

27.05.21

## Practical-8

Title : To execute shell programs

### Theory:

#### 1] Shell Scripting

⇒ Usually shells are interactive that mean they accept commands as input from user and execute them. However some time we want to execute bunch of commands, so we have to type in all commands each time in terminal. As shell can also take input from a file, we can write these commands in a file and can execute them in shell to avoid repeatative work. These files are called shell scripts or shell programs. Shell scripts are similar to the batch file in MS-DOS. Each shell script is saved with .sh file extension eg myscript.sh.

A shell script comprises of following elements :-

- 1) Shell keywords - if, else, break etc.
- 2) Shell commands - cd, ls, echo, pwd
- 3) control flow - if...then... else, case, loops, function



# Programs.

1) Print numbers till 5 using while loop

```
=> vi numbers.sh
valid=true
count=1
while [ $valid ]
do
echo $count
if [ $count -eq 5 ];
then
break
fi
((count++))
done.
```

Output: \$ bash numbers.sh

1

2

3

4

5



2) Sum of two numbers

```
=> $ vi sum.sh
echo "Enter first Number"
read num1
echo "Enter Second Number"
read num2
```

```
sum=$((num1 + num2))
echo "the sum of $num1 and $num2 is $sum"
```

Output :

```
bash sum.sh
Enter first Number
5
Enter second Number
5
the sum of 5 and 5 is 10
```

3) To verify if admin password is correct

```
=> $ vi verify.sh
echo "Enter username"
read username
echo "Enter password"
read password
```

```
if [[ ($username == "admin" && password == "secret")
]] ; then
```



```
echo "valid user"  
else  
echo "Invalid user"  
fi
```

Output : bash verify.sh  
Enter username  
admin  
Enter password  
secret  
valid user.

4) To give the factorial of a number  
⇒ vi factorial.sh

```
echo "Enter number to find factorial"  
read num
```

```
factorial=1
```

```
counter=$num
```

```
while [[ $counter -gt 0 ]]; do
```

```
    factorial=$((factorial * counter))
```

```
    counter=$((counter - 1))
```

```
done
```

```
echo "Factorial of $num is $factorial"
```

Output : bash factorial.sh  
Enter a number : 4  
Factorial of 4 is 24



184011

5) To show the use of switch case

⇒ \$ vi case.sh

```
echo "Enter your lucky number"
```

```
read n
```

```
case $n in
```

```
10)
```

```
echo "You got 1st prize" ;;
```

```
5)
```

```
echo "You got second prize" ;;
```

```
9)
```

```
echo "You got 3rd prize" ;;
```

```
*)
```

```
echo "Sorry, try for the next time" ;;
```

```
esac
```

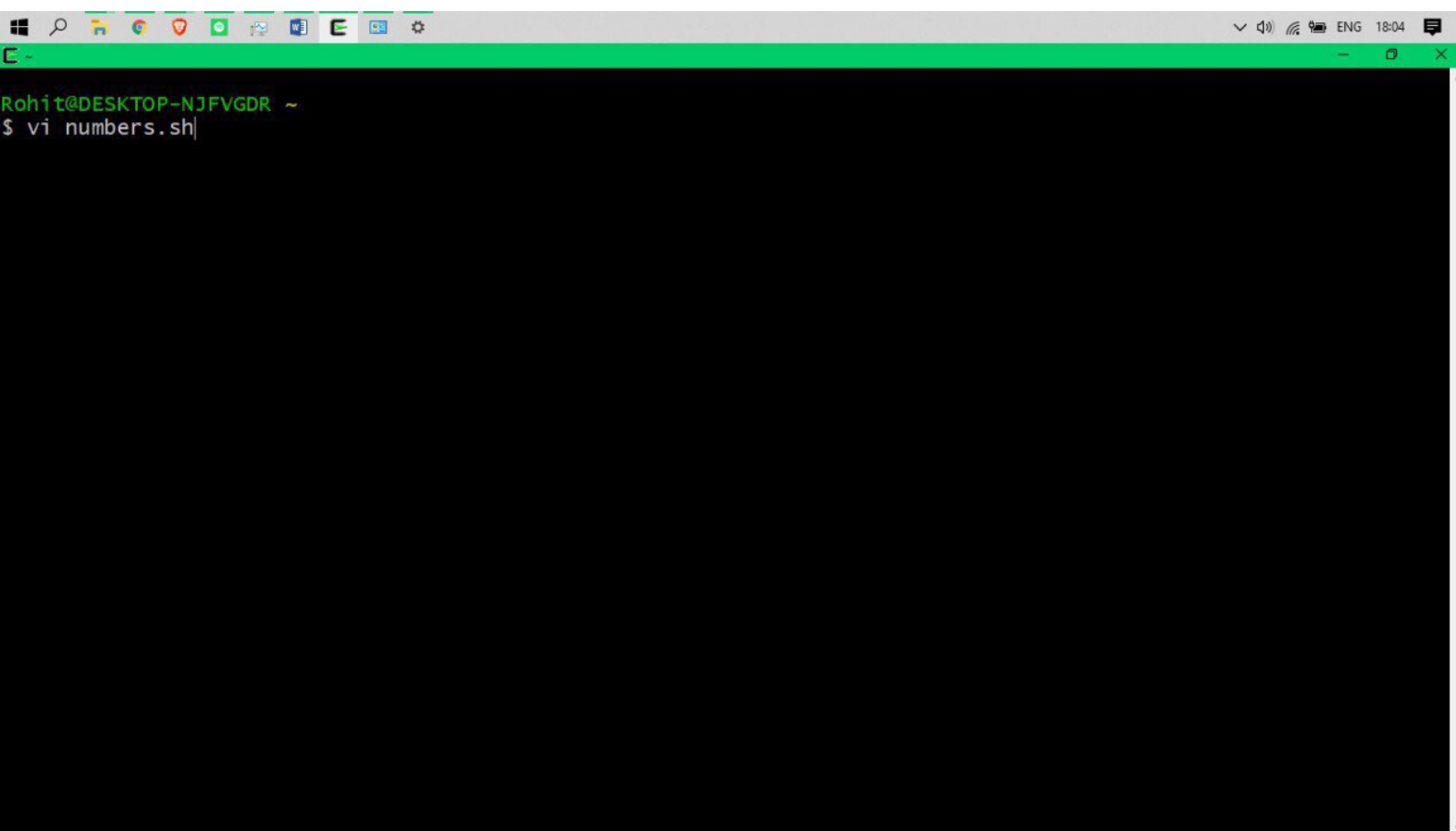
Output : ~~\$~~ bash case.sh

Enter your lucky number

10

You got 1st prize

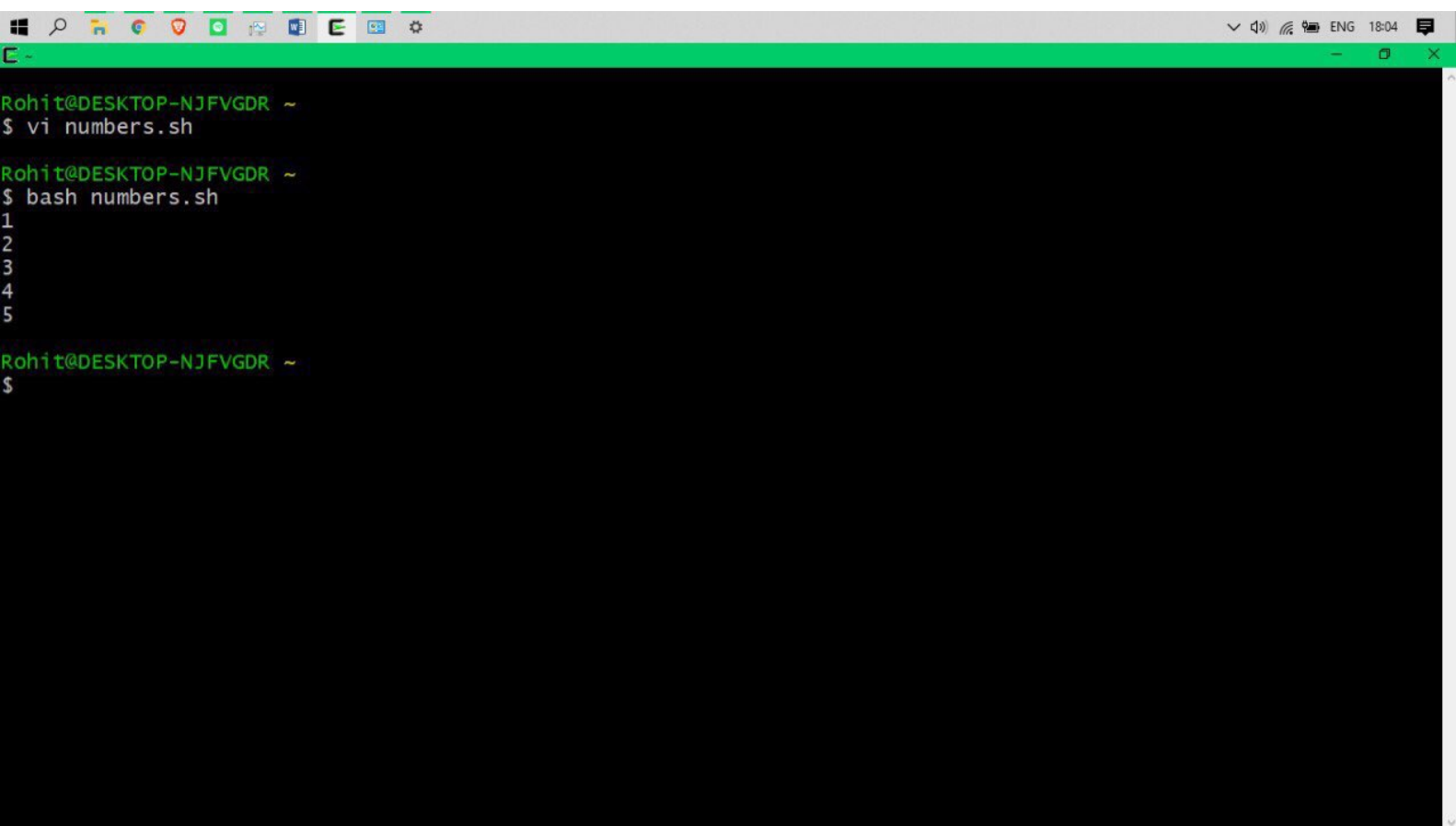
Conclusion : After performing this practical I learnt to run shell commands on linux, cygwin.



A screenshot of a Windows terminal window. The title bar is green and contains the text "Rohit@DESKTOP-NJFVGDR ~". The terminal window is black with green text. The prompt is "\$ vi numbers.sh". The terminal window is open on a desktop with a taskbar at the bottom. The taskbar contains icons for the Start button, Search, File Explorer, Google Chrome, Microsoft Edge, and several other applications. The system tray on the right shows the volume icon, network status, and the time "18:04".

```
Rohit@DESKTOP-NJFVGDR ~  
$ vi numbers.sh
```



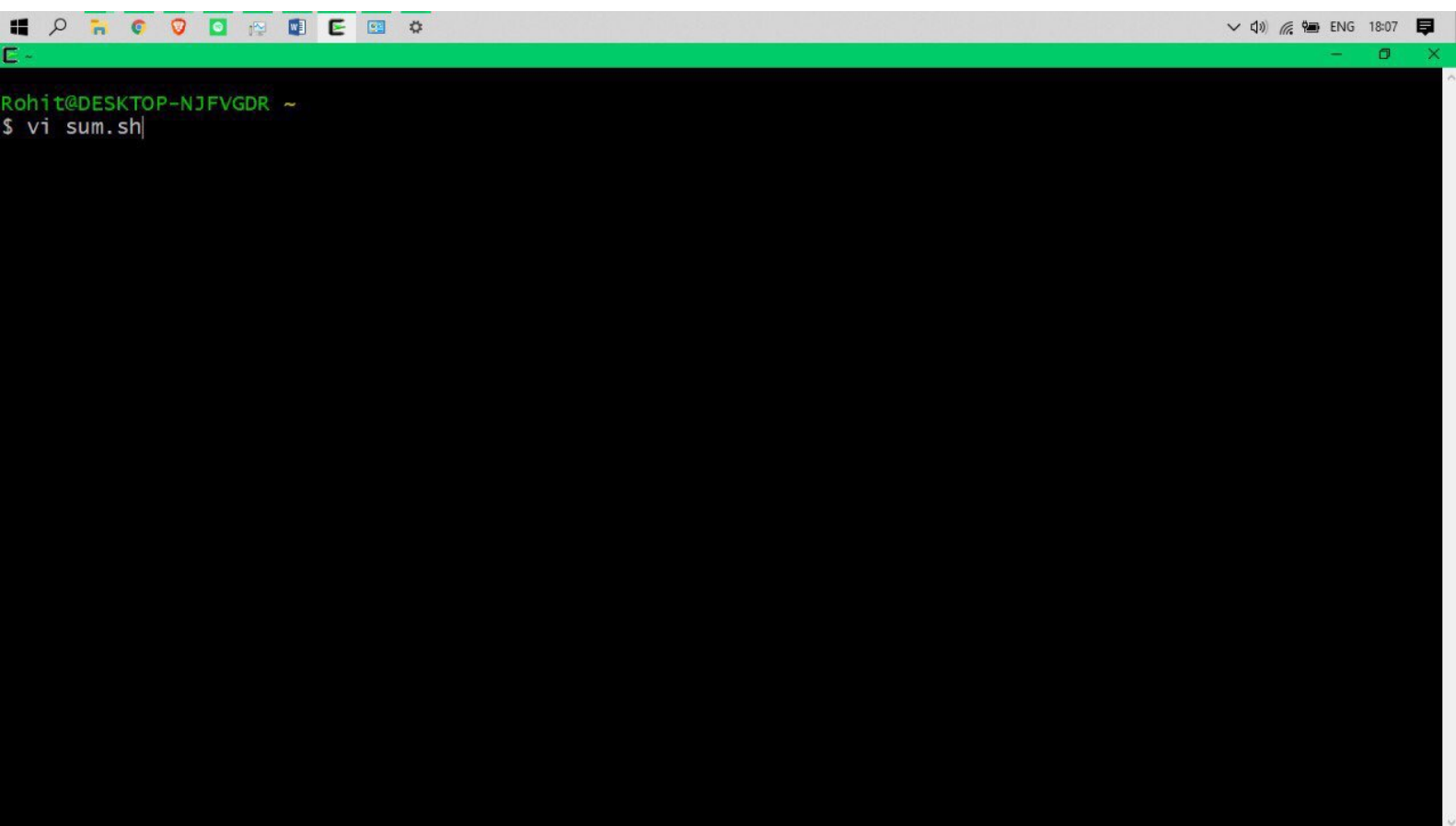


A terminal window with a green title bar and a black background. The window title is not fully visible. The terminal shows the following sequence of commands and output:

```
Rohit@DESKTOP-NJFVGDR ~  
$ vi numbers.sh  
  
Rohit@DESKTOP-NJFVGDR ~  
$ bash numbers.sh  
1  
2  
3  
4  
5  
  
Rohit@DESKTOP-NJFVGDR ~  
$
```

The terminal window has a standard Windows taskbar at the top with various icons and a system tray on the right showing the time as 18:04 and language as ENG.





A screenshot of a Windows terminal window. The title bar is green and contains the text "Rohit@DESKTOP-NJFVGDR ~". The terminal area is black with green text. The prompt is "\$ vi sum.sh". The terminal window is open on a desktop with various icons visible in the taskbar.

```
Rohit@DESKTOP-NJFVGDR ~  
$ vi sum.sh
```

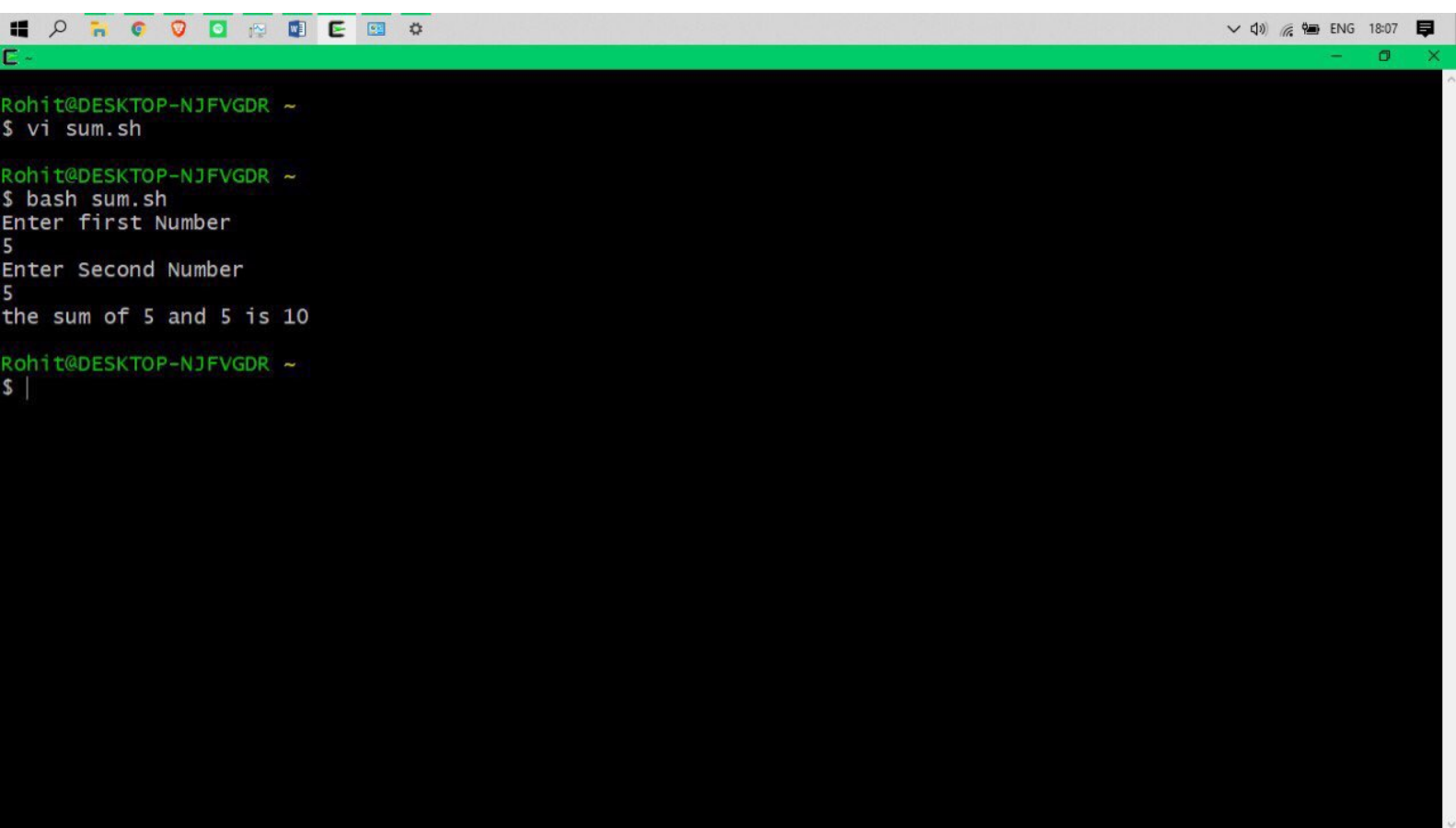
```
echo "Enter first Number"
read num1
echo "Enter Second Number"
read num2

sum=$((num1 + num2))

echo "the sum of $num1 and $num2 is $sum"
```

"sum.sh" 10 lines, 141 characters

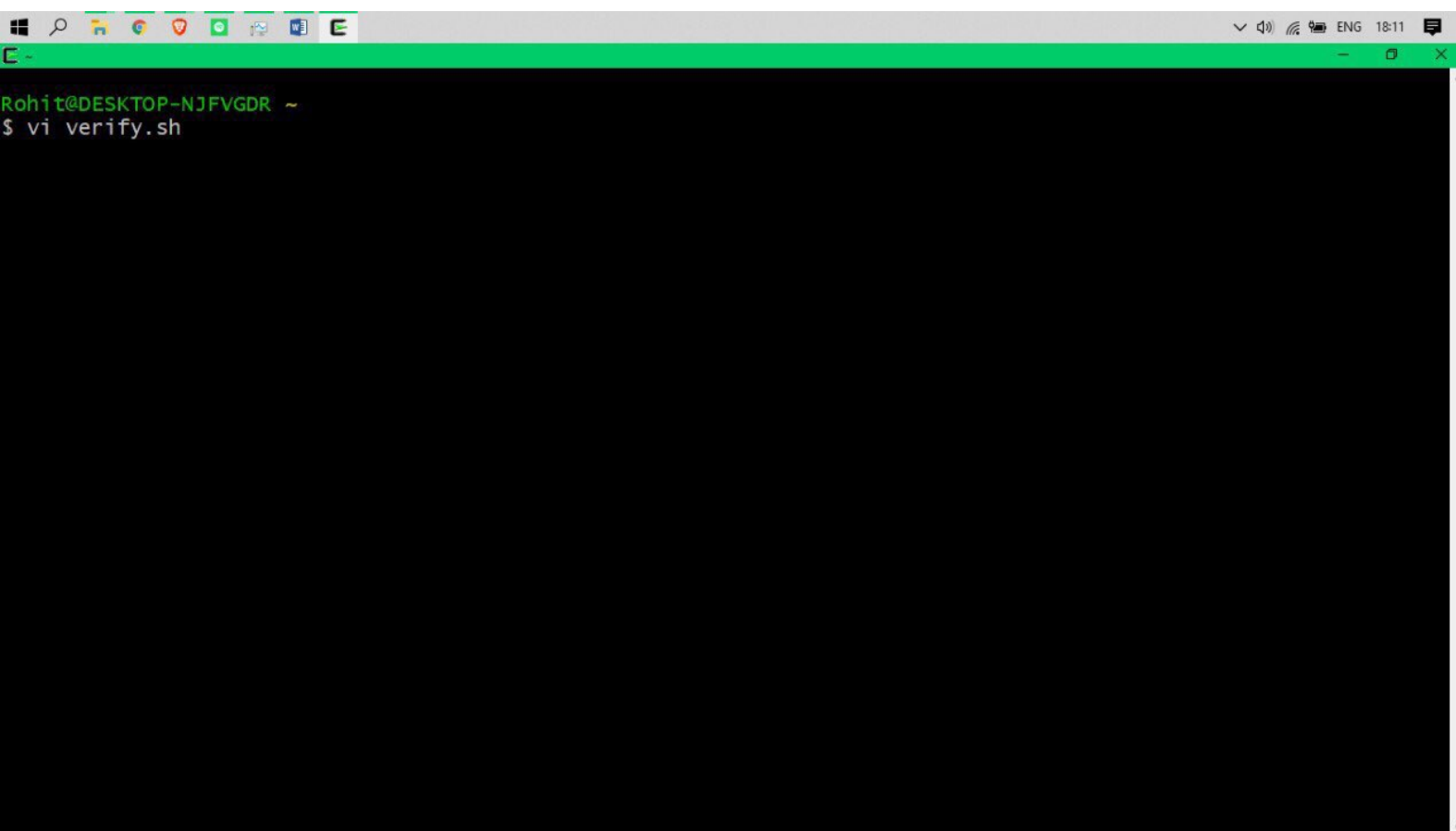




The image shows a Windows terminal window with a green title bar. The terminal displays the following text:

```
Rohit@DESKTOP-NJFVGDR ~  
$ vi sum.sh  
  
Rohit@DESKTOP-NJFVGDR ~  
$ bash sum.sh  
Enter first Number  
5  
Enter Second Number  
5  
the sum of 5 and 5 is 10  
  
Rohit@DESKTOP-NJFVGDR ~  
$ |
```

The terminal window has a taskbar at the top with various application icons and a system tray on the right showing the time as 18:07 and language as ENG.



A screenshot of a Windows terminal window. The title bar is green and contains the text "Rohit@DESKTOP-NJFVGDR ~". The terminal area is black with green text. The prompt is "\$" and the command entered is "vi verify.sh". The terminal window is open on a desktop with a taskbar at the bottom showing various application icons.

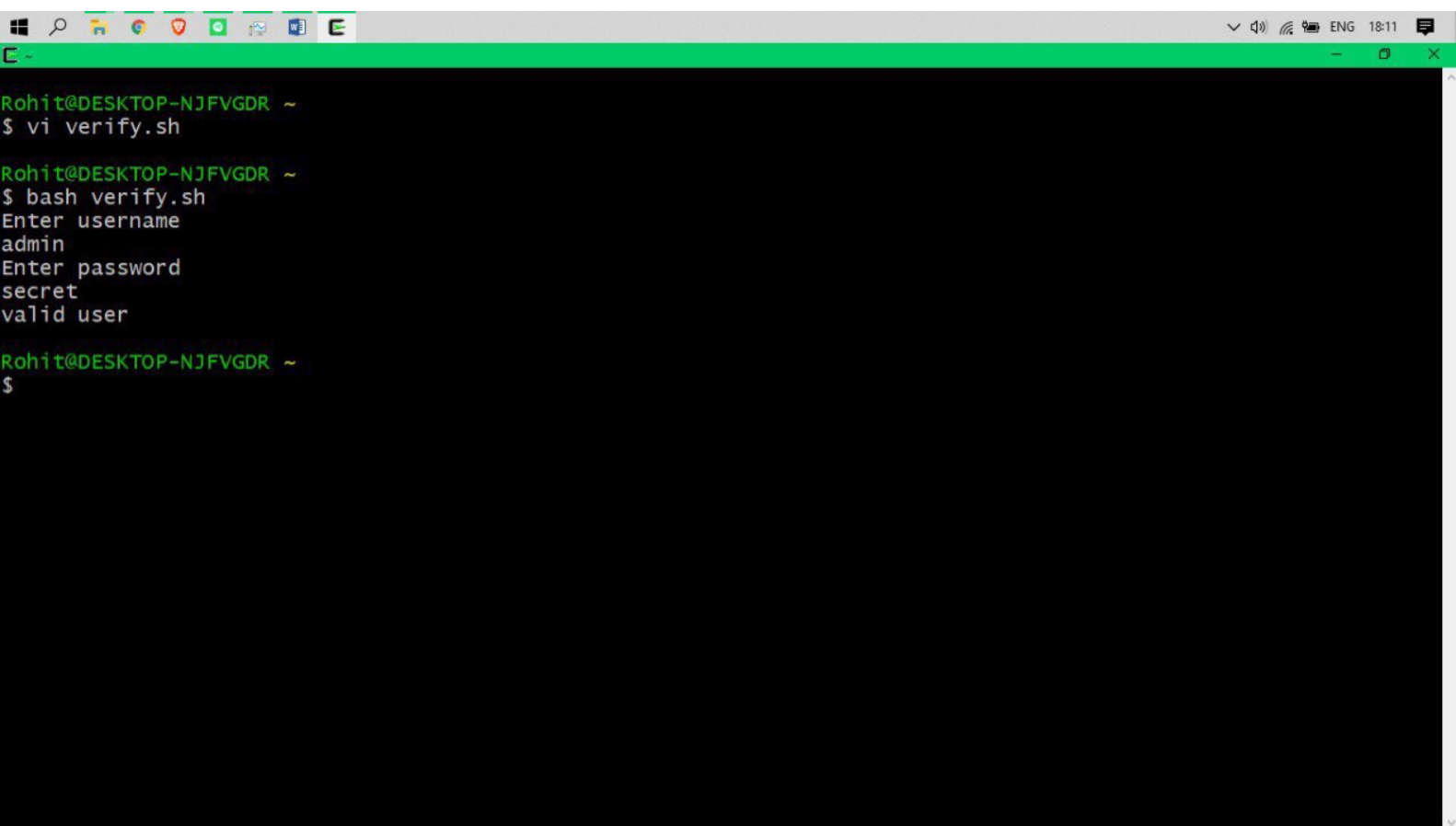
```
Rohit@DESKTOP-NJFVGDR ~  
$ vi verify.sh
```



```
echo "Enter username"
read username
echo "Enter password"
read password

if [[ ( $username == "admin" && $password == "secret" ) ]]; then
echo "valid user"
else
echo "invalid user"
fi
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"
```

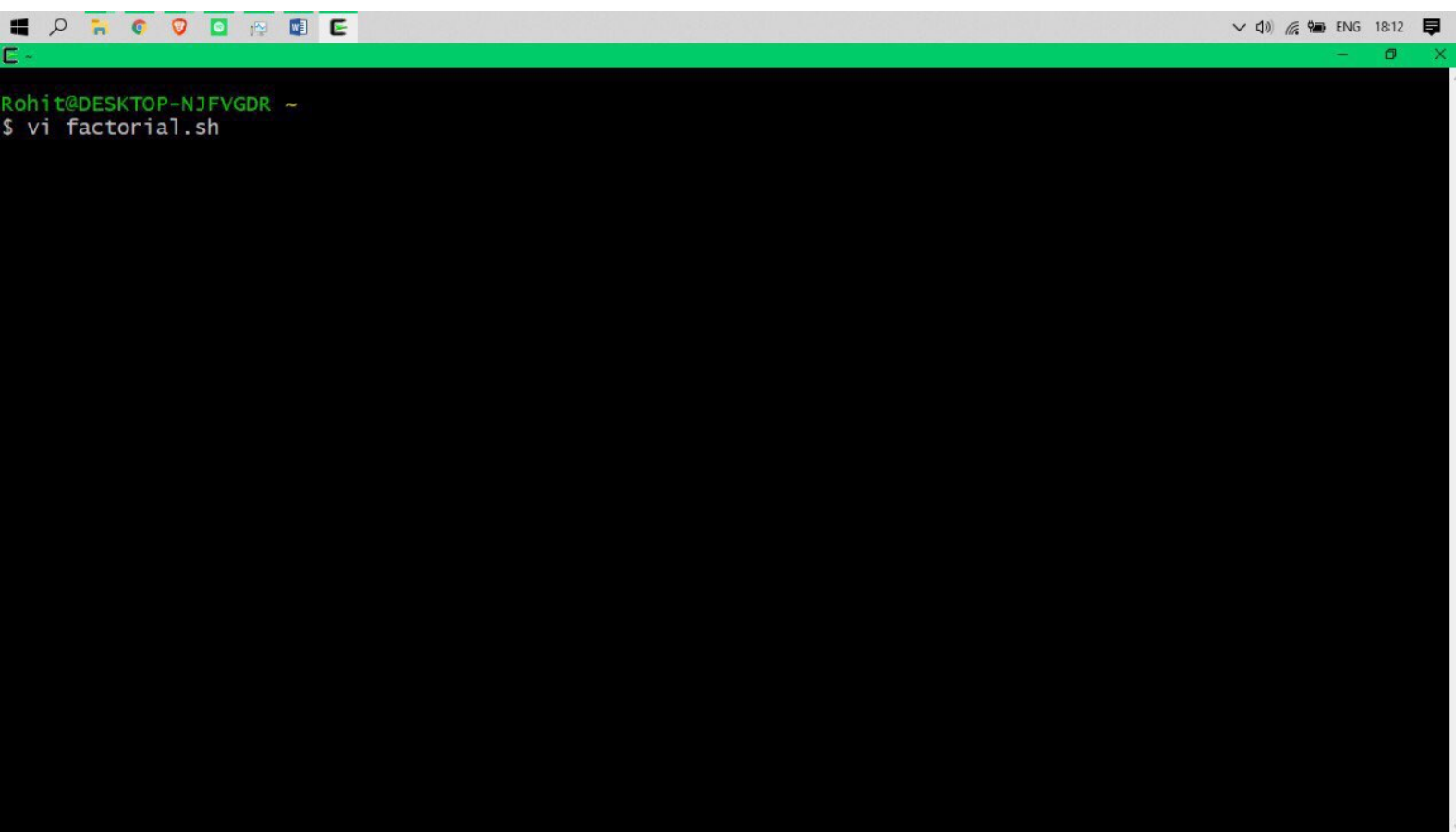
"verify.sh" 10 lines, 184 characters



A screenshot of a Windows terminal window. The title bar shows standard Windows icons and the text "Rohit@DESKTOP-NJFVGDR ~". The terminal has a black background with green text. The user enters the command `vi verify.sh`. The terminal then shows the command `bash verify.sh` being executed. The script prompts for a username, where `admin` is entered, and for a password, where `secret` is entered. The script then outputs `valid user`. The prompt returns to `Rohit@DESKTOP-NJFVGDR ~` with a dollar sign.

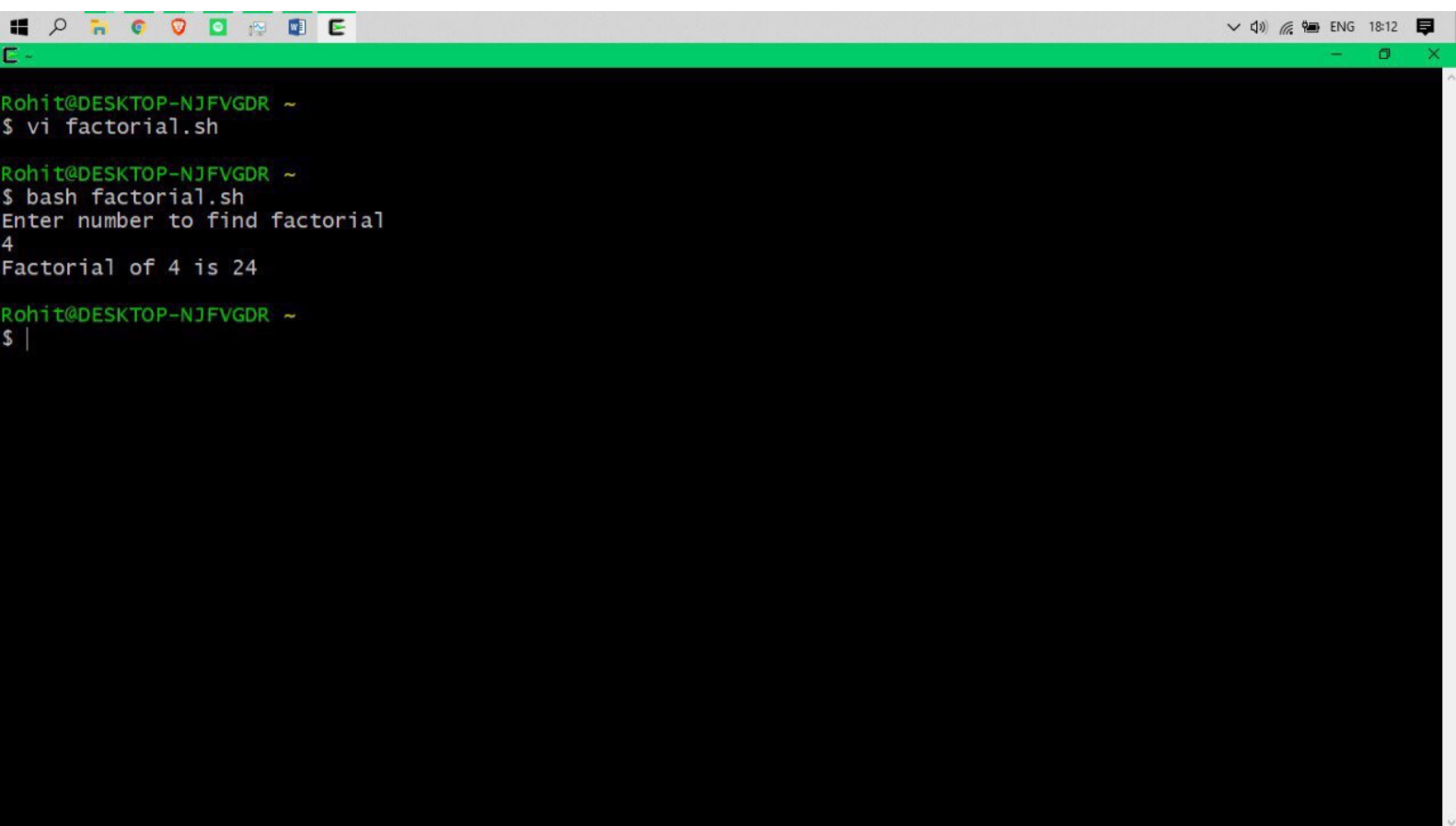
```
Rohit@DESKTOP-NJFVGDR ~  
$ vi verify.sh  
  
Rohit@DESKTOP-NJFVGDR ~  
$ bash verify.sh  
Enter username  
admin  
Enter password  
secret  
valid user  
  
Rohit@DESKTOP-NJFVGDR ~  
$
```



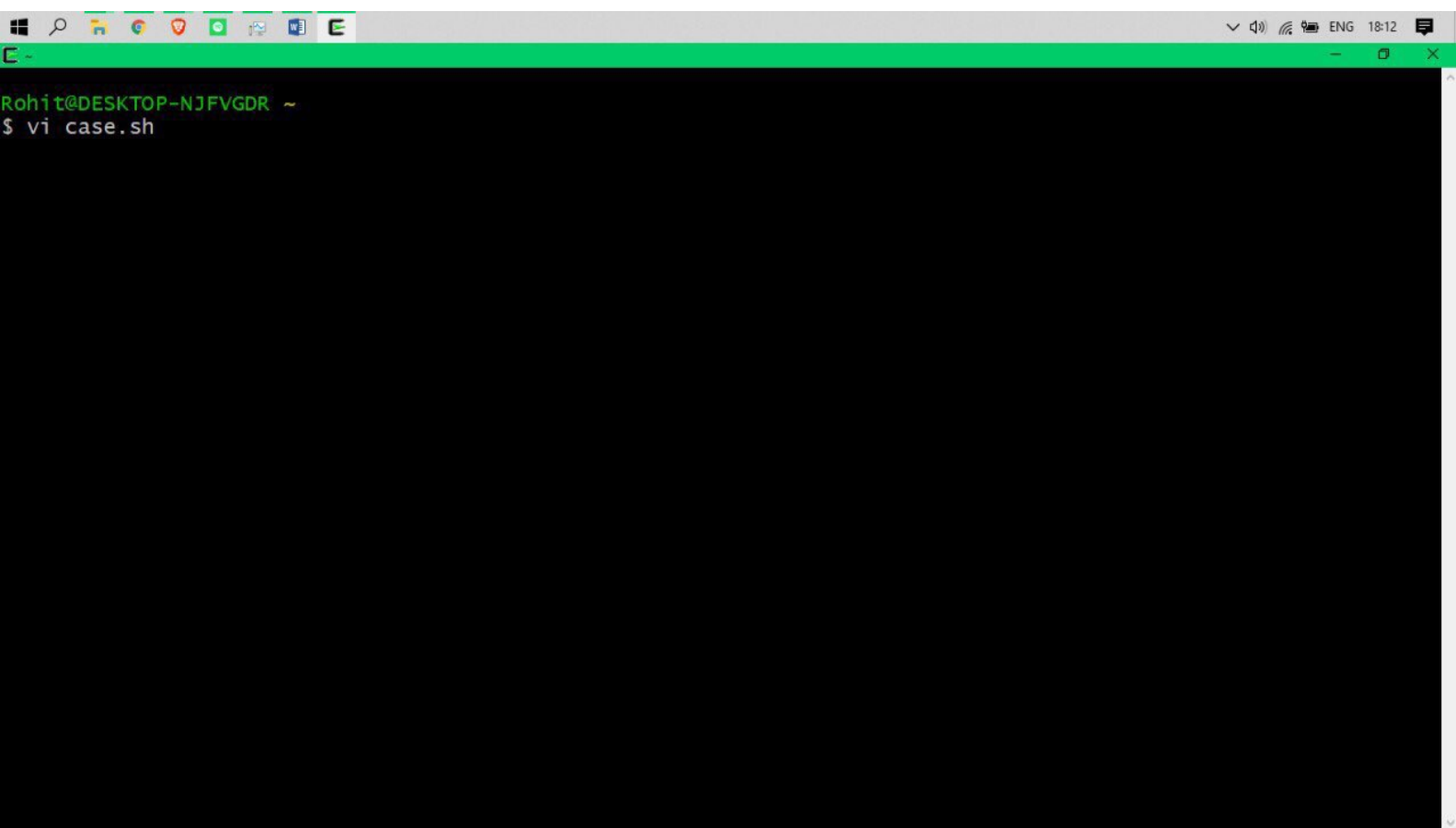






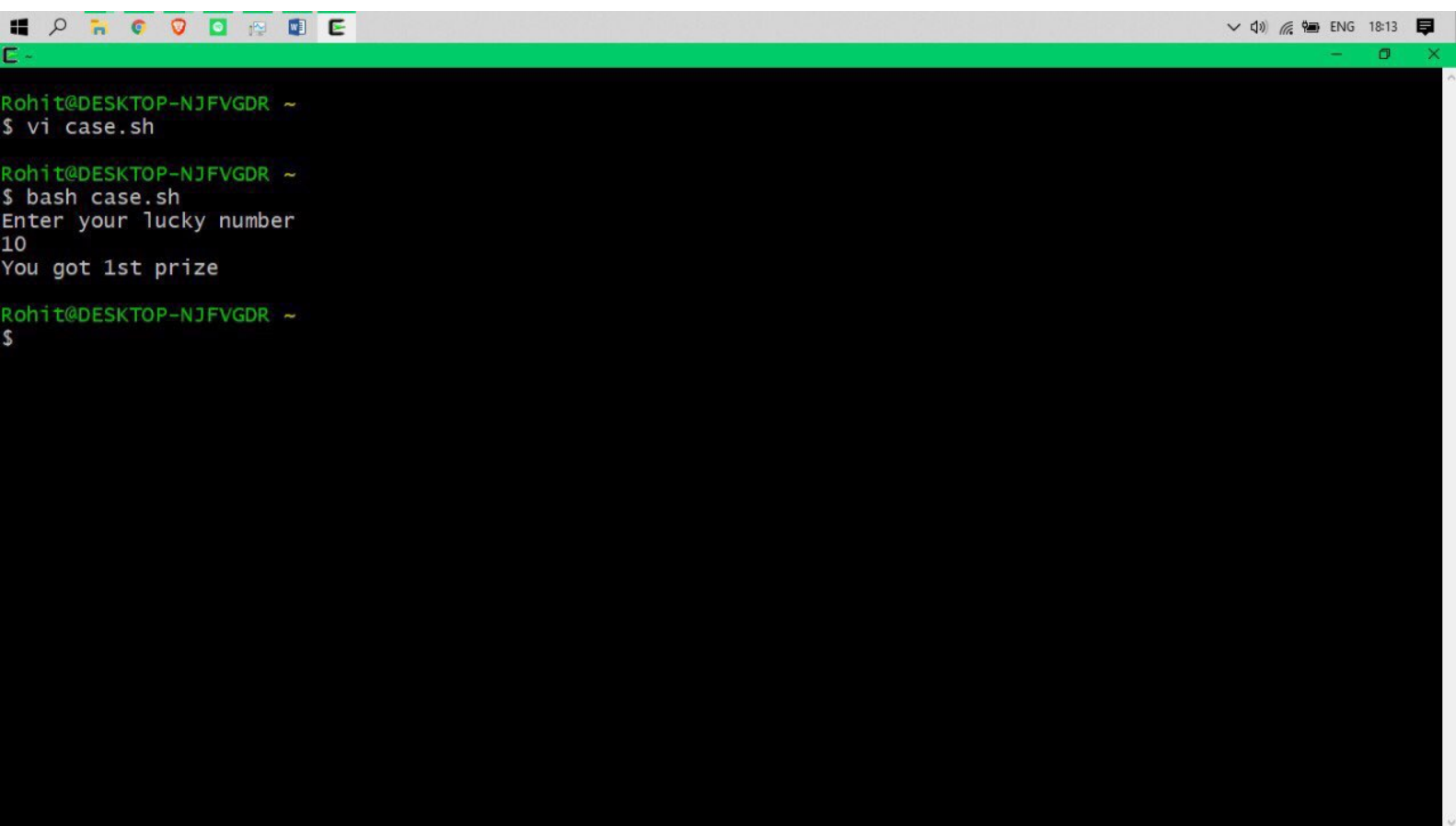
A screenshot of a Windows terminal window with a green title bar. The terminal shows a user named Rohit at a desktop named DESKTOP-NJFVGDR. The user runs 'vi factorial.sh' to edit a script, then 'bash factorial.sh' to execute it. The script prompts for a number, the user enters '4', and the script outputs 'Factorial of 4 is 24'. The prompt returns to the shell.

```
Rohit@DESKTOP-NJFVGDR ~  
$ vi factorial.sh  
  
Rohit@DESKTOP-NJFVGDR ~  
$ bash factorial.sh  
Enter number to find factorial  
4  
Factorial of 4 is 24  
  
Rohit@DESKTOP-NJFVGDR ~  
$ |
```



[illegible]





The image shows a Windows terminal window with a green title bar. The terminal displays the following commands and output:

```
Rohit@DESKTOP-NJFVGDR ~  
$ vi case.sh  
  
Rohit@DESKTOP-NJFVGDR ~  
$ bash case.sh  
Enter your lucky number  
10  
You got 1st prize  
  
Rohit@DESKTOP-NJFVGDR ~  
$
```

The terminal window has a taskbar at the top with various application icons and a system tray on the right showing the time as 18:13 and language as ENG.