String based shell Programming

1. Write a Linux shell script to count number of characters, words & lines, UNIX program to count number of characters, words & lines.

clear  
echo-n "Enter file name:"  
read fn

c='wc-c<$fn  
w='wc-w<$fn  
l='wc-l<$fn

echo"NO. of characters:"$C  
echo"No.of Words:"$w  
echo"No.of Lines:"$1

1. Write a shell script to identify the given string is palindrome or not .

echo Enter the string  
read s  
echo $s > temp  
rvs="$(rev temp)"  
if [ $s = $rvs ]  
then  
echo "it is palindrome"  
else  
echo " it is not"  
fi

1. Another example of palindrome.

#!/usr/bin/ksh  
len=0  
i=1  
echo -n "Enter a String: "  
read str  
len=`echo $str | wc -c`  
len=`expr $len - 1`  
halfLen=`expr $len / 2`  
while [ $i -le $halfLen ]  
do  
c1=`echo $str|cut -c$i`  
c2=`echo $str|cut -c$len`  
if [ $c1 != $c2 ] ; then  
echo "string is not palindrome"  
exit  
fi  
i=`expr $i + 1`  
len=`expr $len - 1`  
done  
echo "String is Palindrome"

1. Shell program to reverse a string

#!/bin/bash

read –p "Enter string:" string

len=${#string}

for (( i=$len-1; i>=0; i-- ))

do

# "${string:$i:1} "extract single single character from string.

reverse="$reverse${string:$i:1}"

done

echo "$reverse

in unix:   
echo "hello world" |sed 's/./&\n/g' |tac |tr -d '\n'

in linux:   
echo "hello world" |rev   
dlrow olleh

1. To print the message when the length of the string is either less than 10 or more than 10

echo "Enter string "

read str

len=`echo $str | wc -c`

if [ $len -le 10 ]

then

echo "LEss then 10"else echo "More than 10"

fi

echo "Length of string is " $len

#!/bin/sh

a="abc"

b="efg"

if [ $a = $b ]

then

echo "$a = $b : a is equal to b"

else

echo "$a = $b: a is not equal to b"

fi

if [ $a != $b ]

then

echo "$a != $b : a is not equal to b"

else

echo "$a != $b: a is equal to b"

fi

if [ -z $a ]

then

echo "-z $a : string length is zero"

else

echo "-z $a : string length is not zero"

fi

if [ -n $a ]

then

echo "-n $a : string length is not zero"

else

echo "-n $a : string length is zero"

fi

if [ $a ]

then

echo "$a : string is not empty"

else

echo "$a : string is empty"

1. Write a Linux shell script to check whether the entered number is Armstrong number or not using the command line argument, Unix program for Armstrong number

clear  
no=$1  
n=$1  
while[ $n-gt 0]  
do  
a='expr $n%10'  
a3='expr$a\\*$a\\*$a'  
temp='expr$temp+ $a3'  
n='expr$temp+$a3'  
done  
if[$temp-eq$no]  
then  
  echo"It is Armstrong number"  
else  
  echo"It is Armstrong number"  
fi

1. Write a Linux shell script to generate fibonacci number from 1 to n, unix program for fibonacci series

clear  
echo "Enter the number upto which you want series :"  
read n  
n1=0  
n2=1  
echo $n1  
echo $n2  
for ((i=0;i<n;i++))  
do  
  n3='expr $n1 + $n2 '  
  echo$n3  
  n1= $n2  
  n2= $n3  
 done

1. Write a shell program to concatenate two strings and find the length of the resultant string

Echo “Enter first string:”  
Read s1   
Echo “Enter second string:”  
Read s2  
s3 = $s1$s2  
len = `Echo $s3 | wc -c`  
len = `expr $len – 1`  
Echo “Concatenated string is $s3 of length $len ”

1. This example shows how to translate the contents of a variable and display the result on the screen with tr

#!/bin/sh

# Translate the contents of a variable

Cat\_name="Piewacket"

echo $Cat\_name | tr 'a' 'i'