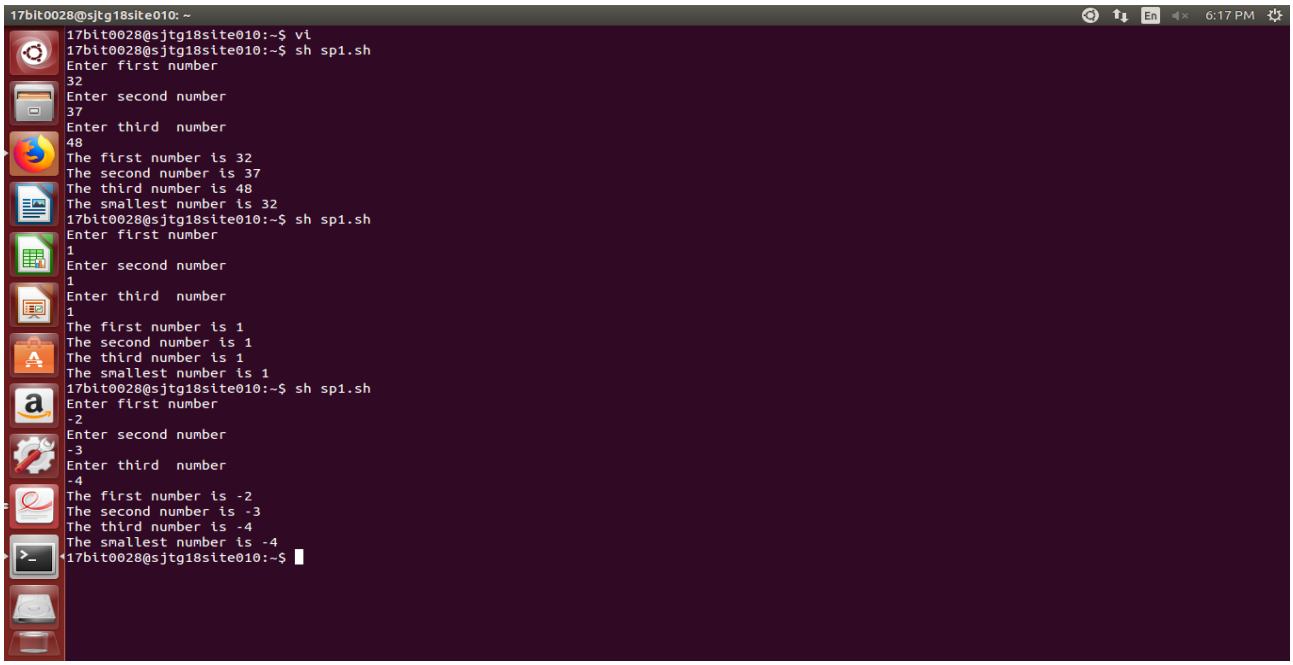
SCHOOL OF INFORMATION TECHNOLOGY AND ENGINEERING
B.Tech IT
OPEARTING SYSTEMS

17BIT0028
SIDDHI SINGH

Shell Programs

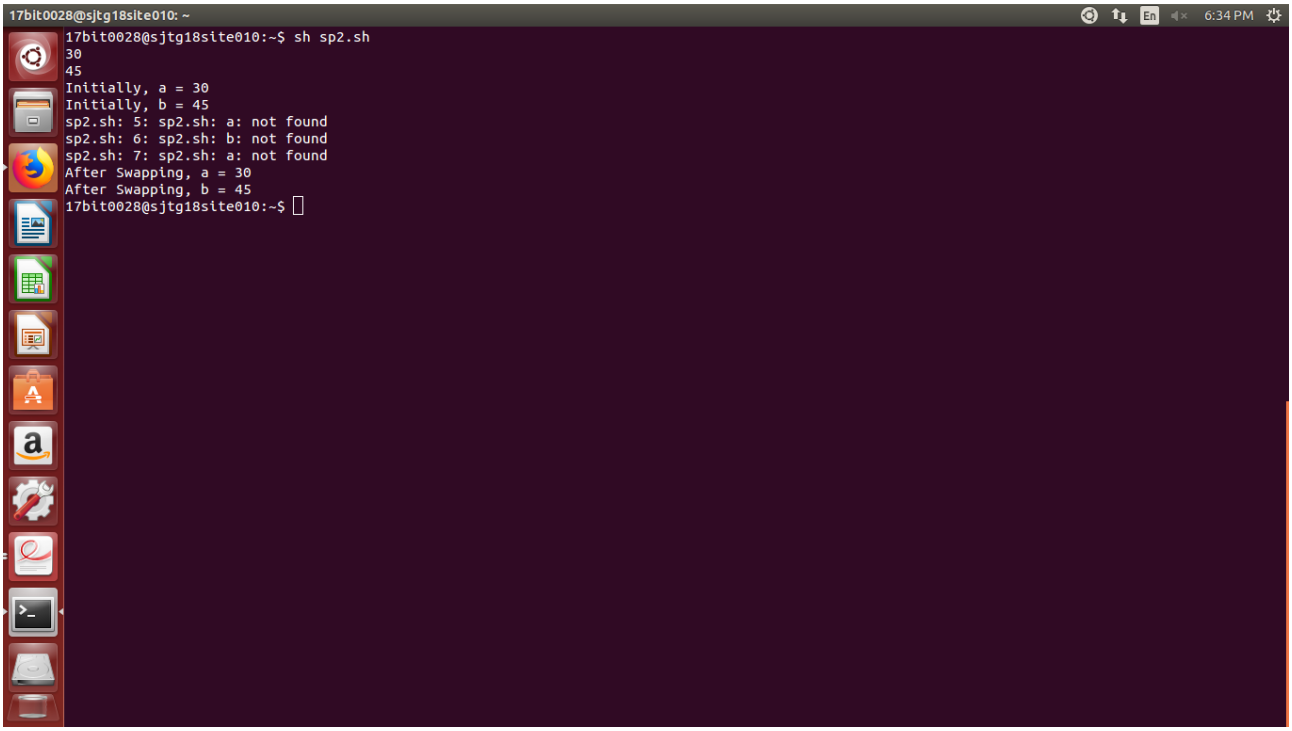
Shell Program 1

```
echo "Enter 1st numbner"
read a
echo "Enter 2nd number "
read b
echo "Enter 3rd number "
read c
echo "the 1st number is $a"
echo "the 2nd number is $b"
echo "the 3rd number is $c"
if [ $a -lt $b ]
then d=$a
else d=$b
fi
if [ $c -lt $d ]
then d=$c
fi
echo "The smallest number is $d"
```



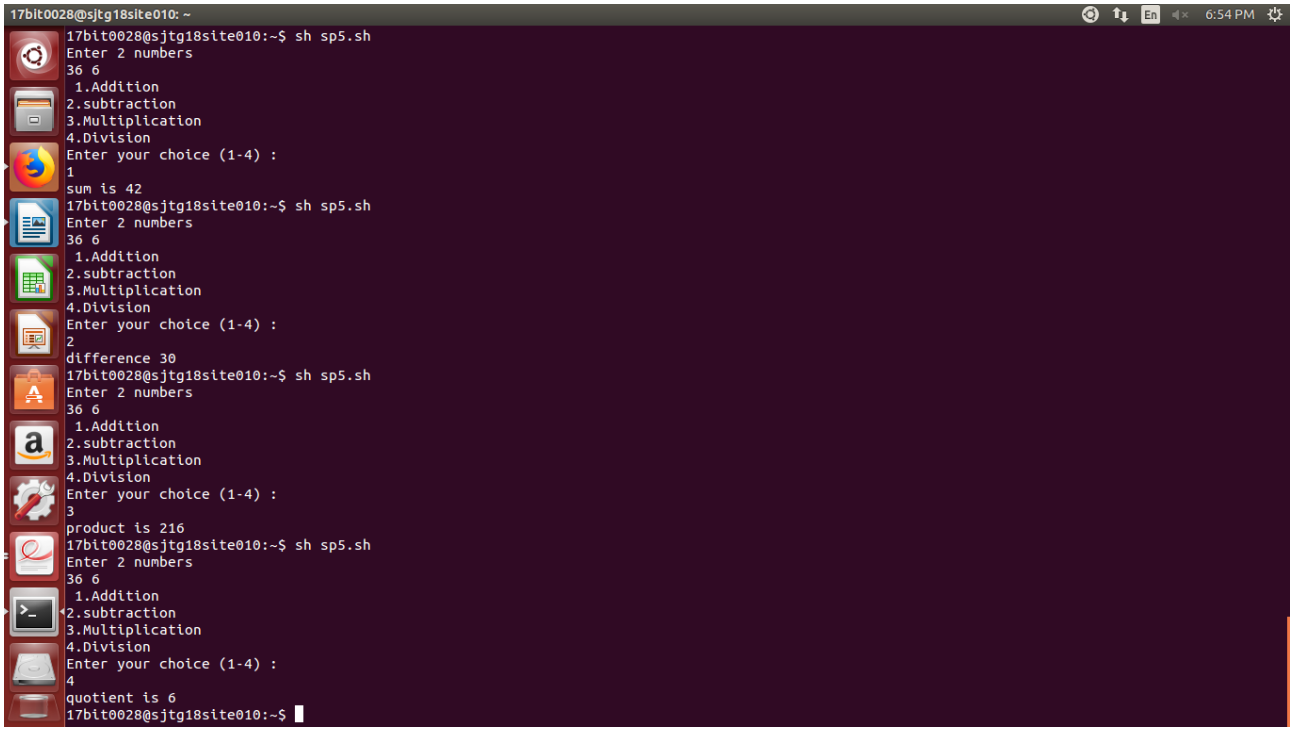
Shell Program 2

```
read a
read b
echo " Initially, a = $a "
echo " Initially, b = $b "
a=`expr $a + $b`
b=`expr $a - $b`
a=`expr $a - $b`
echo " After Swapping, a = $a "
echo " After Swapping, b = $b "
```



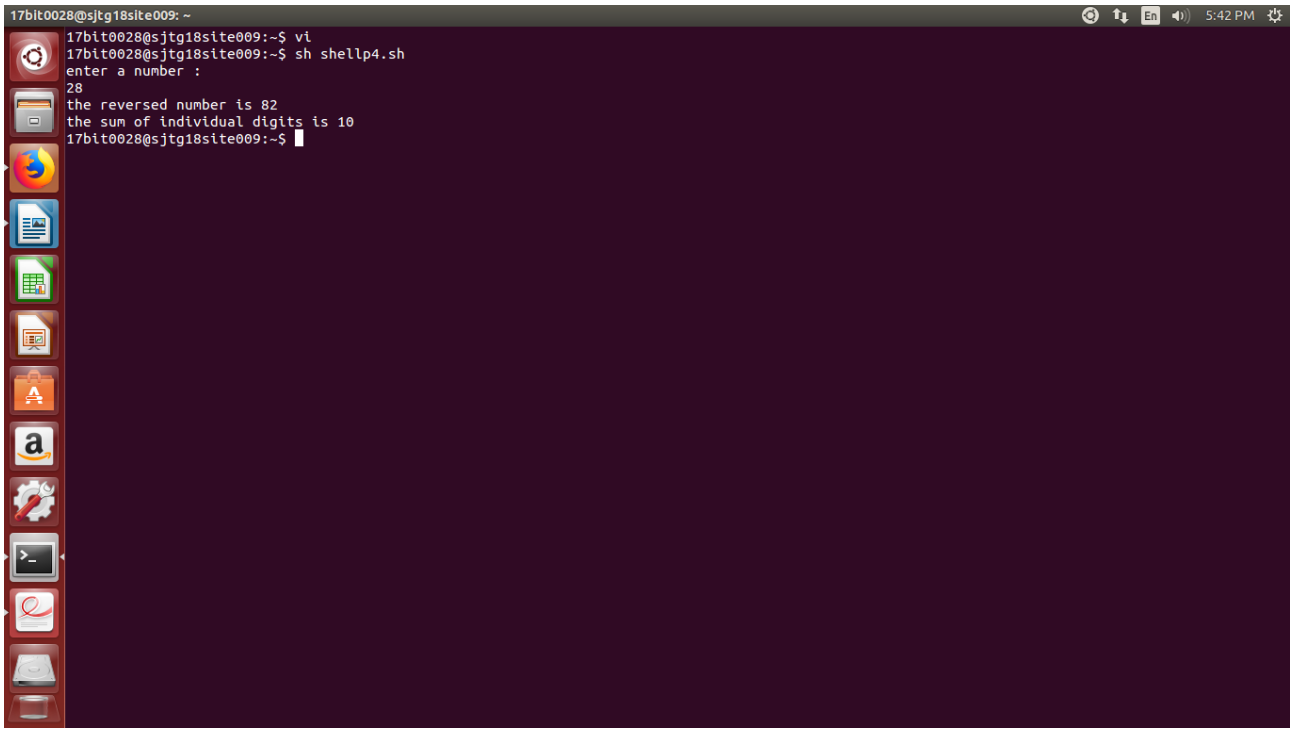
Shell Program 3

```
echo Enter 2 numbers
read a b
echo " 1.Addition
2.subtraction
3.Multiplication
4.Division
Enter your choice (1-4) : "
read c
case $c in
1) echo sum is `expr $a + $b`
;;2) echo difference `expr $a - $b`
;;
3) echo product is `expr $a \* $b`
;;
4) echo quotient is `expr $a / $b`
;;
esac
```



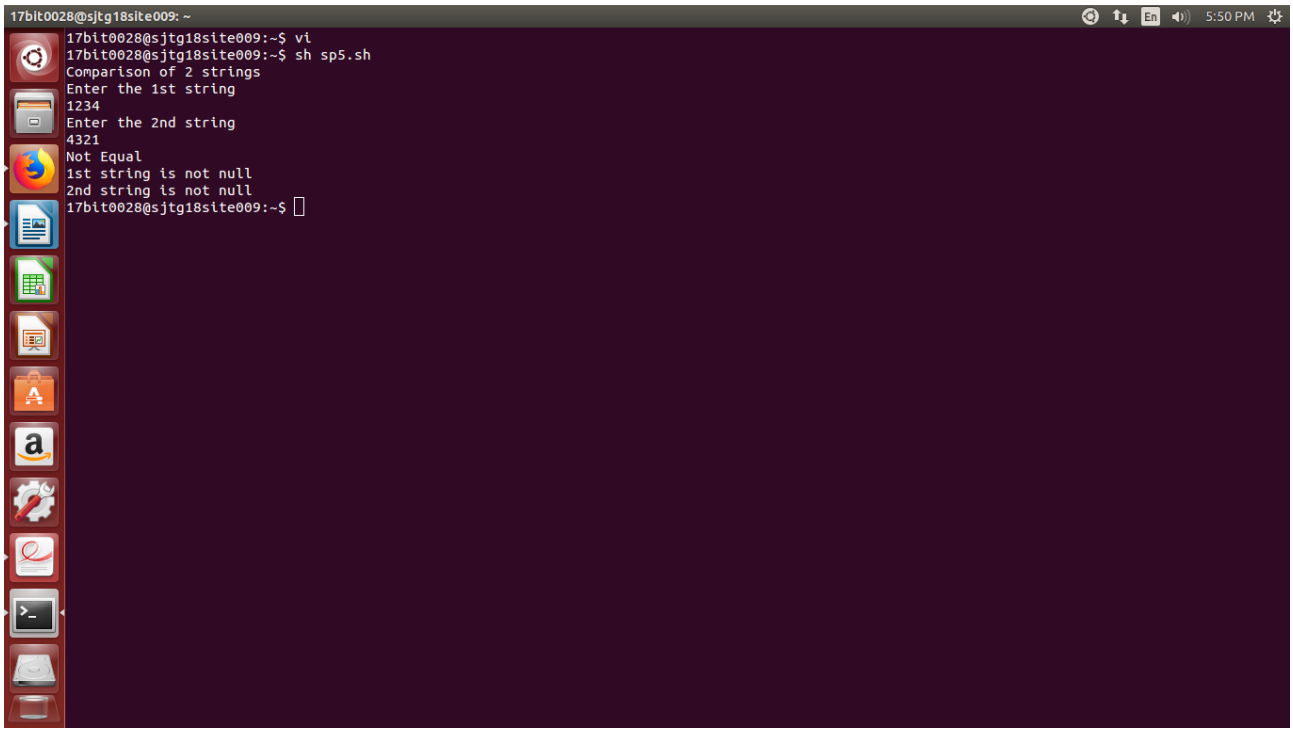
Shell Program 4

```
echo "enter a number :"
read n
sum1=0
sum=0
while [ $n -ne 0 ]
do
r=`expr $n % 10`
n=`expr $n / 10`
sum=`expr $sum \* 10`
sum=`expr $sum + $r`
sum1=`expr $sum1 + $r`
done
echo the reversed number is $sum
echo the sum of individual digits is $sum1
```



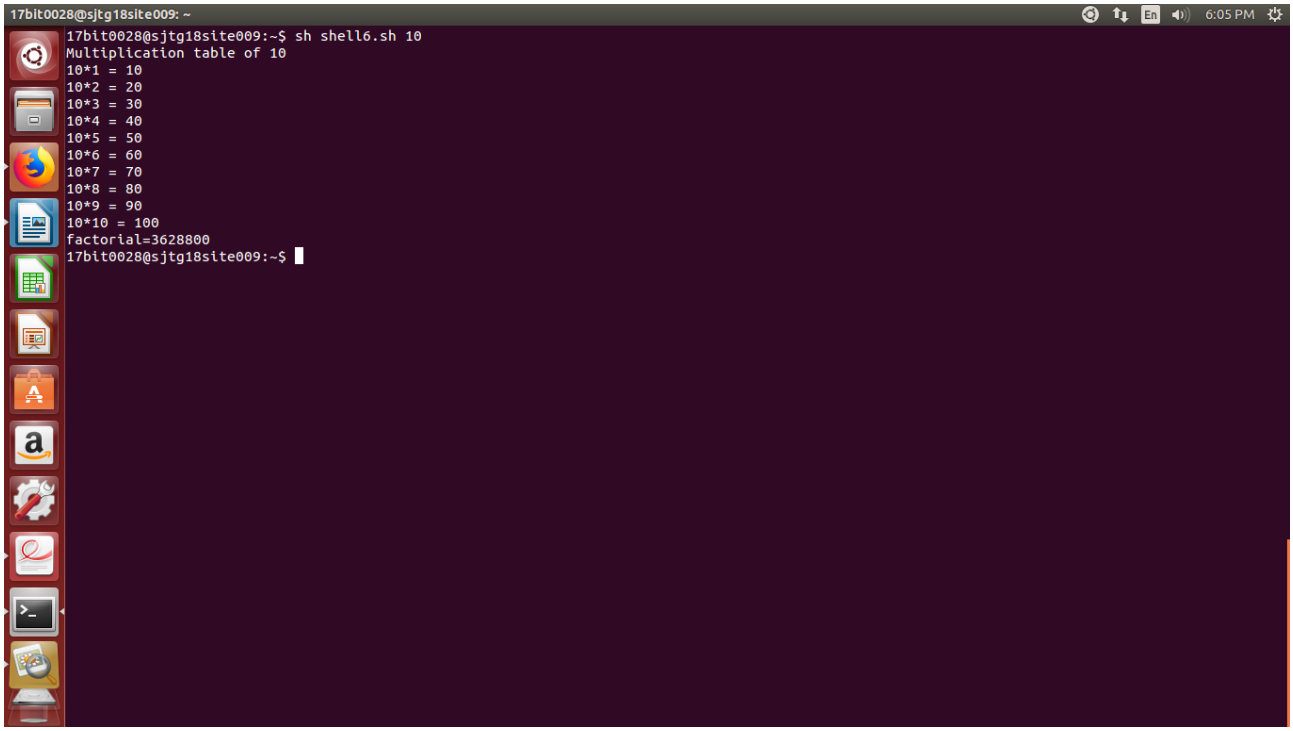
Shell Program 5

```
echo Comparison of 2 strings
echo Enter the 1st string
read s1
echo Enter the 2nd string
read s2
if [ $s1 = $s2 ]
then echo Equal
else echo Not Equal
fi
if [ -z $s1 ]
then echo 1st string is null
else echo 1st string is not null
fiif [ -z $s2 ]
then echo 2nd string is null
else echo 2nd string is not null
fi
```



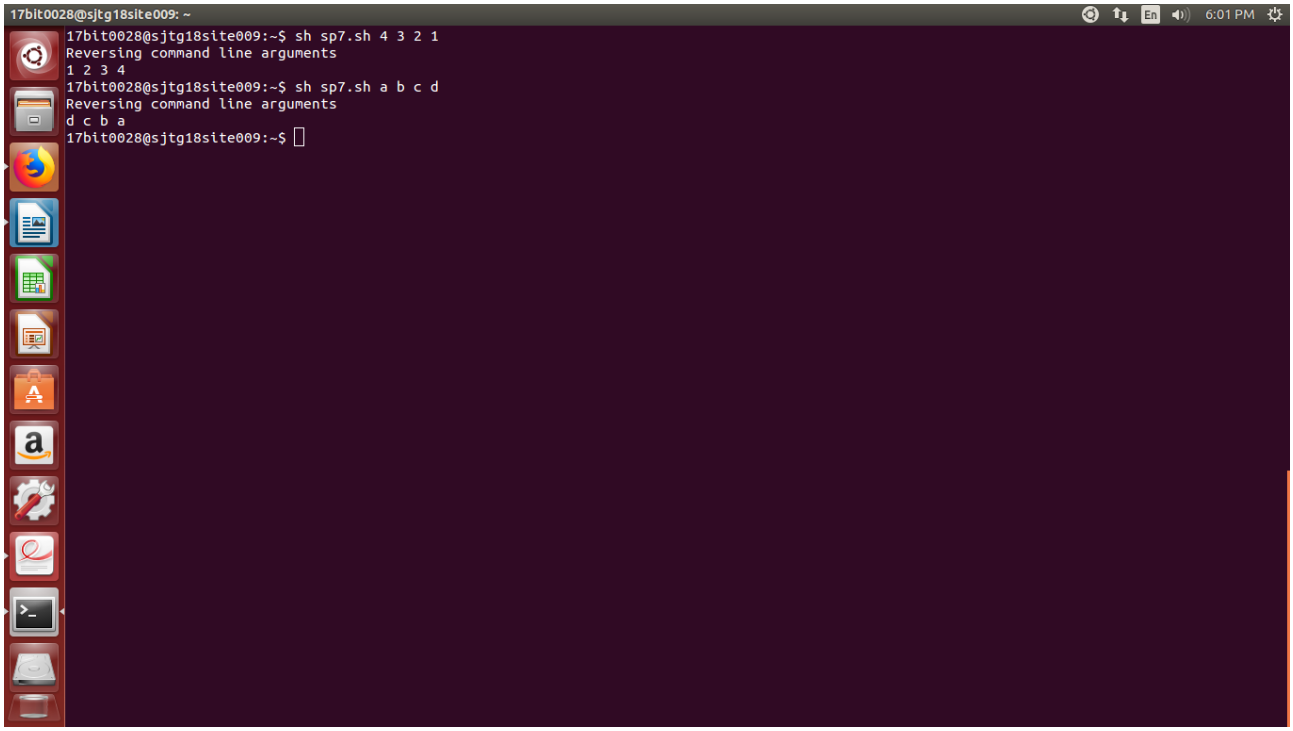
Shell Program 6

```
echo Multiplication table of $1
c=`expr $1 - 1`
for (( i=1 ; i<=10 ; i++ ))
do
echo $1*$i = `expr $1 \* $i`
done
c=1
for(( i=1 ; i<=$1 ; i++ ))
do
c=`expr $c \* $i`
done
echo factorial=$c
```



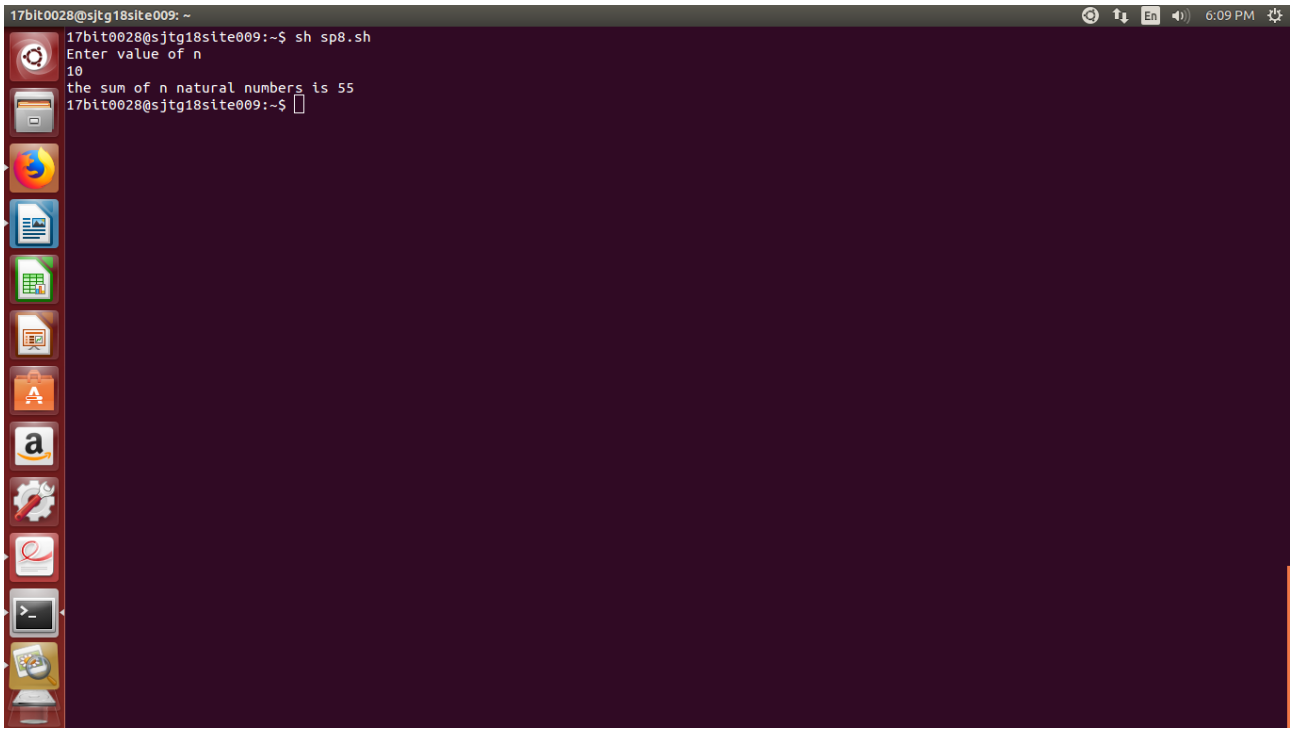
Shell Program 7

```
echo Reversing command line arguments
for i in $*
do
a=$i
b="$a $b"
done
echo $b
```



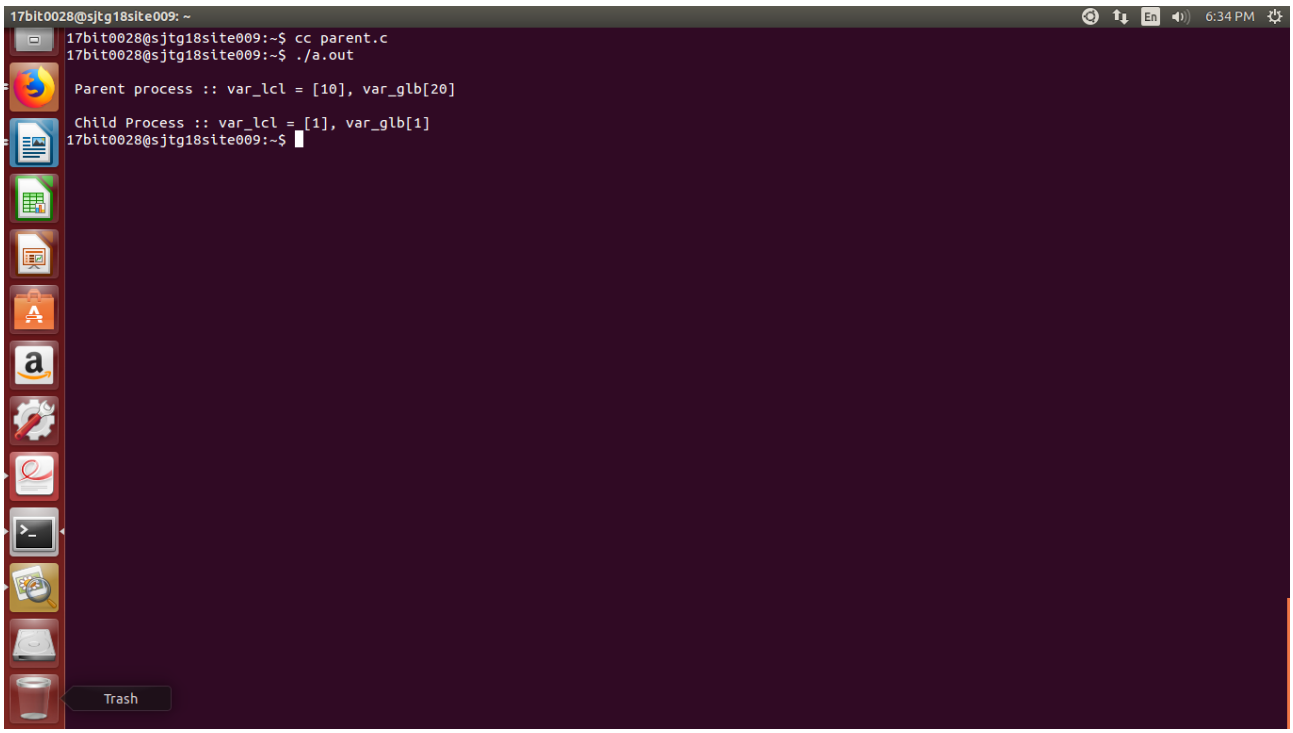
Shell Program 8

```
echo Enter value of nread n
c=1
s=0
while [ $c -le $n ]
do
s=`expr $s + $c`
c=`expr $c + 1`
done
echo "the sum of n natural numbers is $s"
```



PROCESSES

```
#include <unistd.h>
#include <sys/types.h>
#include <errno.h>
#include <stdio.h>
#include <sys/wait.h>
#include <stdlib.h>

int var_glb; /* A global variable*/

int main(void)
{
   pid_t childPID;
   int var_lcl = 0;

   childPID = fork();

   if(childPID >= 0) // fork was successful

   {
      if(childPID == 0) // child process
      {
         var_lcl++;
         var_glb++;
         printf("\n Child Process :: var_lcl = [%d], var_glb[%d]\n", var_lcl, var_glb);
      }
      else //Parent process
      {
         var_lcl = 10;
         var_glb = 20;
         printf("\n Parent process :: var_lcl = [%d], var_glb[%d]\n", var_lcl, var_glb);
      }
   }
   else // fork failed
   {
      printf("\n Fork failed, quitting!!!!!!\n");
      return 1;
   }
 return 0;
}
```


17bit0028@sjtg18site009: ~
En
6:34 PM
17bit0028@sjtg18site009:~$ cc parent.c
17bit0028@sjtg18site009:~$ ./a.out
Parent process :: var_lcl = [10], var_glb[20]
Child Process :: var_lcl = [1], var_glb[1]
17bit0028@sjtg18site009:~$
Trash