**ASSIGNMENT 1**

**AIM:**

Perform following string operations using pointers:

1. Length

2. Copy

3. Concat

4. Compare

5. Reverse.

**OBJECTIVE:**

The objective of this program is to learn how the pointers work and implement that on these functions, although these functions are predefined in the string library.

**THEORY:**

**Pointer**: In computer science, a pointer is a programming language object that stores the memory address of another value located in computer memory. A pointer references a location in memory, and obtaining the value stored at that location is known as dereferencing the pointer.

**String Functions:** Strings are defined as an array of characters. The difference between a character array anda string is the string is terminated with a special character '\0'. It will append copy of the

source string in the destination string. The C programming language has a set of functions implementing operations On strings (character strings and byte strings) in its standard library. Various operations, such as Copying, concatenation searching, etc. are supported.

The string operations to be performed are:

1) Length: The string length method returns the number of characters written in the String.

Eg: hello, the length is 5.

2) Copy: it is used to copy one string to another.

3) Concat: string concatenation is the operation of joining character strings end-to-end.

For example, the concatenation of "snow" and "ball" is "snowball".

4) Compare: It compares the value of the string object (or a substring) to the sequence of

characters.

5) Reverse: it reverses the string character by character.

Eg: Boy will be Yob

**CODE**:

#include<iostream>

using namespace std;

char str1[20],str2[20],\*p1,\*p2;

int l1=0,l2=0,i,cnt=0,len=0;

void stringcpy()

{

cout<<"\n Enter string:";

cin>>str1;

p1=str1;

p2=str2;

while(\*p1!='\0')

{

\*p2=\*p1;

p1++;

p2++;

}

\*p2='\0';

cout<<"\n Original String="<<str1<<endl;

cout<<"\n Copied String="<<str2<<endl;

}

void stringcompare()

{

cout<<"\n Enter string1:";

cin>>str1;

cout<<"\n Enter String2:";

cin>>str2;

p1=str1;

p2=str2;

while(\*p1!='\0')

{

l1++;

p1++;

}

while(\*p2!='\0')

{

l2++;

p2++;

}

if(l1!=l2)

{

cout<<"Strings are not equal"<<endl;

}

else

{

for(i=0;i<l2;i++)

{

if(\*p1==\*p2)

cnt++;

}

}

if(cnt==l1)

cout<<"Strings are equal"<<endl;

}

void stringlength()

{

cout<<"\n Enter String:";

cin>>str1;

p1=str1;

while(\*p1!='\0')

{

len++;

p1++;

}

cout<<"Length of string:"<<len<<endl;

}

void stringconcate()

{

cout<<"\n Enter string1:";

cin>>str1;

cout<<"\n Enter String2:";

cin>>str2;

p1=str1;

p2=str2;

while(\*p1!='\0')

{

p1++;

}

while(\*p2!='\0')

{

\*p1=\*p2;

p1++;

p2++;

}

\*p1='\0';

cout<<"\n Concatenated string: "<<str1<<endl;

}

void stringrev()

{

int len=0;

cout<<"\n Enter string:";

cin>>str1;

p1=str1;

for (int i=0;str1[i]!='\0';i++)

{

len++;

p1++;

}

p2=&str2[0];

for(int i=0;i<=len;i++)

{

\*p2=\*p1;

// cout<<\*r;

p2++;

p1--;

}

\*p2='\0';

p2=str2;

cout<<"The reverse string is: ";

for(int i=0;i<=len;i++)

{

cout <<\*p2;

p2++;

}

}

int main()

{

int ch;

do

{

cout<<"\n 1.Copy \n 2.Compare \n 3.Length \n 4.Concatenate \n 5.Reverse\n 6.Exit";

cout<<"\n";

cout<<"\n Enter choice:";

cin>>ch;

switch(ch)

{

case 1:

stringcpy();

break;

case 2:

stringcompare();

break;

case 3:

stringlength();

break;

case 4:

stringconcate();

break;

case 5:

stringrev();

break;

case 6:

return 0;

break;

default:

cout<<"\n Enter Proper choice";

break;

}

}while(ch<=6);

return 0;

}

**COMPLEXITY:**

stringcompare: O(n)

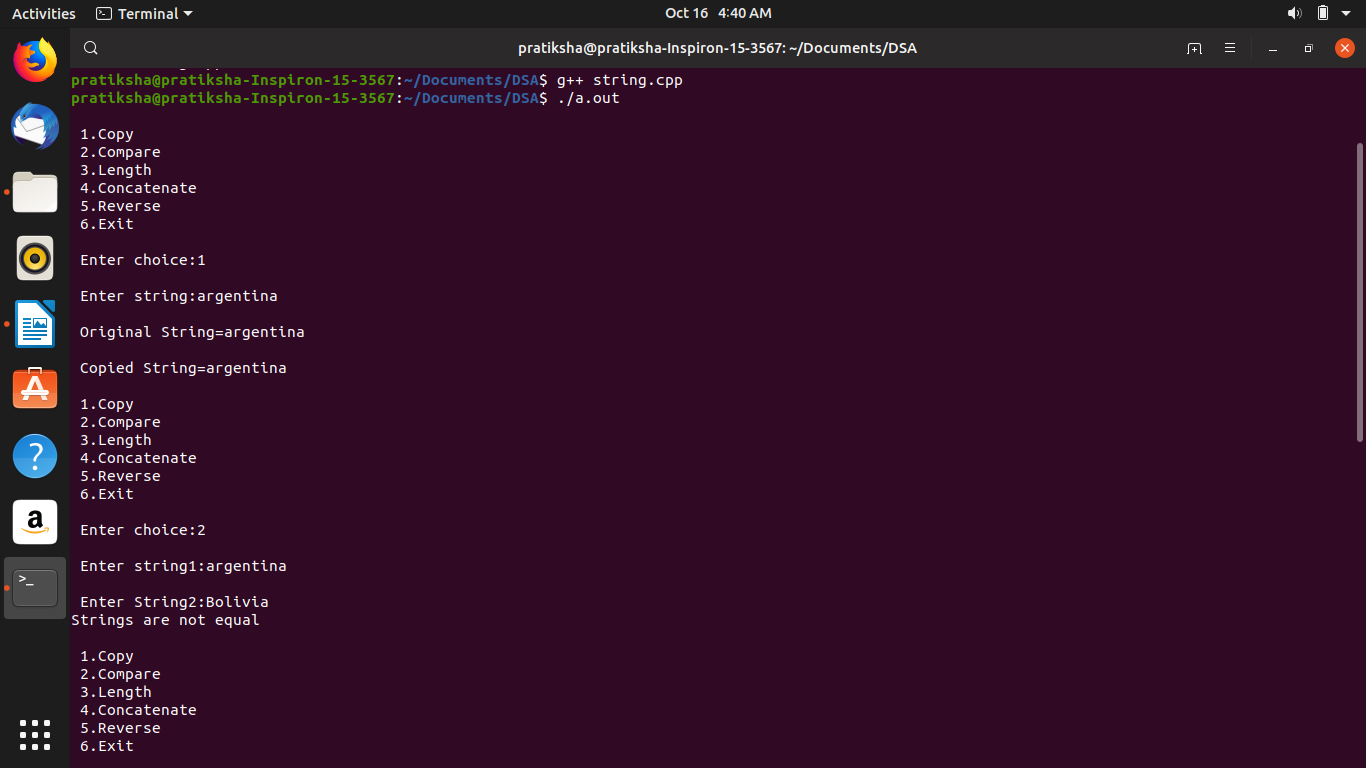
stringconcate: O(n)

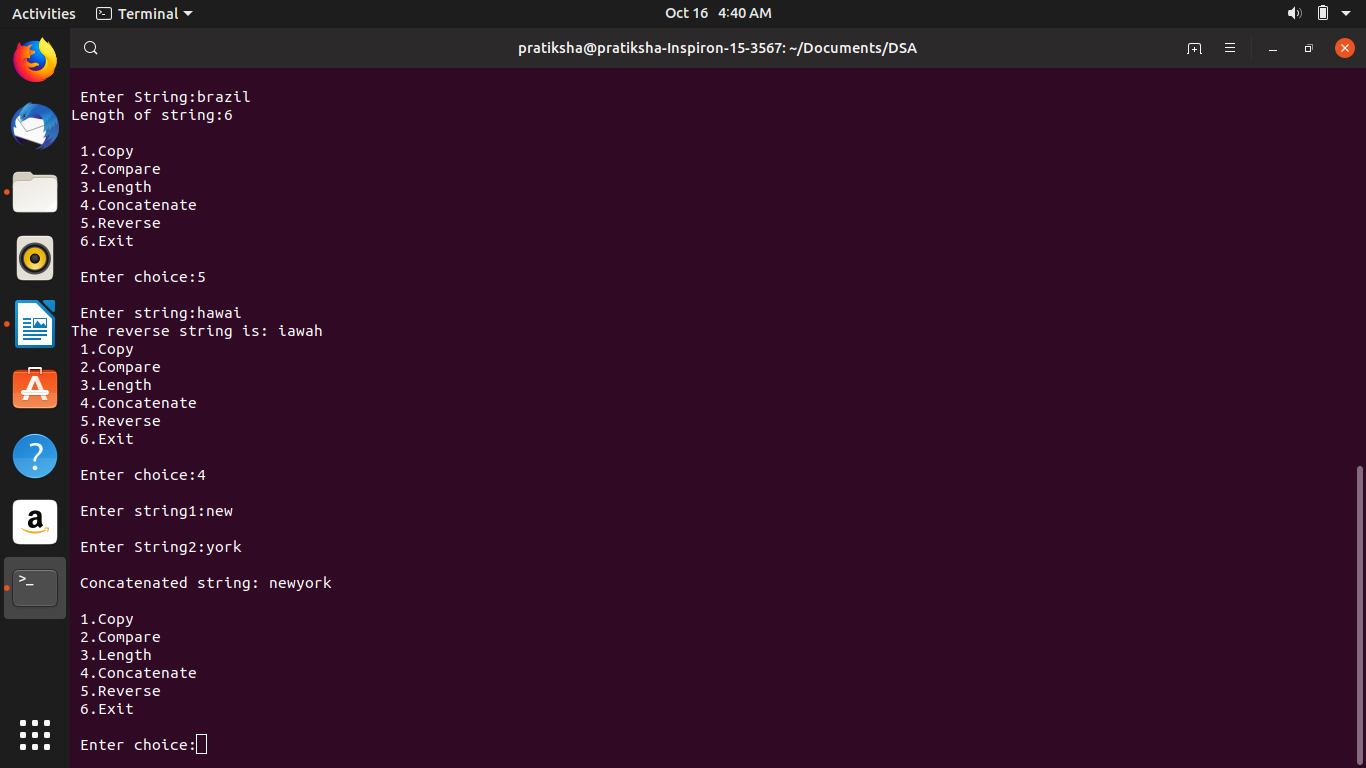
stringlength: O(n)

stringrev: O(n)

stringcpy: O(n)

**OUTPUT:**





**CONCLUSION:**

In this program we learnt the pointers concept and how to use them in the string

Functions.

Last modified: 7:22 am