

OS ASSIGNMENT -3

ROLL NO: U21CS052

NAME : PANCHAL GUNGUN PARESH

Different shells scripting programs

1. To make a file

Sudo gedit prog1.sh

2. To give the execute permission

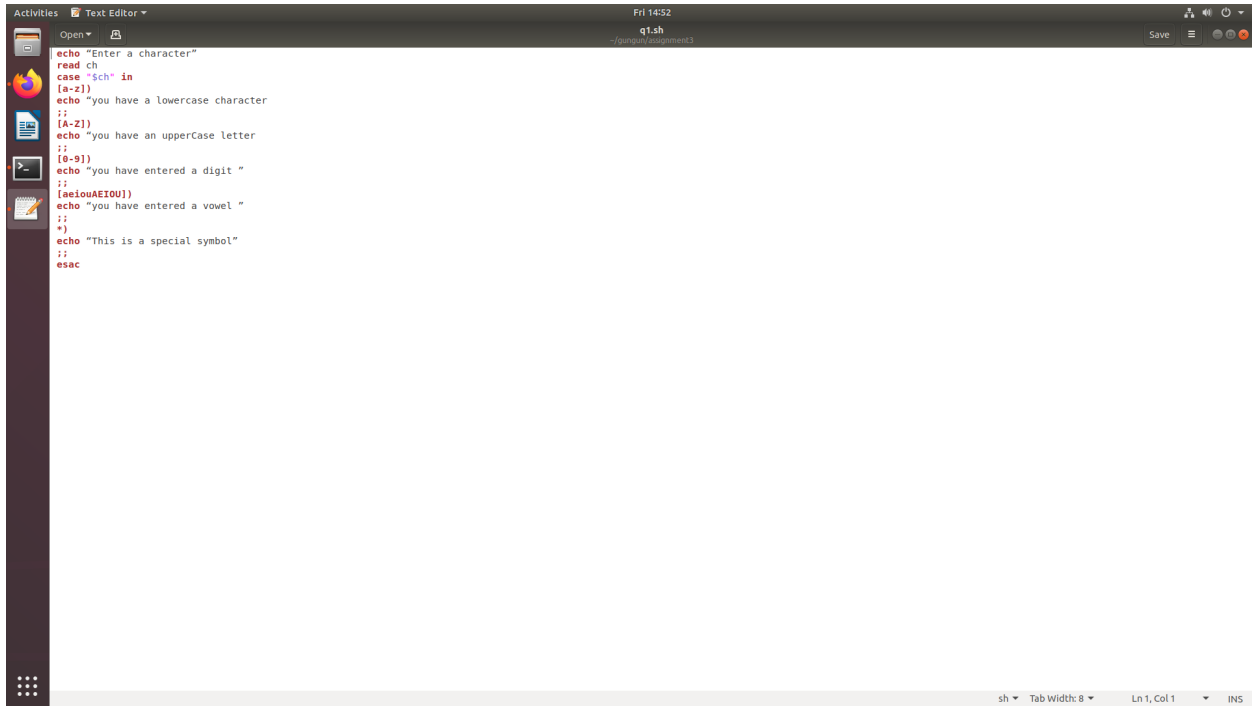
sudo chmod +x q1.sh

3. To run the program

bash q1.file

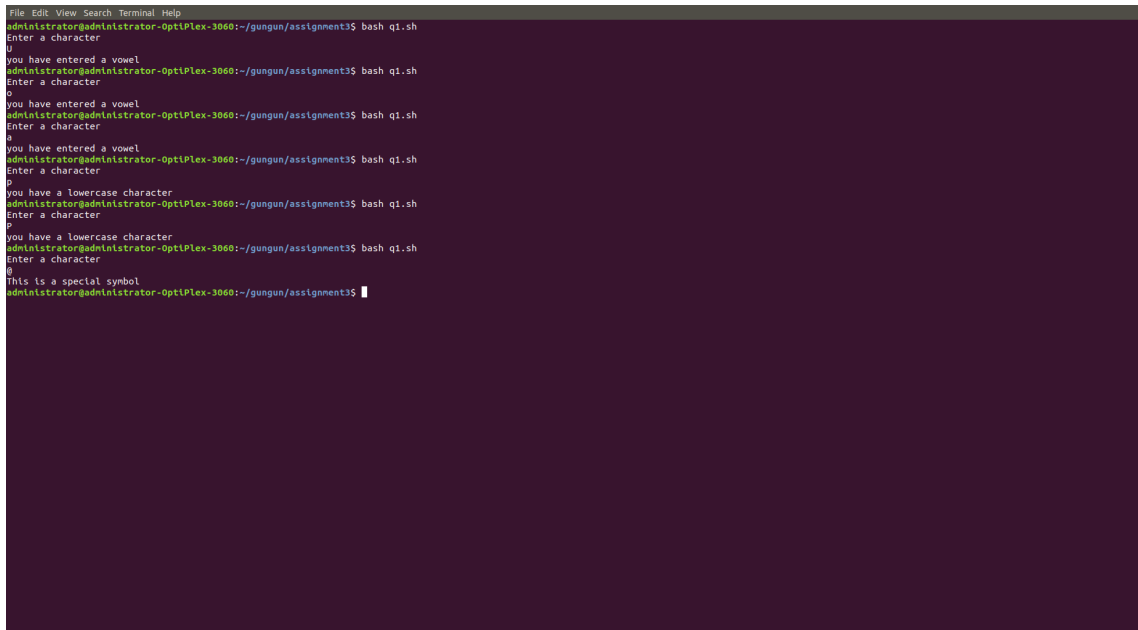
Que1 . Shell program to check whether a character is a

- Uppercase
- Lowercase
- Digit
- Special Symbol
- Vowel



```
Activities Text Editor Fri 14:52
q1.sh
~/gungun/assignment3

echo "Enter a character"
read ch
case "$ch" in
[a-z])
echo "you have a lowercase character"
;;
[A-Z])
echo "you have an upperCase letter"
;;
[0-9])
echo "you have entered a digit "
;;
[aeiouAEIOU])
echo "you have entered a vowel "
;;
*)
echo "This is a special symbol"
;;
esac
```



```
File Edit View Search Terminal Help
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q1.sh
Enter a character
u
you have entered a vowel
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q1.sh
Enter a character
0
you have entered a digit
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q1.sh
Enter a character
@
This is a special symbol
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$
```

Que 2 .Shell program to check whether a given file is a directory or not

```
administrator@administrator-OptiPlex-3060: ~/gungun/assignment3
File Edit View Search Terminal Help
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q2.sh q1.sh
The provided argument is the file.
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q2.sh tempFolder/
The provided argument is the directory.
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$
```

```
#!/bin/sh
#Using -d option we are checking whether the first argument is a directory or not.
#$1 refers to the first argument
if [ -d $1 ]
then
    echo "The provided argument is the directory."
#Using -f option we are checking whether the first argument is a file or not.
elif [ -f $1 ]
then
    echo "The provided argument is the file."
#if the provided argument is not file and directory then it does not exist on the system.
else
    echo "The given argument does not exist on the file system."
fi
```

Que 3 .Shell program to count the number of files and directories in the given folder

```
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q3.sh
No. of files is 4
No. of directories is 2
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$
```

```
#!/bin/sh
```

```
echo "No. of files is $(find "$@" -type f | wc -l)"
echo "No. of directories is $(find "$@" -type d | wc -l)"
```

Que 4 . Add two numbers supplied by the user and also by the command line argument .

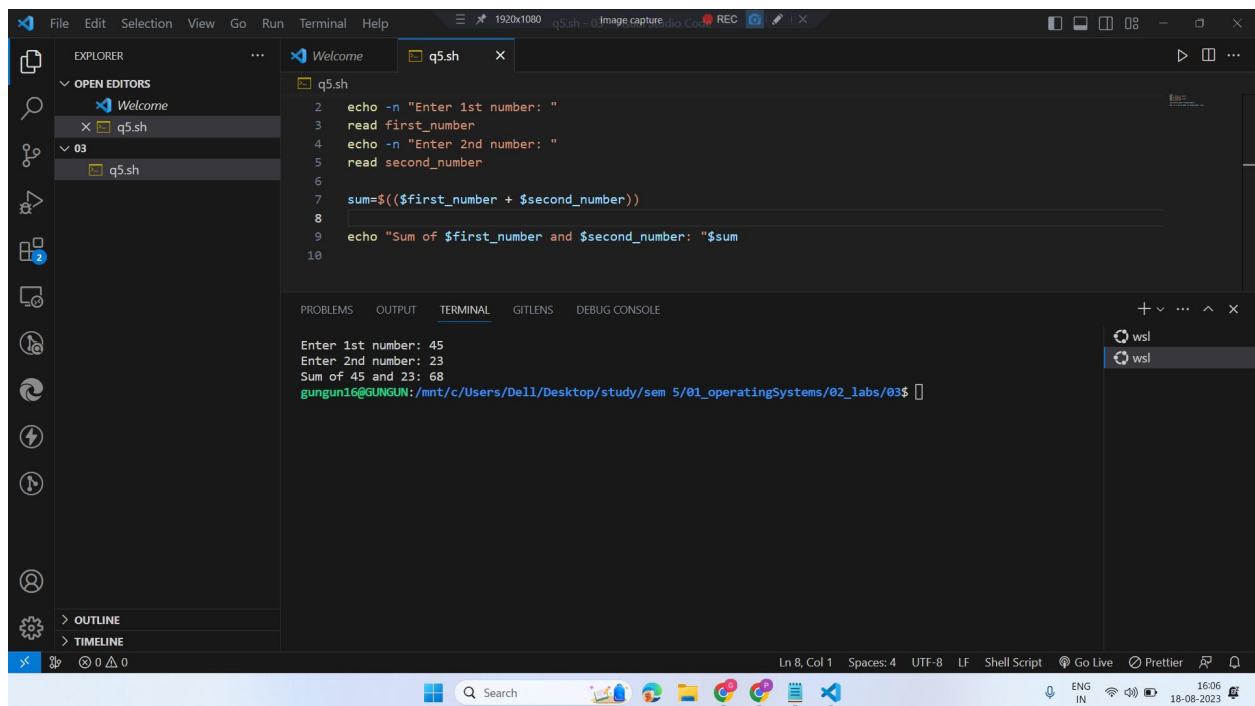
```
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ sudo chmod +x q4.sh
administrator@administrator-OptiPlex-3060:~/gungun/assignment3$ bash q4.sh
Sum is: 30
```

```
#!/bin/bash
# Calculate the sum of two integers with pre initialize values
# in a shell script

a=10
b=20

sum=$(( $a + $b ))

echo "Sum is: $sum"
```

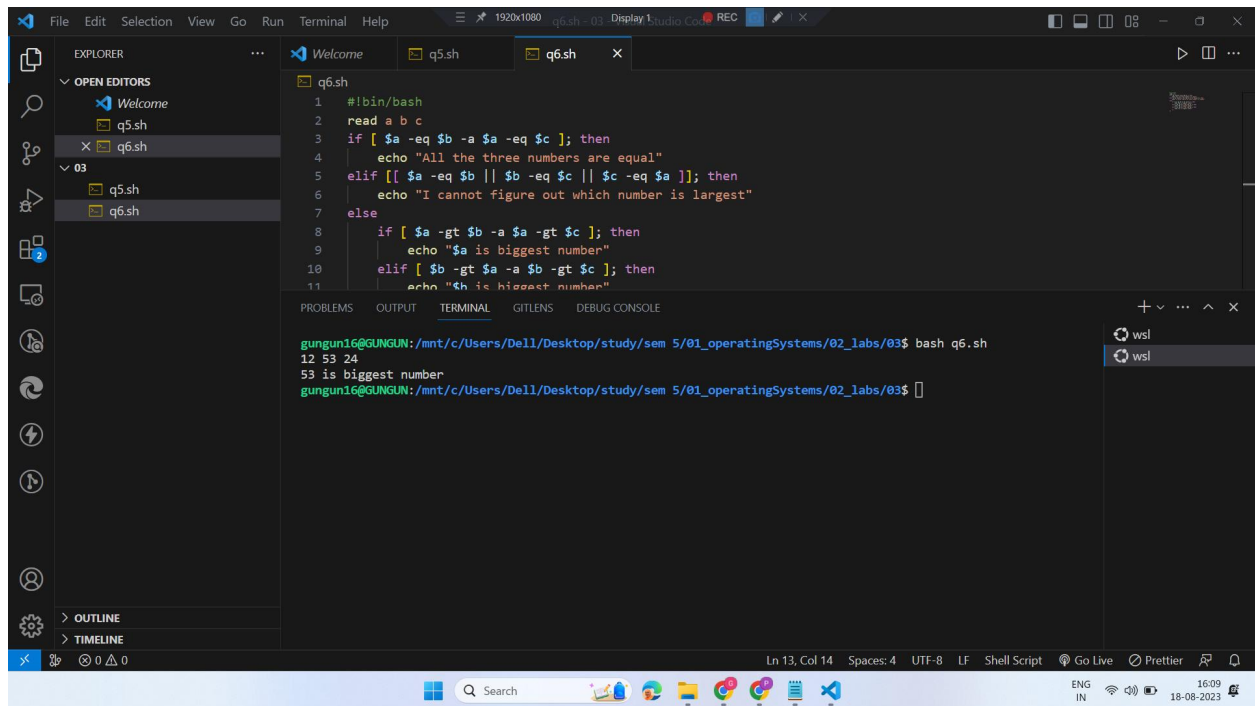


The screenshot shows a Visual Studio Code editor window with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named '03' containing a file 'q5.sh'. The editor displays the contents of 'q5.sh', which is a shell script that prompts the user for two numbers, reads them, calculates their sum, and prints the result. The terminal shows the execution of the script, where the user enters '45' and '23', and the script outputs 'Sum of 45 and 23: 68'.

```
q5.sh
1 echo -n "Enter 1st number: "
2 read first_number
3 echo -n "Enter 2nd number: "
4 read second_number
5
6
7 sum=$((first_number + $second_number))
8
9 echo "Sum of $first_number and $second_number: "$sum
10
```

```
Enter 1st number: 45
Enter 2nd number: 23
Sum of 45 and 23: 68
gungun16@GUNGUN: /mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$
```

Que 5 .Find out the largest number among the following

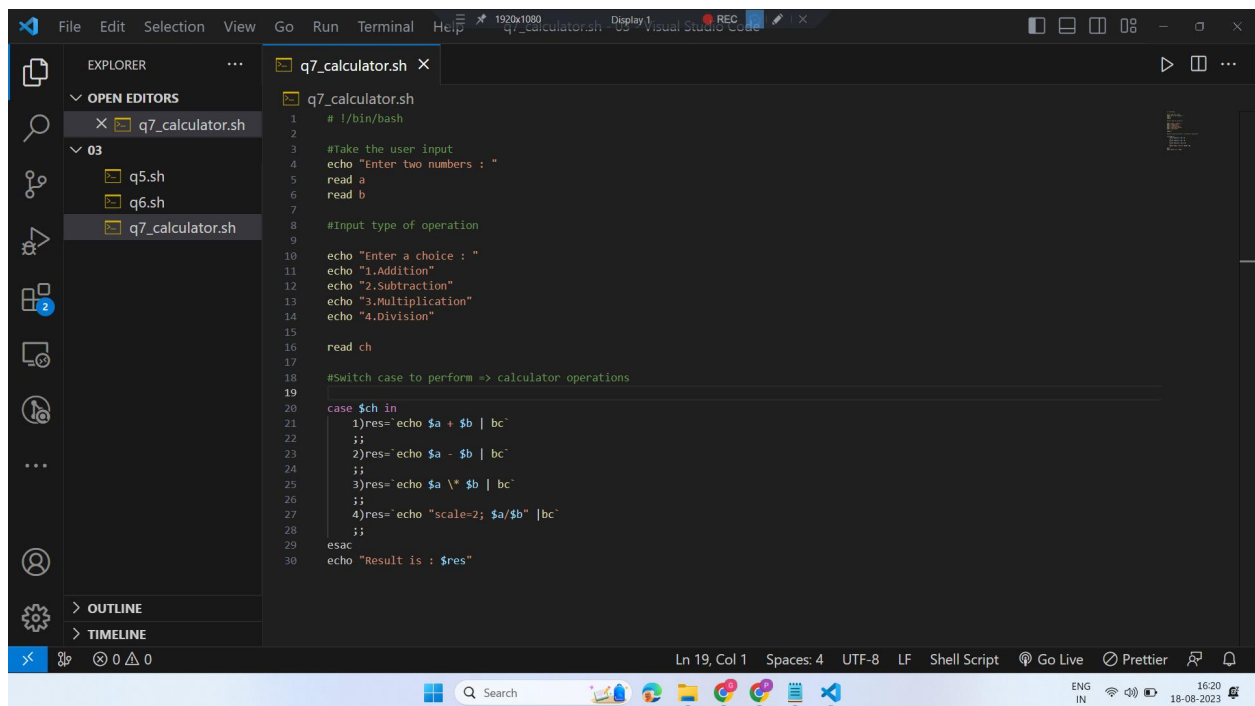


The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named '03' containing files 'q5.sh' and 'q6.sh'. The editor displays the content of 'q6.sh', which is a shell script to find the largest of three numbers. The terminal shows the command 'bash q6.sh' being executed, with input '12 53 24' and output '53 is biggest number'.

```
1 #!/bin/bash
2 read a b c
3 if [ $a -eq $b -a $a -eq $c ]; then
4     echo "All the three numbers are equal"
5 elif [ [ $a -eq $b || $b -eq $c || $c -eq $a ]; then
6     echo "I cannot figure out which number is largest"
7 else
8     if [ $a -gt $b -a $a -gt $c ]; then
9         echo "$a is biggest number"
10    elif [ $b -gt $a -a $b -gt $c ]; then
11        echo "$b is biggest number"
```

gungun16@GUNGUN: /mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03\$ bash q6.sh
12 53 24
53 is biggest number
gungun16@GUNGUN: /mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03\$

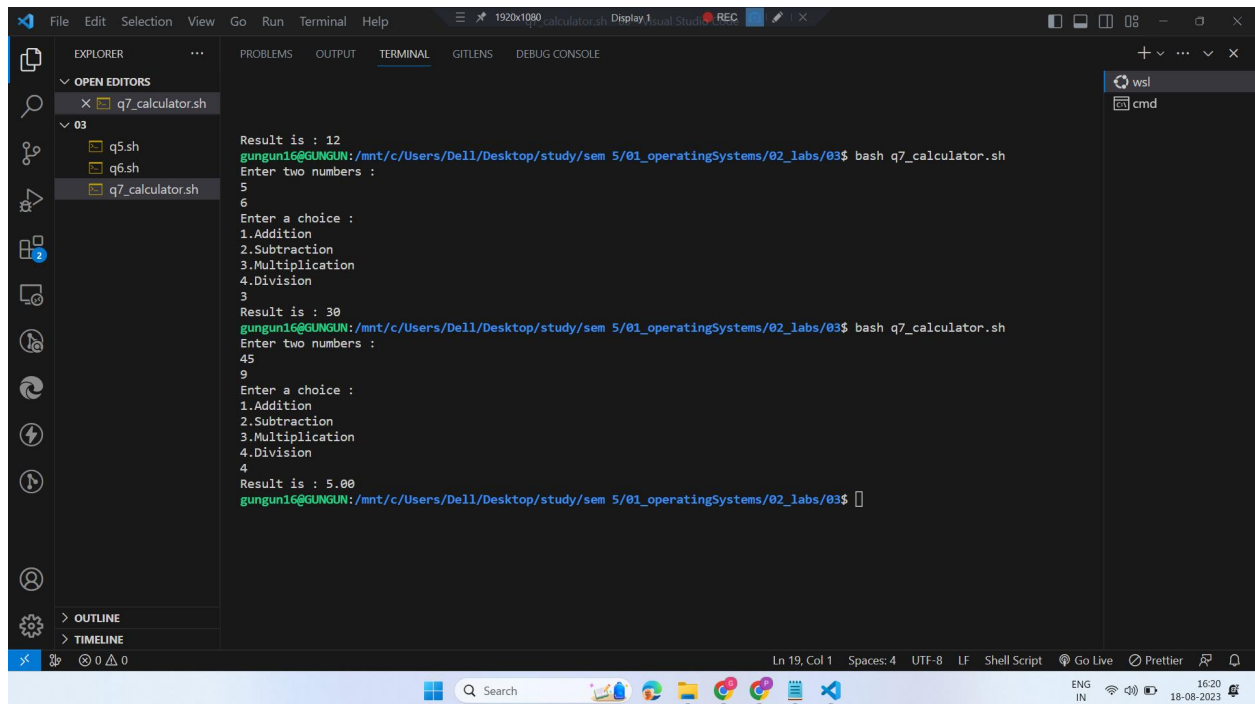
Que 6 Make a simple calculator



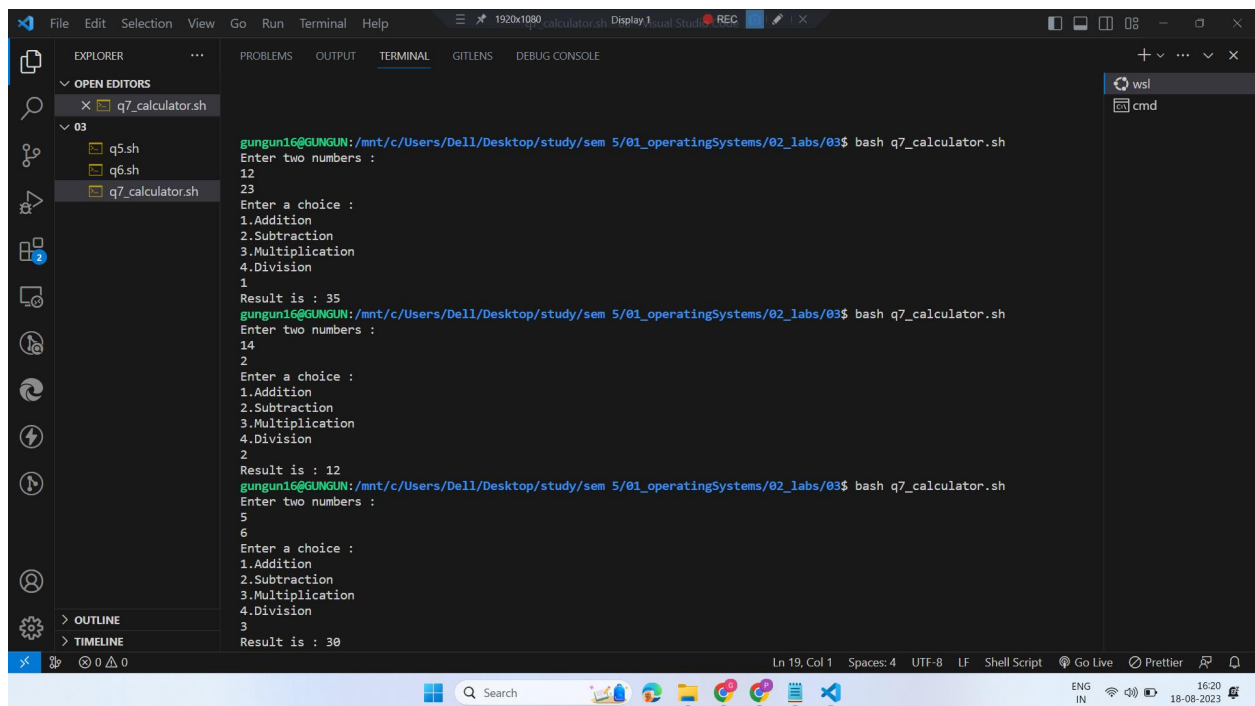
The screenshot shows the Visual Studio Code interface with a file explorer on the left and a terminal at the bottom. The file explorer shows a project named '03' containing files 'q5.sh', 'q6.sh', and 'q7_calculator.sh'. The editor displays the content of 'q7_calculator.sh', which is a shell script to create a simple calculator. The terminal is empty.

```
1 #!/bin/bash
2
3 #Take the user input
4 echo "Enter two numbers : "
5 read a
6 read b
7
8 #input type of operation
9
10 echo "Enter a choice : "
11 echo "1.Addition"
12 echo "2.Subtraction"
13 echo "3.Multiplication"
14 echo "4.Division"
15
16 read ch
17
18 #Switch case to perform => calculator operations
19
20 case $ch in
21     1)res=`echo $a + $b | bc`
22     ;;
23     2)res=`echo $a - $b | bc`
24     ;;
25     3)res=`echo $a * $b | bc`
26     ;;
27     4)res=`echo "scale=2; $a/$b" | bc`
28     ;;
29 esac
30 echo "Result is : $res"
```

Output of the different calculations :

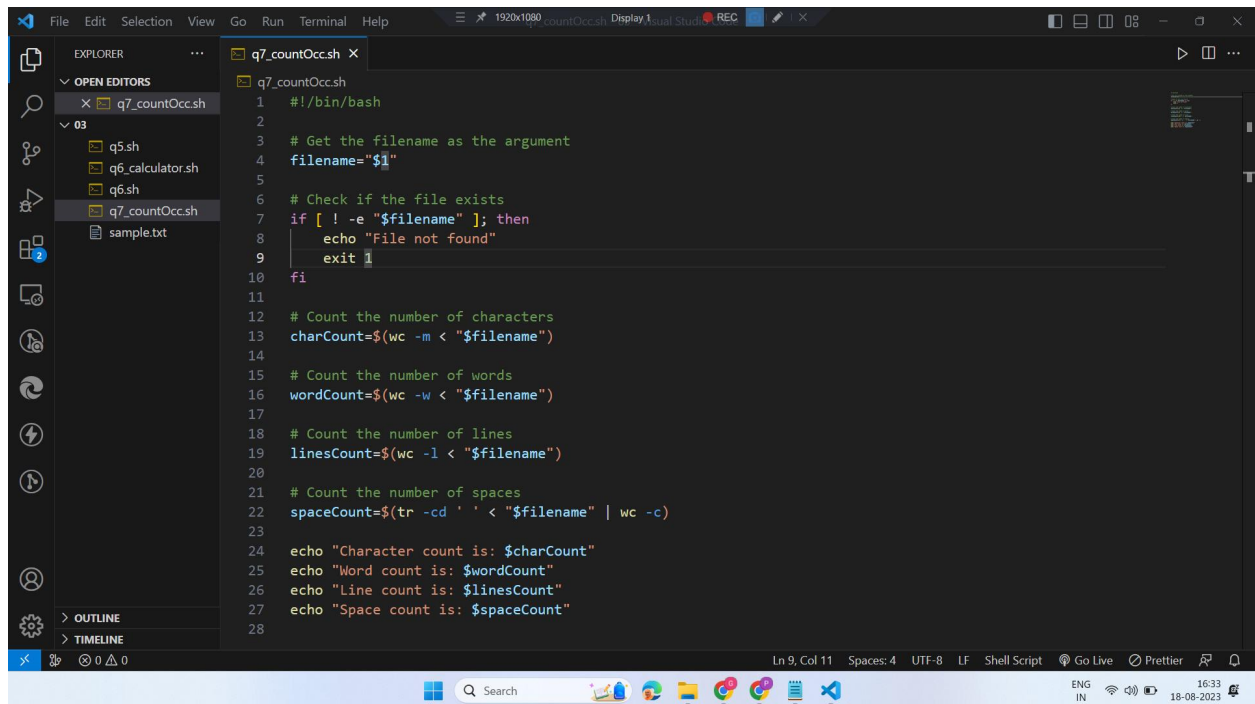


```
Result is : 12
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$ bash q7_calculator.sh
Enter two numbers :
5
6
Enter a choice :
1.Addition
2.Subtraction
3.Multiplication
4.Division
3
Result is : 30
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$ bash q7_calculator.sh
Enter two numbers :
45
9
Enter a choice :
1.Addition
2.Subtraction
3.Multiplication
4.Division
4
Result is : 5.00
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$
```



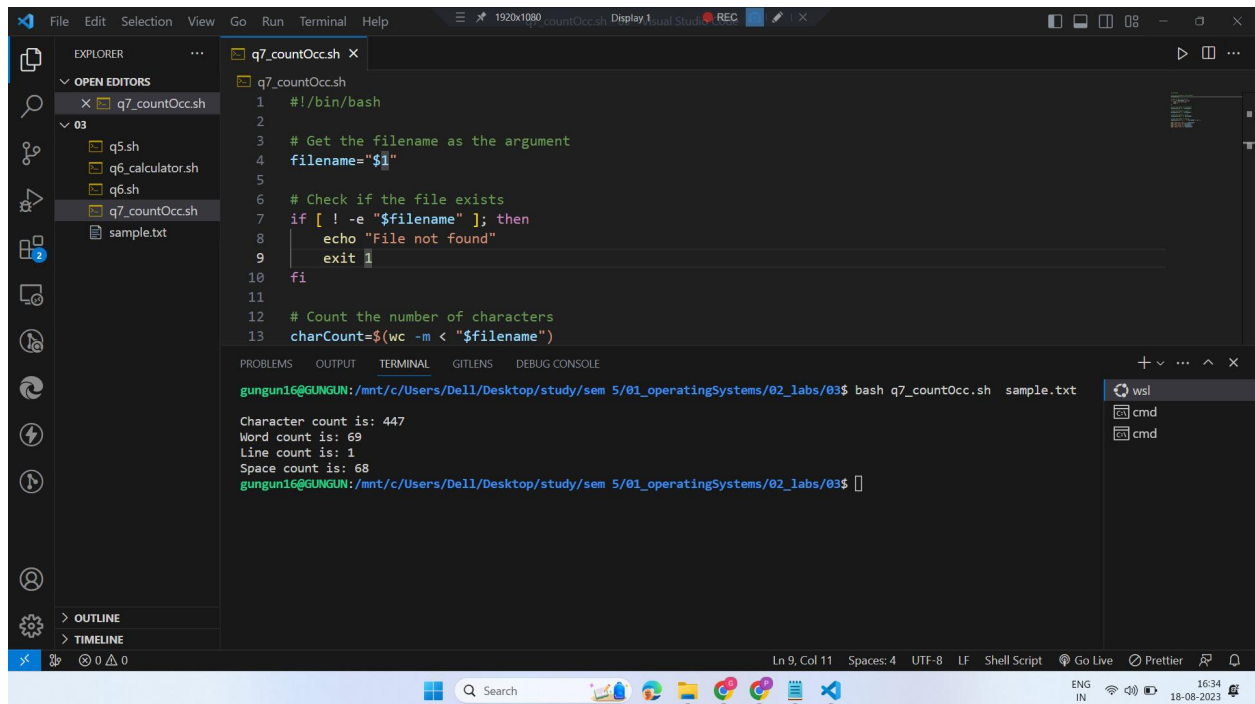
```
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$ bash q7_calculator.sh
Enter two numbers :
12
23
Enter a choice :
1.Addition
2.Subtraction
3.Multiplication
4.Division
1
Result is : 35
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$ bash q7_calculator.sh
Enter two numbers :
14
2
Enter a choice :
1.Addition
2.Subtraction
3.Multiplication
4.Division
2
Result is : 12
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$ bash q7_calculator.sh
Enter two numbers :
5
6
Enter a choice :
1.Addition
2.Subtraction
3.Multiplication
4.Division
3
Result is : 30
```

Que7. Count the number of words,spaces,characters and lines



The screenshot shows the Visual Studio Code editor with the file `q7_countOcc.sh` open. The script is a shell script that takes a filename as an argument and counts the number of characters, words, lines, and spaces in the file. The script is as follows:

```
1  #!/bin/bash
2
3  # Get the filename as the argument
4  filename="$1"
5
6  # Check if the file exists
7  if [ ! -e "$filename" ]; then
8      echo "File not found"
9      exit 1
10 fi
11
12 # Count the number of characters
13 charCount=$(wc -m < "$filename")
14
15 # Count the number of words
16 wordCount=$(wc -w < "$filename")
17
18 # Count the number of lines
19 linesCount=$(wc -l < "$filename")
20
21 # Count the number of spaces
22 spaceCount=$(tr -cd ' ' < "$filename" | wc -c)
23
24 echo "Character count is: $charCount"
25 echo "Word count is: $wordCount"
26 echo "Line count is: $linesCount"
27 echo "Space count is: $spaceCount"
28
```



The screenshot shows the Visual Studio Code editor with the file `q7_countOcc.sh` open. The terminal window is open, showing the output of the script when run on `sample.txt`. The output is as follows:

```
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$ bash q7_countOcc.sh sample.txt
Character count is: 447
Word count is: 69
Line count is: 1
Space count is: 68
gungun16@GUNGUN:/mnt/c/Users/Dell/Desktop/study/sem 5/01_operatingSystems/02_labs/03$
```