

EX.NO:3

SHELL PROGRAMMING

Date : 12.08.2024

3.1 SUM OF “n” NATURAL NUMBERS

AIM: To write a shell script to find the sum of “N” natural numbers.

ALGORITHM:

Step1: Start the program.

Step2: Enter the number.

Step3: Assign “S” is equal to zero and i is even to one.

Step4: Using while loop calculate the sum, display the sum.

Step5: Stop the program.

PROGRAM:

```
echo "Enter n:"
read n
i=1
sum=0
while [ $i -le $n ]
do
    sum=`expr $sum + $i`
    i=`expr $i + 1`
done
echo "The sum is: $sum"
```

OUTPUT:

```
enter n 10
the sum is : 55
```

RESULT:

Thus a shell script to find the sum of “N” natural numbers was executed successfully.

3.2 FIBONACCI SERIES

AIM:

To write a shell script to generate Fibonacci series.

ALGORITHM:

- Step1: Start the program.
- Step2: Enter the number.
- Step3: Initialize “b” and “d” is equal to zero.
- Step4: Using while loop, the Fibonacci series is generated.
- Step5: Terminate the program.

PROGRAM:

```
#!/bin/bash
echo "Enter the number of terms in the Fibonacci series:"
read n
a=0
b=1
count=0
echo "Fibonacci series up to $n terms:"
while [ $count -lt $n ]; do
    echo $a
    fn=$((a + b))
    a=$b
    b=$fn
    count=$((count + 1))
done
echo "Program terminated."
```

OUTPUT:

```
$ ./fibonacci.sh
Enter the number of terms in the Fibonacci series:
10
Fibonacci series up to 10 terms:
0
1
1
2
3
5
8
13
21
34
Program terminated.
```

RESULT:

Thus a shell script to generate Fibonacci series was executed successfully.

3.3. CALCULATE EMPLOYEE'S PAYROLL

AIM:

To write the shell script to find net and gross pay.

ALGORITHM:

Step1: Start the program.
Step2: Enter the basic pay.
Step3: Calculate HRA, PF and DA.
Step4: Calculate gross pay, using HRA, PF and DA.
Step5: Calculate the net pay.
Step6: Stop the program.

PROGRAM:

```
#!/bin/bash
echo "Enter basic pay:"
read basic_pay
echo "Enter DA (Dearness Allowance):"
read da
echo "Enter HRA (House Rent Allowance):"
read hra
echo "Enter PF (Provident Fund):"
read pf
gross_pay=$(expr $basic_pay + $da + $hra)
net_pay=$(expr $gross_pay - $pf)
echo "Net Pay: $net_pay"
echo "Gross Pay: $gross_pay"
```

OUTPUT:

```
$ ./pay_calculator.sh
Enter basic pay:
10000
Enter DA (Dearness Allowance):
500
Enter HRA (House Rent Allowance):
2000
Enter PF (Provident Fund):
1800
Net Pay: 10700
Gross Pay: 12500
```

RESULT:

Thus a shell script to find net and gross pay was executed successful.

3.4 REVERSE THE STRING AND TO CALCULATE THE LENGTH OF THE STRING.

AIM:

To write the shell script to find the length of the string and reverse the string.

ALGORITHM:

- Step1: Start the program.
- Step2: Enter the string.
- Step3: Calculate the length of the string using “wc” command.
- Step4: Using while loop, check whether the length of string is greater than zero.
- Step5: Using the test command, cut each other character and store it in a temporary variable.
- Step6: Stop the program.

PROGRAM:

```
echo " enter the string " read str
len=`echo $str | wc -c` len=`expr $len - 1`
echo " the length of the string is $len " while [ $len -gt 0 ]
do temp=`echo $str | cut -c
$len` rev=`echo $rev$temp` len=`expr $len - 1`
done
echo " the reversed sytring is $rev "
```

enter the string welcome the length of the string is 7
the reversed sytring is emoclew

OUTPUT:

```
Enter the string:
welcome
The length of the string is 7
The reversed string is emoclew
```

RESULT:

Thus the shell script to find the length of the string and reverse the string was executed successfully.

3.5 FIND GIVEN STRING IS PALINDROME OR NOT

AIM:

To write the shell script to find given string is palindrome or not.

ALGORITHM:

Step1: Start the program.
Step2: Enter the string.
Step3: Calculate the length of the string using “wc” command.
Step4: Using while loop, check whether the length of string is greater than zero.
Step5: Using the test command, cut each other character and store it in a temporary variable.
Step6: Check reverse of the string and entered string are equal,if it is equal print string is
 palindrome else print string is not a palindrome.
Step7: Stop the program.

PROGRAM:

```
echo "Enter the string: "  
read str  
len=$(echo -n "$str" | wc -c)  
rev=""  
while [ $len -gt 0 ]; do  
    temp=$(echo "$str" | cut -c $len)  
    rev="$rev$temp"  
    len=$((len - 1))  
done  
  
echo "The reversed string is $rev"  
if [ "$rev" = "$str" ]; then  
    echo "$str is a palindrome"  
else  
    echo "$str is not a palindrome"  
fi
```

OUTPUT:

```
Enter the string: madam  
The length of the string is 5  
The reversed string is madam  
madam is a palindrome
```

RESULT:

Thus, the shell script to determine if the given string is a palindrome was executed successfully.