**PROGRAM CODE:**

#!/bin/bash

#biggest of 3 nos

if [ $# -ne 3 ]

then

echo Insufficient no.of arguments

exit 1

fi

n1=$1

n2=$2

n3=$3

if [ $n1 -gt $n2 ] && [ $n1 -gt $n3 ]

then

echo $n1 is the greatest number

elif [ $n2 -gt $n1 ] && [ $n2 -gt $n3 ]

then

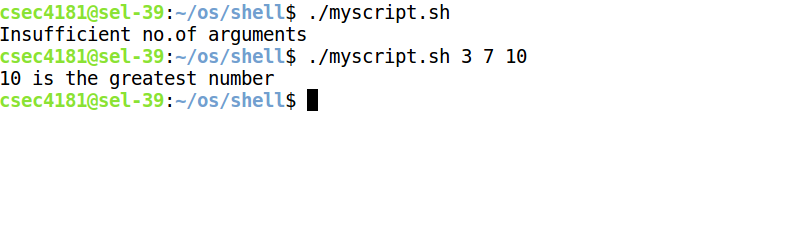
echo $n2 is the greatest number

else

echo $n3 is the greatest number

fi

**OUTPUT:**

**PROGRAM CODE:**

#!/bin/bash

#print 5,4,3,2,1

i=5

while [ $i -ge 1 ]

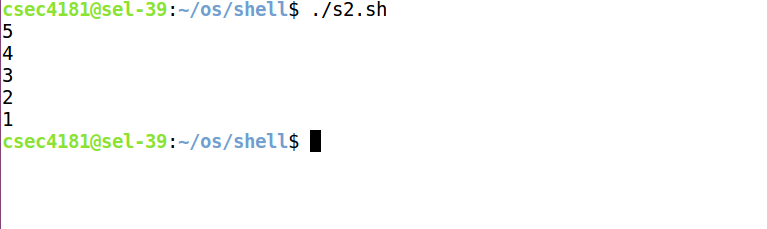
do

echo $i

i=$(( i-1 ))

done

**OUTPUT:**

****

**PROGRAM CODE:**

#!/bin/bash

#math operations using case

if [ $# -ne 3 ]

then

echo Insufficient no.of arguments

exit 1

fi

case $2 in

+) r=`expr $1 + $3` ;;

-) r=`expr $1 - $3` ;;

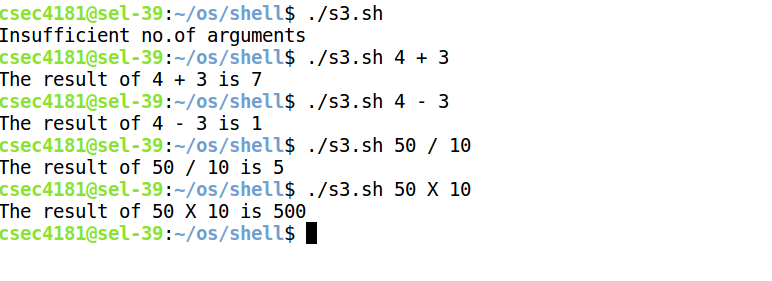
/) r=`expr $1 / $3` ;;

X) r=`expr $1 \\\* $3` ;;

esac

echo The result of $1 $2 $3 is $r

**OUTPUT:**

****

**PROGRAM CODE:**

#!/bin/bash

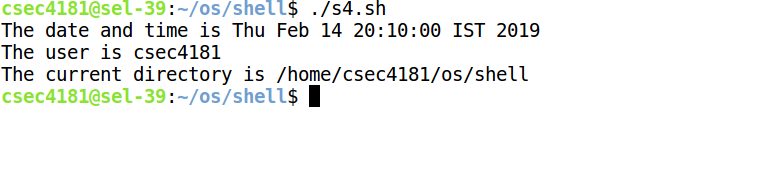
#print date time username current directory

echo The \date and time is `date`

echo The user is `whoami`

echo The current directory is `pwd`

**OUTPUT:**

****

**PROGRAM CODE:**

#!/bin/bash

#pattern

for i in {0..5}

#(( i=0;i<5;i++ ))

do

for (( j=0;j<i+1;j++ ))

do

echo -n "\* "

done

echo " "

#echo -e "\n"

done

echo -e "\n"

n=9

for ((i=1;i<=n;i++))

do

for ((j=n;j>=i;j--))

do

echo -n " "

done

for ((k=1;k<=i;k++ ))

do

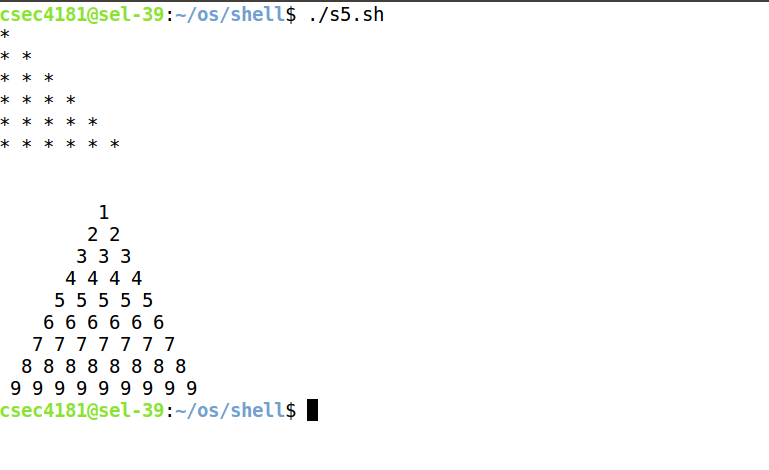
echo -n "$i "

done

echo " "

done

**OUTPUT:**

****

**PROGRAM CODE:**

#!/bin/bash

#file exist or not

if [ $# -ne 2 ]

then

if [ -e $1 ]

then

echo File $1 exists

else

echo File $1 does not exist

fi

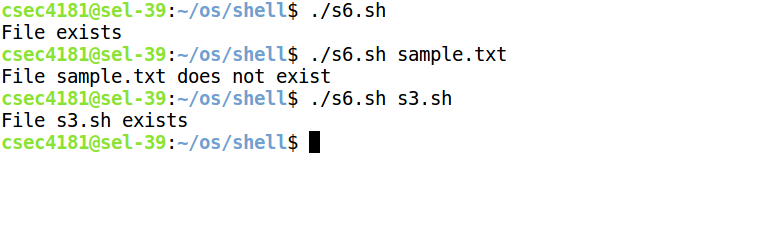
exit 1

else

echo Insufficient no.of arguments

fi

**OUTPUT:**

****

**PROGRAM CODE:**

if [ $# -ne 3 ]

then

echo Insufficient no. of arguments

exit 1

fi

if [ -e $3 ];

then

tail -n +$1 $3 | head -n $2

else

echo "Error opening file"

fi

**OUTPUT:**

****