ASSIGNMENT - 3.

* AIM:

Implement shell script to perform various string operations.

* HARDWARE /SOFTWARE

Desktop / Laptop, Linux OS, Processor, RAM, Storage, Terminal emulator.

* Problem Statements:

a) Compare 2 strings.
b) Concatenate 2 strings.
c) Find the length of a string.
d) Print odd-position characters.
e) Find whether a given string is a palindrome or not.
f) Print reverse of the string.is given.
g). Find the occurence of a character is a string.

* THEORY:

a) Compare 2 strings: This experiment involves checking if the 2 strings entered by user are same or not. It can be done by using a conditional statement to compare each character of the string. It uses the "==" operator which is used to compare the string characters.

Ways of doing it :
Firstly, it takes 2 strings. Secondly, the "==" operator
is used to compare each character. Lastly, it prints
whether the strings are same or not by using an
if - else statement.

b) Concate 2 strings : Concatenating means combining
2 strings into 1. For eg, echo Aryan & echo Mulik
are concatenated into AryanMulik. The shell script prompts
the user to enter two strings and then it joins them
by using "$var1 $var2" and it becomes a new string.

c) Length of string : The length of the string is the total
count of characters present in the whole string.
It can be achieved in 2 ways :
① can be done using inbuilt f'n "~ expr length~".
② can be done by using ${#var} syntax.

d) Print odd position characters : It is used to print odd
position characters for e.g. 1,3,5,7,... etc.
This can be done using a for loop putting conditions in it.

e) Palindrome or not : It is used to check if the given word
is a palindrome or not.
Palindrome word is a word written same on both sides.
It is done by reversing a string and then using a
conditional statement on it.
Eg- Saras, Madam, etc.

f) Reverse of a string: It is used to reverse a string. Eg. Aryan written in reverse is nayrA. It is done by using a for loop and reversing it by doing i--.

g) Occurance of a character: This is used if we want to check if the character is present in the word or not.
Eg. "This is a word". "a" is occuring in the string

CONCLUSION

In this experiment, we learnt how to implement different string operation on the shell and get the output.

```bash
1 #compare two strings
2 echo "Enter the first word: "
3 read str1
4 echo "Enter the second word: "
5 read str2
6
7 if [ "$str1" == "$str2" ] ;
8 then
9         echo "Strings are equal"
10 else
11         echo "Strings are not equal"
12 fi
13
14 #concatenation
15 concat=$str1$str2
16 echo "The concatenated string is $concat "
17
18 #finding the length of a string
19 length=`expr length "$concat"`
20 echo "The Length of the string is $length"
21
22 #Printing the characters at odd positions
23 length1=`expr length "$str2"`
24 for((i=0;i<length;i+=2))
25 do
26         echo "${str2:i:1}"
27 done
28
29 # To Print reverse of the string
30 reverse=""
31 for((i=length-1;i>=0;i--))
32 do
33         reverse="$reverse${str2:i:1}"
34 done
35 echo "Reversed string is $reverse"
36
37 #Chech if the string is Palindrome or not
38 echo "Enter the word: "
39 read str
40 reverse=""
41 for((i=length-1;i>=0;i--))
42 do
43         reverse="$reverse${str2:i:1}"
44 done
45 if [ "$str" == "$reverse" ] ;
46 then
47         echo "It is a palindrome word"
48 else
49         echo "It is not a palindrome word"
50 fi
51
52 #Occurence of the character in the string
53 echo "Enter the string:"
54 read st
55 echo "Enter the char you want to search for: "
56 read char
57 #use grep to find the occurence
58 count=`echo $st| grep -o "$char" | wc -l`
59 echo "The character '$char' appears $count times int the string"
60
```

```
aryan@aryan-VirtualBox:~$ bash stringopr.sh
Enter the first word:
aryan
Enter the second word:
aryan
Strings are equal
The concatenated string is aryanaryan
The Length of the string is 10
a
y
n


Reversed string is nayra
Enter the word:
saras
It is not a palindrome word
Enter the string:
hello
Enter the char you want to search for:
l
The character 'l' appears 2 times int the string
```