**USP Lab**

**Week 7**

1. Write a shell script to print the multiplication table where the table number has to be interactive input and length of the table is always 10.
2. Write a shell script that accepts two integers as its arguments and computes the value of first number raised to the power of the second number.
3. Write a shell script that reads a directory name from the user and lists all files only which the user has read, write and execute permissions in the given directory.
4. Write a shell script that receives even number of filenames as its arguments and copies the contents of the files at the odd-numbered positions on to the files at the following even-numbered positions. If odd number of files are supplied then copying does not takes place, instead error message will be displayed.
5. Create a script file that reads the filename and output its properties like its permission, hardlink count, file size, owner etc…
6. Write a Shell script to change the Attributes of a file using chmod command upon user choice on symbolic method or octal method and then gets the input correspondingly and changes the file permissions accordingly.
7. Write a shell script to list the files and directory of a user specified directory without an ls statement. The program also reports an error if it is not a directory.
8. Write a script where a password is stored in a variable. The program allows user to enter the password and checks for it. The script has to be executed until a correct guess is made.
9. password="open"
10. answer=""
11. until test "$answer" = "$password"
12. do
13. echo "Guess the password to quit the program"
14. read answer
15. done
16. Write a shell script that accepts filenames as arguments and prints them in the reverse order.

1.

echo Enter the table:

read table

for i in $table

do

for j in {1..10}

do

value=`expr $i \\* $j`

echo -n "$i \* $j = $value "

done

echo

done

2.

i=$1

k=$2

pow=1

if [ $k -lt 0 ]

then

echo "The second number is a negative number. It will be converted to positive number"

echo

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

fi

j=1

while [ $j -le $k ]

do

pow=`expr $pow \\* $i`

j=`expr $j + 1`

done

echo "Power of $1^$2 is :$pow"

3.

echo "enter the directory name"

read dir

if [ -d $dir ]

then

cd $dir

for i in \*

do

if [ -f "$i" ]

then

if [ -r $i -a -w $i -a -x $i ]

then

echo "$i has all permissions"

else

echo "$i not having all permissions"

fi

fi

done

else

echo "$dir is not a directory"

fi

4.

rem=`expr $# % 2`

if [ $rem -ne 0 ]

then

echo “Filenames as arguments should be given”

else

while [ “$\*” != “ “ ]

do

src=$1

desc=$2

cp $src $des

shift 2

done

fi

5

echo Enter the filename

read file

a=`ls -l $file`

echo $a

set $a

echo Enter the choice

read ch

case $ch in

1 ) echo "file permission $1";;

2 )echo "link info $2";;

3 )echo "owner info $3";;

4 )echo "group info $4";;

5 )echo "file size $5";;

6 )echo "date of creation $6";;

7 )echo "time $7";;

8 )echo "file name $8";;

\* )echo "invald option"

esac

6.

n=1  
while [ $n -ne 0 ]  
do  
echo "Menu"  
echo "1. To change the permission normally"  
echo "2. To change the permission using octal notation"  
echo " Enter the choice"  
read a  
case "$a" in  
1 ) echo "Enter the name of the file"  
read f1  
echo "Enter 1 to set the permission"  
echo "Enter 2 to remove the permission"  
read c  
echo "Enter the group of user we want to change permission"  
echo "u-users"  
echo "g-group"  
echo "o-others"  
echo "a-all"  
read f2  
echo "Enter the permission we want to set"  
echo "r-read"  
echo "w-write"  
echo "x-execute"  
read f3  
if test $c -eq 1  
then  
chmod $f2+$f3 $f1  
elif test $c -eq 2  
then  
chmod $f2-$f3 $f1  
else  
echo "invalid option"  
fi  
echo "The permission are as follows"  
ls -l $f1 ;;  
2 ) echo "Enter the name of the file"  
read f1  
echo "Menu"  
echo "0. to remove all permission"  
echo "1. to set execute permission"  
echo "2. to set write permission"  
echo "3. to set write and execute permission"  
echo "4. to set read permission"  
echo "5. to set read and execute permission"  
echo "6. to set read and write permission"  
echo "7. to set all permissions"  
ls -l $f1  
echo "Enter the permission we want to change for each group of users"  
echo "User"  
read uu  
echo "Group"  
read gg  
echo "Others"  
read oo  
  
chmod $uu$gg$oo $f1  
echo "The permission for $f1 are as follows"  
ls -l $f1 ;;  
\*) echo "invalid choice" ;;  
esac  
echo "Do u want to continue(1 YES / 0 NO )"  
read n  
done