Assignment 3

a1)

echo -n "Enter file name: "

read file

if [ test -e $file ];

then

echo "The file:- $file exists"

ls -a -l |grep $file

else

echo "Creating new file:- $file"

touch $file

ls -a -l | grep $file

fi

op

Enter file name: a4

Creating new file:- a4

-rw-r--r-- 1 user user 12288 Oct 8 09:20 .a4.sh.swp

-rw-r--r-- 1 user user 0 Oct 8 09:37 a4

-rw-r--r-- 1 user user 107 Oct 8 09:19 a4.sh

a2)

echo -n "Enter a floating point number: "

read a

echo -n "Enter another floating point number: "

read b

echo -n "$a + $b = "

echo `echo $a + $b` | bc

echo -n "$a - $b = "

echo `echo $a - $b` | bc

echo -n "$a \* $b = "

echo `echo $a\*$b` | bc

echo -n "$a / $b = "

echo `echo $a / $b` | bc

op

Enter a floating point number: 1.2

Enter another floating point number: 2.4

1.2 + 2.4 = 3.6

1.2 - 2.4 = -1.2

1.2 \* 2.4 = 2.8

1.2 / 2.4 = 0

a3)

echo "Enter 1st no : "

read a

echo "Enter 2nd no : "

read b

if [ $a -gt $b ]

then

echo "$a is Greater number"

else

echo "$b is Greater number"

fi

op

Enter 1st no :

5

Enter 2nd no :

1

5 is Greater number

a4)

echo "Enter a number"

read a

if [ `expr $a % 2` -eq 0 ];

then

echo "$a is even"

else

echo "$a is odd"

fi

op

Enter a number

4

4 is even

b1)

echo -n "To merge enter 1 or to find pattern pattern 2: "

read opt

case $opt in

1)

echo "merge two files here: "

echo -n "Enter File name 1: "

read file1

echo -n "Enter file name 2: "

read file2

echo -n "Enter Destination file name: "

read file

cat $file1 $file2 >> $file

cat $file;;

2)

echo "Find pattern from file"

echo -n "Enter file name: "

read file

echo -n "Enter the pattern you want to find: "

read pattern

grep $pattern $file;;

\*)

echo "invalid option";;

esac

op

To merge enter 1 or to find pattern pattern 2: 2

Find pattern from file

Enter file name: a3.sh

Enter the pattern you want to find: if

if [ $a -gt $b ]

b2)

echo -n "Enter the file name: "

read file

echo -n "Enter starting line number: "

read start

echo -n "Enter number of lines: "

read gap

total=`expr $start + $gap - 1`

head -$total $file | tail -$gap

op

Enter the file name: a3.sh

Enter starting line number: 2

Enter number of lines: 2

read a

echo "Enter 2nd no : "

b3)

echo "Menu"

echo "Option 1: Check number of presently active users"

echo "Option 2: See the top n number of lines of a file"

echo "Option 3: Update the access time of a file"

echo -n "Enter your choice"

read opt

case $opt in

1)

users

;;

2)

echo -n "Enter the name of the file: "

read file

echo -n "Enter the number of lines: "

read num

head -$num $file

;;

3)

echo -n "Enter the file name: "

read file

touch $file

echo "file updated"

echo "Updated Information - "

ls -l | grep $file

;;

\*)

echo "Invalid option!"

esac

op

Menu

Option 1: Check number of presently active users

Option 2: See the top n number of lines of a file

Option 3: Update the access time of a file

Enter your choice2

Enter the name of the file: a3.sh

Enter the number of lines: 4

echo "Enter 1st no : "

read a

echo "Enter 2nd no : "

read b

b4)

echo "Enter the text"

read text

echo

echo "No of words"

echo $text | wc -w

echo

echo "No of characters"

echo $text | wc -m

echo

space=$(expr length "$text" - length `echo "$text" | sed "s/ //g"`)

echo "No of whitespaces $space"

echo

special=$(expr length "${text//[^\~!@#$&\*()]/}")

echo "No of special symbols $special"

echo

op

Enter the text

M$@##$ ^$$^ 533 gygyafd 433%^

No of words

5

No of characters

30

No of whitespaces 4

No of special symbols 7

C1)

max=`cut -d ' ' -f1 numbers`

min=`cut -d ' ' -f1 numbers`

for num in `cat numbers`

do

if [ $num -gt $max ]

then

max=$num

fi

done

for num in `cat numbers`

do

if [ $num -lt $min ]

then

min=$num

fi

done

echo "The max elements is $max"

echo "The min element is $min"

op

10 20 30 40 50 60 70 80 90 99 5

The max elements is 99

The min element is 5

c2)

s=0

echo "Enter a number :"

read n

t=$n

while [ $n -ne 0 ]

do

r1=$(($n % 10))

rrr=$((r1 \* r1 \* r1))

s=$((s + rrr))

n=$(($n / 10))

done

if [ $t -eq $s ]

then

echo "$t is a ARMSTRONG number"

else

echo "$t is a NON ARMSTRONG number"

fi

op

Enter a number :

153

153 is a ARMSTRONG number

c3)

echo "Enter 1st number "

read n1

echo "Enter 2nd number "

read n2

echo "Enter 3rd number "

read n3

for i in $n1 $n2 $n3;

do

for j in $n1 $n2 $n3;

do

for k in $n1 $n2 $n3;

do

echo "$i" "$j" "$k"

done

done

done

op

Enter 1st number

6

Enter 2nd number

1

Enter 3rd number

3

6 6 6

6 6 1

6 6 3

6 1 6

6 1 1

6 1 3

6 3 6

6 3 1

6 3 3

1 6 6

1 6 1

1 6 3

1 1 6

1 1 1

1 1 3

1 3 6

1 3 1

1 3 3

3 6 6

3 6 1

3 6 3

3 1 6

3 1 1

3 1 3

3 3 6

3 3 1

3 3 3

c4)

echo "Enter how many "

read n

x=0

y=1

i=2

echo "Fibonacci Series "

echo "$x"

echo "$y"

while [ $i -lt $n ];

do

i=$(expr $i + 1)

z=$(expr $x + $y)

echo "$z"

x=$y

y=$z

done

op

Enter how many

6

Fibonacci Series

0

1

1

2

3

5

c5)

echo "Enter how many "

read n

i=0

f=0

f1=1

f2=1

while [ $i -le $n ] ;

do

if [ $i -ge $f ];

then

f1=$f2

f2=$f

f=`expr $f1 + $f`

if [ $i -ne $f ];

then

i=`expr $i + 1`

fi

else

echo $i

i=`expr $i + 1`

fi

done

op

Enter how many

15

4

6

7

9

10

11

12

14

15

c6)

echo "Enter a number :"

read n

f=1

i=2

while [ $i -le $n ]

do

f=$((f \* i))

i=`expr $i + 1`

done

echo "Factorial of $n is : "

echo $f

op

Enter a number :

5

Factorial of 5 is :

120