**USP Lab**

**Week 11**

**Agenda: grep, Arrays and Functions**

1. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it. [use grep]
2. Write a shell scripts which performs the following
3. Add an element to the existing array
4. Remove an element from the array
5. Find the length of the nth element in the array
6. Length of the whole array
7. Copy an array
8. Concatenate two arrays
9. Write a shell script which performs factorial of a number using functions. (optional, can be skipped)
10. Write a shell script to find the maximum of two integers using functions. Write a function which handles error.

**1.**

if [ $# -lt 1]

then

echo “ Check the arguments once”

exit

fi

echo “Enter a word”

read word

for file in $\*

do

grep –iv “$word” $file > temp && mv temp $file

done

echo “ lines containing given word are deleted”

2.

c=(violet indigo blue green yellow orange red)

k=(cyan safron purple)

echo "Program on Arrays"

echo "

1) Add an element to the existing array

2) Remove an element from the array

3) Find the length of the nth element in the array

4) Length of the array

5) Copy an array

6) Concatenate two arrays"

echo "Enter your choice"

read ch

case $ch in

1) echo "enter the element to be added in the array"

read c1

c=("${c[@]}" "$c1")

echo ${c[@]}

;;

2) echo "Enter the position to be removed"

read p1

unset ${c[$p1]}

echo ${c[@]}

;;

3) echo "Enter the position to find the length"

read n

echo "The length of the "$n"th element ${c[$n]} is ${#c[$n]}"

;;

4) len=${#c[\*]}

echo "The length of the array c is $len"

;;

5) d=("${c[@]}")

echo "The array c is copied to d and d is ${d[@]}"

;;

6) cate=("${c[@]}" "${k[@]}")

echo ${cate[@]}

;;

\*) echo " Invalid Choice"

;;

esac

3.

factorial()

{

if [ "$1" -gt "1" ]; then

i=`expr $1 - 1`

j=`factorial $i`

k=`expr $1 \\* $j`

echo $k

else

echo 1

fi

}

while :

do

echo "Enter a number:"

read x

factorial $x

done

4.

USG\_ERR=7

max\_two ( ) {

if [ "$1" -eq "$2" ] ; then

echo 'Equal'

exit 0

elif [ "$1" -gt "$2" ] ; then

echo $1

else

echo $2

fi

}

err\_str ( ) {

echo "Usage: $0 <number1> <number2>"

exit $USG\_ERR

}

NUM\_1=$1

NUM\_2=$2

x

if [ $# -ne 2 ] ; then

err\_str

elif [ `expr $NUM\_1 : '[0-9]\*'` -eq ${#NUM\_1} ] ; then

if [ `expr $NUM\_2 : '[0-9]\*'` -eq ${#NUM\_2} ] ; then

max\_two $NUM\_1 $NUM\_2

else

err\_str

fi

else

err\_str

fi

exit 0