Lab Manual Book

Operating System Lab

AjayDilipKumar M

5th August, 2024

Introduction

About Shell program:

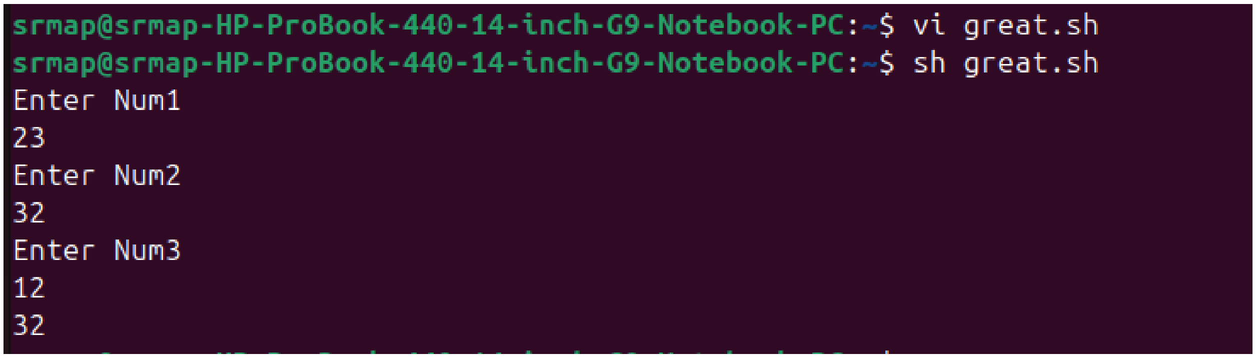
https://www.geeksforgeeks.org/introduction-linux-shell-shell-scripting/

1. Write a shell script to find the greatest of three numbers.

   Solution:

```
echo "Enter Num1"
read num1
echo "Enter Num2"
read num2
echo "Enter Num3"
read num3

if [ $num1 -gt $num2 ] && [ $num1 -gt $num3 ]
then

  echo $num1

elif [ $num2 -gt $num1 ] && [ $num2 -gt $num3 ]

then

        echo $num2

else

        echo $num3

fi
```

   Output:



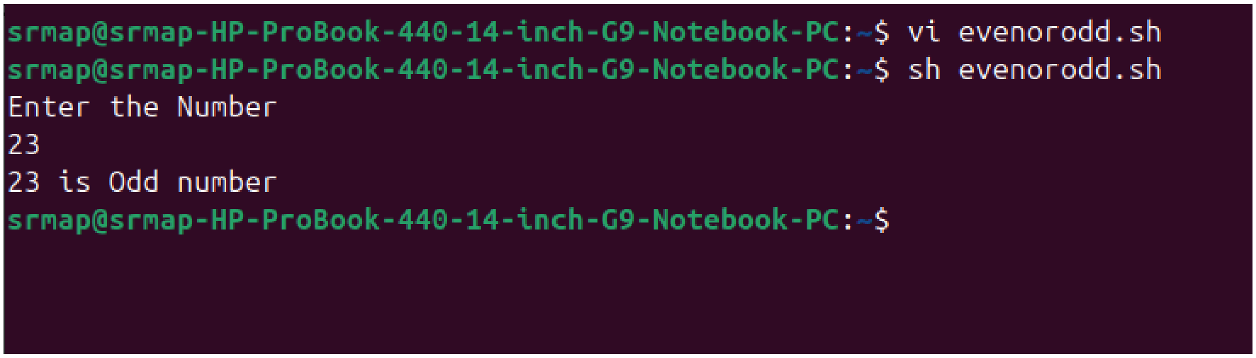
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi great.sh
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh great.sh
Enter Num1
23
Enter Num2
32
Enter Num3
12
32
```

2. Write a shell script to check whether the given no is even/odd.

   Solution:

```
echo "Enter the Number"
read n
 r=`expr $n % 2`
 if [ $r -eq 0 ]
then
echo "$n is Even number"
else
echo "$n is Odd number"
 fi
```

   Output:



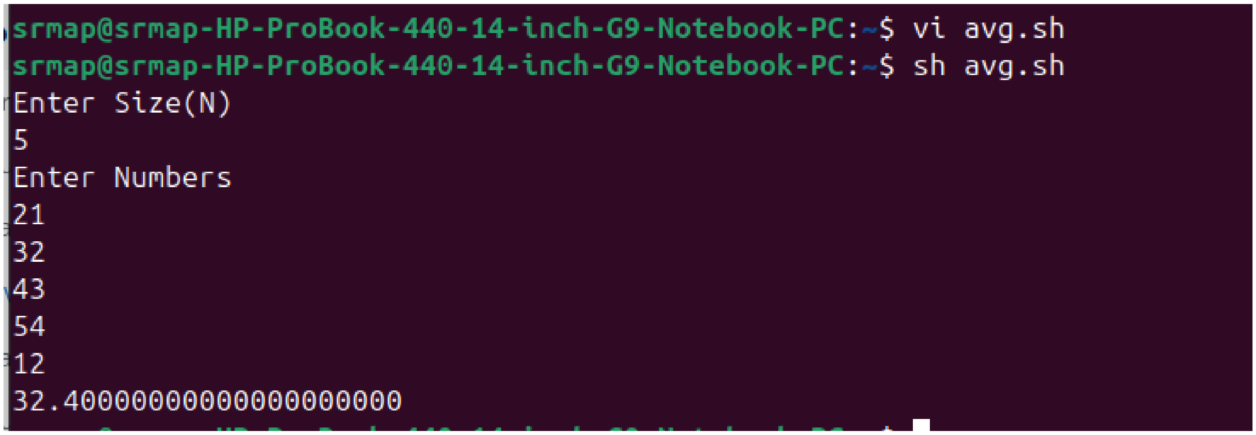
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi evenorodd.sh
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh evenorodd.sh
Enter the Number
23
23 is Odd number
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$
```

3. Write a shell script to calculate the average of n numbers.

   Solution:

```
echo "Enter Size(N)"
read N
i=1
sum=0
echo "Enter Numbers"
while [ $i -le $N ]
do
  read num            #get number
  sum=$((sum + num))    #sum+=num
  i=$((i + 1))
done
avg=$(echo $sum / $N | bc -l)
echo $avg
```

   Output:



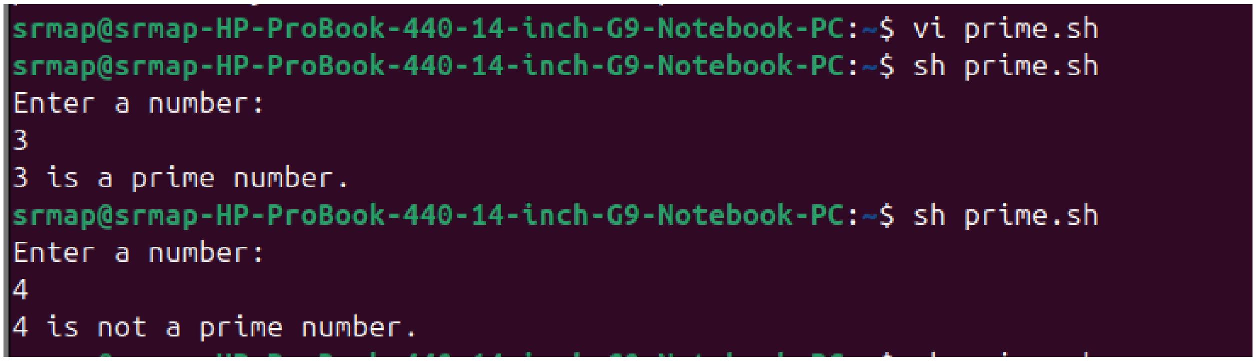
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi avg.sh
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh avg.sh
Enter Size(N)
5
Enter Numbers
21
32
43
54
12
32.40000000000000000000
```

4. Write a shell script to check whether the given number is prime or not.

Solution:

```
echo "Enter a number:"
read number
i=2

if [ $number -lt 2 ]
then
  echo "$number is not a prime number."
  exit
fi

while [ $i -lt $number ]
do
  if [ `expr $number % $i` -eq 0 ]
  then
    echo "$number is not a prime number."
    exit
  fi
  i=`expr $i + 1`
done

echo "$number is a prime number."
```

Output:



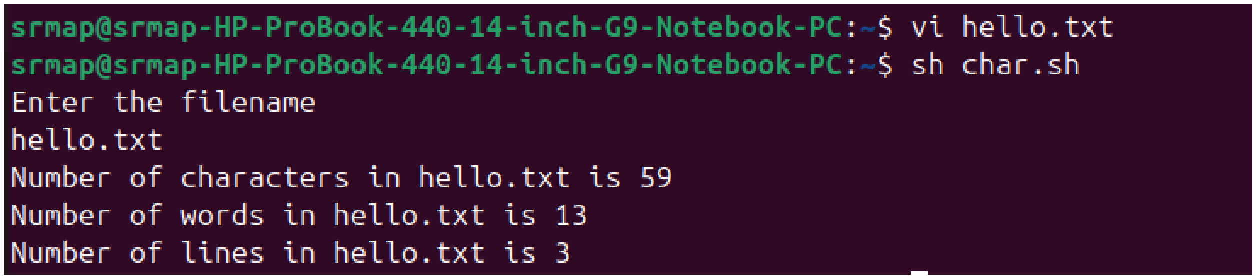
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi prime.sh
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh prime.sh
Enter a number:
3
3 is a prime number.
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh prime.sh
Enter a number:
4
4 is not a prime number.
```

5. Write a shell script to compute no. of characters and words in each line of a given file.

   Solution:

```
echo Enter the filename
read file
c=`cat $file | wc -c`
w=`cat $file | wc -w`
l=`grep -c "." $file`
echo Number of characters in $file is $c
echo Number of words in $file is $w
echo Number of lines in $file is $l
```

   Output:



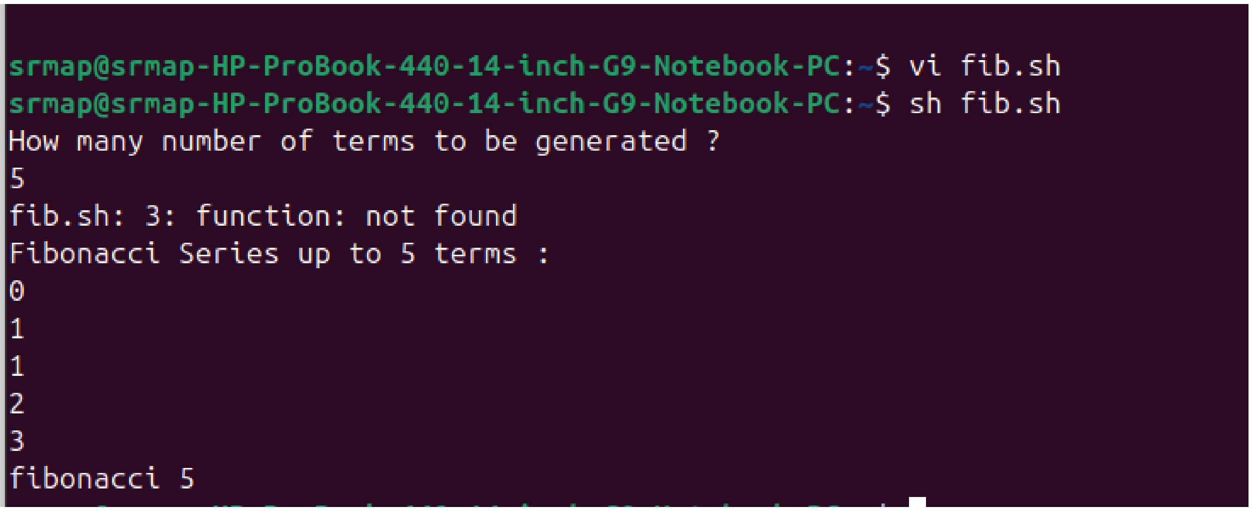
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi hello.txt
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh char.sh
Enter the filename
hello.txt
Number of characters in hello.txt is 59
Number of words in hello.txt is 13
Number of lines in hello.txt is 3
```

6. Write a shell script to print the Fibonacci series up to n terms.

Solution:

```
echo "How many numbers of terms to be generated ?"
read n
function fibonacci
{
 x=0
 y=1
 i=2
 echo "Fibonacci Series up to $n terms :"
 echo "$x"
 echo "$y"
 # -lt stands for equal to
 while [ $i -lt $n ]
 do
    i=`expr $i + 1`
    z=`expr $x + $y`
    echo "$z"
    x=$y
    y=$z
 done
}
r=”fibonacci $n”
echo "$r"
```

Output:



```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi fib.sh
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh fib.sh
How many number of terms to be generated ?
5
fib.sh: 3: function: not found
Fibonacci Series up to 5 terms :
0
1
1
2
3
fibonacci 5
```

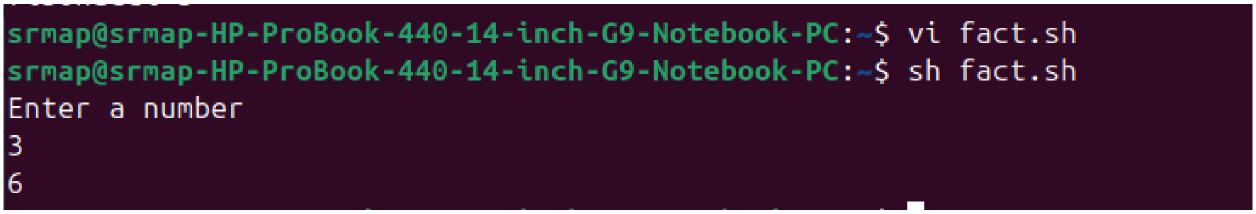
7. Write a shell script to calculate the factorial of a given number.

Solution:

```
#shell script for factorial of a number
#factorial using while loop
```

```
echo "Enter a number"
read num

fact=1

while [ $num -gt 1 ]
do
 fact=$((fact * num))  #fact = fact * num
 num=$((num - 1))     #num = num - 1
done

echo $fact
```

Output :



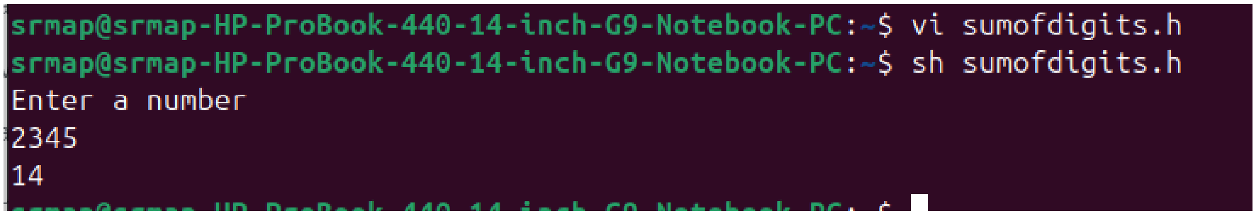
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi fact.sh
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh fact.sh
Enter a number
3
6
```

8. Write a shell script to calculate the sum of digits of a given number.

   Solution:

```
echo "Enter a number"

read num

sum=0

while [ $num -gt 0 ]

do

   mod=$((num % 10))    #It will split each digits

   sum=$((sum + mod))   #Add each digit to sum

   num=$((num / 10))    #divide num by 10.

done
echo $sum
```

Output:



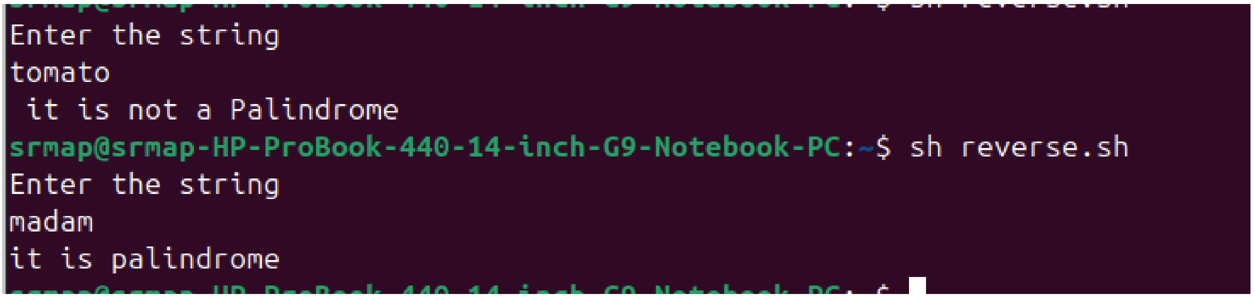
```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ vi sumofdigits.h
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh sumofdigits.h
Enter a number
2345
14
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$
```

9. Write a shell script to check whether the given string is palindrome or not.

Solution :

```
echo Enter the string
read s
echo $s>temp
rvs="$(rev temp)"
if [ $s = $rvs ]
then
echo "it is palindrome"
else
echo " it is not a Palindrome"
fi
```

Output:



```
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh reverse.sh
Enter the string
tomato
 it is not a Palindrome
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$ sh reverse.sh
Enter the string
madam
it is palindrome
srmap@srmap-HP-ProBook-440-14-inch-G9-Notebook-PC:~$
```

Conclusion