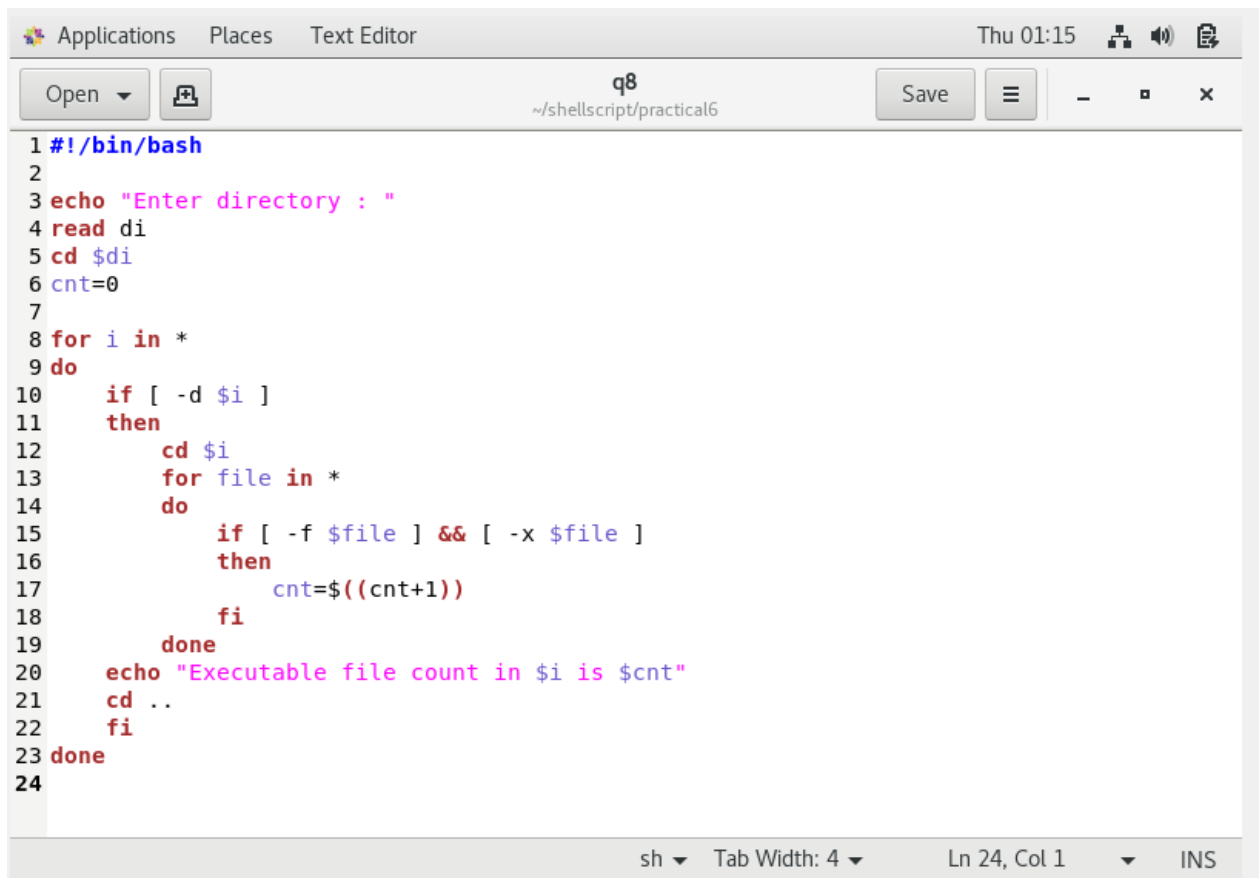
The screenshot shows a terminal window titled "19it006@19it006:~/p6". The window has a menu bar with "File", "Edit", "View", "Search", "Terminal", and "Help". The terminal content shows the following sequence of commands and output:

```
[19it006@mylinuxvm practical6]$ gedit q7
[19it006@mylinuxvm practical6]$ sh q7
Enter word to search :
hello

q3data
[19it006@mylinuxvm practical6]$ cat q3data
Hello this is
a test file
for
q3.
[19it006@mylinuxvm practical6]$
```

8. Write a script to print only the number of executable files in each sub-dir of the argument directory specified.

Program:

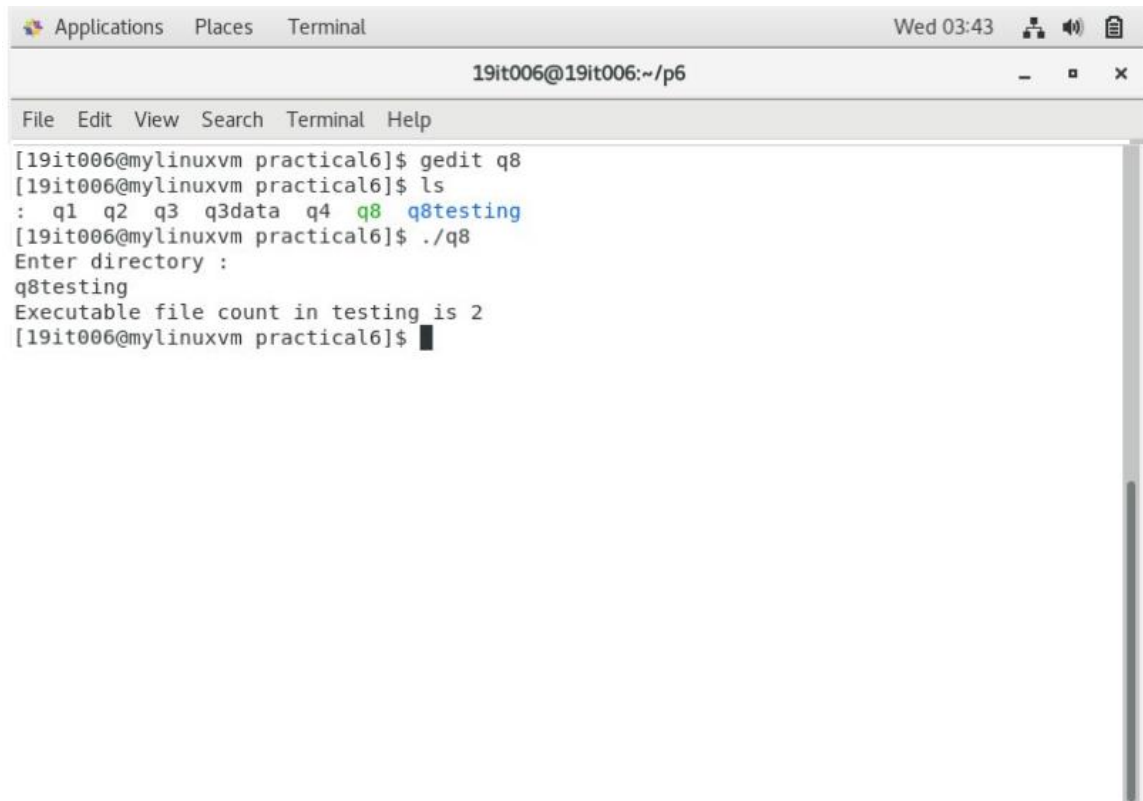


The screenshot shows a text editor window titled 'q8' with the file path '~/shellscrip/practical6'. The window contains a shell script with the following code:

```
1 #!/bin/bash
2
3 echo "Enter directory : "
4 read di
5 cd $di
6 cnt=0
7
8 for i in *
9 do
10     if [ -d $i ]
11     then
12         cd $i
13         for file in *
14         do
15             if [ -f $file ] && [ -x $file ]
16             then
17                 cnt=$((cnt+1))
18             fi
19         done
20         echo "Executable file count in $i is $cnt"
21         cd ..
22     fi
23 done
24
```

The status bar at the bottom indicates 'sh', 'Tab Width: 4', 'Ln 24, Col 1', and 'INS'.

Output:



```
19it006@19it006:~/p6
File Edit View Search Terminal Help
[19it006@mylinuxvm practical6]$ gedit q8
[19it006@mylinuxvm practical6]$ ls
: q1 q2 q3 q3data q4 q8 q8testing
[19it006@mylinuxvm practical6]$ ./q8
Enter directory :
q8testing
Executable file count in testing is 2
[19it006@mylinuxvm practical6]$
```

9. Write a shell script file named `exercise6.sh` that makes a list of files in your home directory that were changed less than 24 hours ago, but leave out directories.

Program:

Output:

10. Write a shell script which will take file name as argument and check whether the file name is a dir or not and then proceed further only if it is a dir, else give usage message. The script should then print in the tabular format, name of each sub-dir (within the argument dir) and a count of the number of top-level files in that sub-dir.

Program:

Output: