Expt.no.:6

Date:01/02/23 **PROGRAMS ON STRINGS**

1. CHECKING PALINDROMIC STRING

ALGORITHM:

Step 1: Start

Step 2: Define a function for palindrome(n) and proceed to 2.1

2.1: set rev=n[::-1]

2.2: if n is equal to rev proceed to 2.2.1, else go to step 2.3

2.2.1: print “palindrome”

2.3:print ”not palindrome”

Step 3: Read name from the user

Step 4: Call function from step 2 palindrome(name)

Step 5: Stop

PROGRAM:

```
def palindrome(n):
    rev = n[::-1]
    if(n==rev):
        print("Palindrome")
    else:
        print("Not palindrome")
name=input("Enter a name: ")
palindrome(name)
```

OUTPUT:

```
Enter a name: Malayalam
Palindrome
```

2.REVERSING A STRING

ALGORITHM:

Step 1: Start

Step 2: Define a function for reverse(n) and proceed to 2.1

2.1: set rev=n[::-1]

2.2: print rev

Step 3: Read name from the user

Step 4: Call function from step 2 reverse(name)

Step 5: Stop

PROGRAM:

```
def reverse(n):
    rev = n[::-1]
    print(rev)
name=input("Enter a name: ")
reverse(name)
```

OUTPUT:

```
Enter a name: python
nohtyp
```

3.COUNT THE LENGTH OF THE STRING

ALGORITHM:

Step 1: Start
Step 2: Initialize count=0
Step 3: Read str1(string) from the user
Step 4: for i in str1 go to step 4.1 else go to Step 5
    4.1: count=count+1
Step 5: Print(count)
Step 6: Stop

PROGRAM:

```
Str1=input("Enter the string:")
Count=0
for i in Str1:
    count+=1
print(count)
```

OUTPUT:

```
Enter the String: Interpreter
11
```

4.REPLACING VOWELS WITH @ SYMBOL

ALGORITHM:

Step 1: Start
Step 2: Read str1 from the user
Step 3: create a list for vowels
Step 4: for i in str1 go to step 4.1
    4.1: if i is in vowels go to step 4.1.1 else go to 4.2
        4.1.1: i=@
        4.1.2: print i
    4.2: print i
Step 5: Stop

PROGRAM:

```
str1=input("Enter the String:")
vow=['a','e','i','o','u','A','E','I','O','U']
for i in str1:
    if(i in vow):
        i="@"
        print(i,end=" ")
    else:
        print(i)
```

OUTPUT:

```
Enter the String: vowel
v@w@l
```

PRACTICE PROGRAM

PROGRAM:

```
A="Python"
print(len(A))
b=A.lower()
print(b)
B="Interpreter"
c=A+" "+B
print(c)
f=c.split( )
print(f)
j=A[0]
n=j.isupper()
print(n)
```

OUTPUT:

```
6
python
Python Interpreter
['Python', 'Interpreter']
True
```