**IV BCA Unix Lab Manual**

**Part - A**

1. **Shell script to find a factorial of a given number.**

clearecho "enter a number"read ni=1if [ $n -lt 0 ]; then echo " you cannot find factorial for a negative number"else if [ $n -eq 0 ]; then echo "factorial of zero is 1"else fact=1 while [ $i -le $n ] do fact=`expr $fact \\* $i` i=`expr $i + 1` done echo "the factorial of $n is $fact"fifi

1. **Shell script To print all prime numbers between m and n.**

clear

echo " Enter the range m and n , m <n :"

read m n

echo " The given range is from $m to $n"

echo "The prime numbers are "

while [ $m -le $n ]

do

i=2

flag=1

while [ $i -lt $m ]

do

if [ `expr $m % $i` -eq 0 ];

then

flag=0

break

else

i=`expr $i + 1`

fi

done

if [ $flag -eq 1 ];

then

echo $m

fi

m=`expr $m + 1`

done

1. **Shell script to reverse a given number and check whether it is palindrome or not.**

#To check whether the given number is palindrome or not

clear

echo "Enter a number"

read num

original=$num

rev=0

while [ $num -gt 0 ]

do

dig=`expr $num % 10`

rev=`expr $rev \\* 10 + $dig`

num=`expr $num / 10`

done

echo "Reverse is : $rev"

if [ $original -eq $rev ];

then

echo "$original is a palindorme"

else

echo "$original is not a palindorme"

fi

1. **Shell script to find maximum and minimum of a given set.**

#Shell script to find maximum and minimum of a list of numbers

clear

echo "How many entries"

read n

i=1

echo "\n Enter the first number:"

read num1

max=$num1

min=$num1

echo "Enter `expr $n - 1` numbers"

while [ $i -lt $n ]

do

read num

if [ $num -gt $max ]

then

max=$num

fi

if [ $num -lt $min ]

then

min=$num

fi

i=`expr $i + 1`

done

echo "Maximum value = $max"

echo "Minimum value = $min"

1. **Shell script to count total number of vowels in a given string.**

clear

echo "enter the string"

read str

len=`echo $str | wc -c`

len=`expr $len - 1`

i=1;

while [ $i -le $len ]

do

ch=`echo $str | cut -c $i`

case $ch in

a | A) count=`expr $count + 1`;;

e | E) count=`expr $count + 1`;;

i | I) count=`expr $count + 1`;;

o | O) count=`expr $count + 1`;;

u | U) count=`expr $count + 1`;;

esac

i=`expr $i + 1`

done

echo "Total number of vowels in the given string are : $count "

1. **Write a menu driven program to calculate**

**i) Simple interest ii) Compound interest**

#to calculate simple and compound interest

clear

tput cup 5 10

echo "Bank Interest calculation form"

tput cup 6 10

echo "---------------------------"

tput cup 8 10

echo "1. Simple Interest"

tput cup 10 10

echo "2. Compound Interest"

tput cup 12 10

echo "3. Exit"

tput cup 15 10

echo "Enter your choice:"

tput cup 15 30

read choice

clear

case $choice in

1) echo "Enter principle, time and rate"

read p t r

si=`echo "scale=2; $p \* $t \* $r / 100" | bc`

echo "Principle = Rs. $p"

echo "Time = $t years"

echo "Rate = $r %"

echo "Simple Interest = Rs. $si";;

2) echo "Enter principle rate and time"

read p t r

amt=`echo "scale=2; $p \* ( 1 + $r / 100 ) ^ $t " | bc`

ci=`echo "scale=2; $amt - $p " | bc`

echo "Principle = Rs. $p"

echo "Time = $t years"

echo "Rate = $r %"

echo "Compound interest = Rs. $ci" ;;

\*) echo "Wrong choice"

exit;;

esac

1. **Shell script to create a file contains the following fields : student no, student name , age, sex , height and weight. Print all the details in neat format.**

#Shell script to accept student details and print in proper format

clear

echo "Enter number of records"

read n

i=1

while [ $i -le $n ]

do

echo "Enter student registration number ( 4 digits )"

read num

echo "Enter Student Name :"

read name

echo "Enter Student age"

read age

echo "Enter Male/Female"

read sex

echo "Enter student height"

read ht

echo "Enter Student weight"

read wt

echo "$num|$name|$age|$sex|$ht|$wt" >> stdlist

i=`expr $i + 1`

done

echo "\*\*\*\*\*\*\* Student Information \*\*\*\*\*\*\*\*\*"

echo

printf "Regno \t Name \t Age \t Sex \t Height \t Weight \n"

echo "==================================================="

awk -F "|" '{ printf("%5d %-10s %3d %5s %8d %8d \n",$1,$2,$3,$4,$5,$6)}' stdlist

echo "==================================================="

1. **Shell script to create emp file containg empname, empno, deptno and designation.**
2. **Display empname and empno of any particular dept and the count of employees.**
3. **Display empname and empno of employees who are not managers**.

clear

echo "Enter number of records"

read n

i=1

while [ $i -le $n ]

do

echo "Enter employee no:\c"

read empno

echo "Enter employee name :\c"

read ename

echo "Enter Department:\c"

read dept

echo "Enter Designation:\c"

read desg

echo "$empno|$ename|$dept|$desg" >> emplist

i=`expr $i + 1`

done

echo "\*\*\*\*\*\*\* Employee Information \*\*\*\*\*\*\*\*\*"

echo

printf "Empno \t Name \t Department \t Designation\n"

echo "==================================================="

awk -F "|" '{ printf("%5d %-10s %-15s %-10s \n",$1,$2,$3,$4)}' emplist

echo "==================================================="

echo

#Query 1 - to count number of employees in given department

echo "Enter the employee department : \c"

read sdept

awk -v se=$sdept -F "|" 'BEGIN {count=0} { if( $3==se ) { print $1,$2;count++ }} END {printf("Number of employees in %s department are : %d\n",se, count)}' emplist

#Query 2 - to display employee names and numbers who are not managers

echo "The employees who are not managers"

awk -F "|" ' {if ( $4 != "manager" ) print $1,$2}' emplist

1. **Shell script to create two data file and compare them to display unique and common entries.**

#To find the common and unique entries

clear

echo "enter the name of file1"

read file1

echo "enter the contents of $file1"

cat >$file1

echo " enter the name of file2"

read file2

echo "enter the contents of $file2"

cat >$file2

#Sort both the files before using comm command

sort -o $file1 $file1

sort -o $file2 $file2

echo

clear

echo "The items of $file1 are"

cat $file1

echo "The items of $file2 are"

cat $file2

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

echo "The items that are unique to $file1 are"

comm -23 $file1 $file2

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

echo "The items that are unique to $file2 are :"

comm -13 $file1 $file2

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

echo "The items that are common to both are :"

comm -12 $file1 $file2

echo "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*"

1. **Shell script to display all file names and file permissions.**

#To display the file names and their permissions

clear

#find out if file has write permission or not

echo " The files and their permissions are"

echo "File name Read Write Execute"

for file in \*

do

if [ -f $file ];

then

if [ -w $file ] ;

then

W="Yes"

else

W="No"

fi

if [ -r $file ] ;

then

R="Yes"

else

R="No"

fi

if [ -x $file ];

then

X="Yes"

else

X="No"

fi

printf "%-20s %6s %6s %6s\n" $file, $R, $W, $X

fi

done

**\*\*\*\*\*\*\*\***