**DSA LAB – 3**

**Name:** Etcherla Sai Manoj **Mis. No:** 112015044 **Branch:** CSE

**1)Question: (with Library functions)**

**Code:**

#include<iostream>

#include<cstring>

using namespace std;

void stringLen(char a[]){

cout << "The length of first string : " << strlen(a);

}

void stringCopy(char a[], char b[]){

strcpy(b,a);

cout << "String 1 : " << a << endl;

cout << "String 2 : " << b << endl;

cout << "First string is copied into Second string";

}

void stringConcatenation(char a[], char b[]){

strcat(a,b);

cout << a;

}

void stringReverse(char a[]){

cout << "The reverse of String : " << strrev(a) << endl;

}

void stringPalindrome(char a[]){

char b[50];

strcpy(b,a);

strrev(b);

int check = strcmp(a,b);

if(check == 0) cout << "The string entered is a Palindrome.";

else cout << "The string is not a Palindrome.";

}

void stringCompare(char a[], char b[]){

int check = strcmp(a,b);

if(check == 0) cout << "Strings are equal";

if(check < 0) cout << "String 1 is greater than String 2";

if(check > 0) cout << "String 1 is less than String 2";

}

int Substring(char a[], char b[]){

int count = 0,sub\_len = strlen(b);

char \*mainstr = a;

char \*substr = b;

if(sub\_len){

while(mainstr = strstr(mainstr,substr)){

mainstr = mainstr + sub\_len;

count++;

}

}

//cout << count;

return count;

}

int main(){

cout << "\n\*\*\*MENU FOR STRING OPERATIONS\*\*\*\n" << endl;

cout << "1. String Length\n2. String Copy\n3. String Concatenation\n4. String Reverse\n";

cout << "5. String Palindrome\n6. String Compare\n7. Substring\n" << endl;

cout << "Select which operation you want to perform : ";

int choice;

cin >> choice;

char str1[50], str2[50];

cout << "\nEnter a string : ";

cin >> str1;

switch (choice)

{

case 1:

stringLen(str1);

break;

case 2:

stringCopy(str1,str2);

break;

case 3:

cout << "Enter another string : ";cin >> str2;

stringConcatenation(str1, str2);

break;

case 4:

stringReverse(str1);

break;

case 5:

stringPalindrome(str1);

break;

case 6:

cout << "Enter another string : ";cin >> str2;

stringCompare(str1, str2);

break;

case 7:

cout << "Enter sub string : ";cin >> str2;

cout << "The substring occurred " << Substring(str1, str2);

cout << " times in main string" << endl;

break;

default:

cout << "Enter a valid choice" << endl;

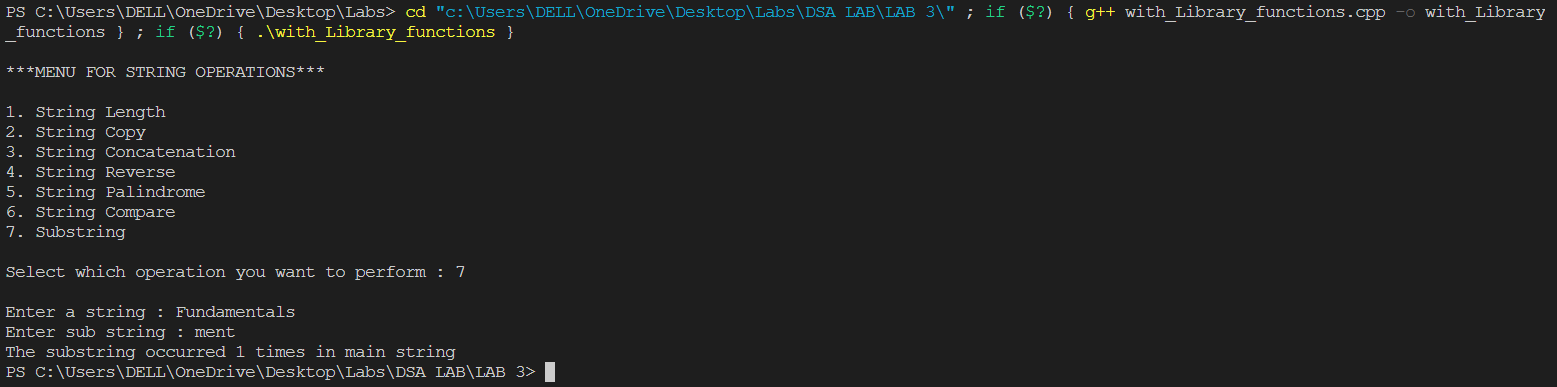
break;

}

return 0;

}

**Input & Output:**

****

**2)Question: (without Library functions)**

**Code:**

#include<iostream>

using namespace std;

int stringLen(char a[]){

int index = 0, len = 0;

while(a[index] != '\0'){

len = len +1; //increment length count

index = index + 1;

}

return len;

}

char \*stringCopy(char a[], char b[]){

int index = 0;

while(a[index] != '\0'){

b[index] = a[index]; //appends one String to other character wise

index = index + 1;

}

b[index] = '\0';

return b;

}

char \*stringConcatenation(char a[], char b[]){

int start = 0, end = 0;

while(a[start] != '\0'){

start = start+ 1; //index count to end of string

}

while(b[end] != '\0'){

a[start] = b[end]; //adds new string to the end of first string

start =start +1;

end = end + 1;

}

a[start] = '\0';

return a;

}

char \*stringReverse(char a[]){

int start = 0, end = 0;

while(a[end] != '\0'){

end = end + 1; //calculates length of string i.e. end value

}

end = end - 1;

while(start < end){

swap(a[start], a[end]); //swap alternate characters from start and end

start = start + 1;

end = end - 1;

}

return a;

}

bool stringPalindrome(char a[]){

int start = 0;

while(a[start] != '\0'){

start = start + 1; //claculates length of string

}

int end = start - 1; start = 0;

while(start < end){

if(a[start] == a[end]){ //checks alternate terms from start and end

start = start + 1;

end = end - 1;

}

else{

goto jump;

}

}

jump:

if(start < end){

return 0; //returns false(0) if not a palindrome

}

else{

return 1; //returns true(1) if palindrome

}

}

int stringCompare(char a[], char b[]){

int index = 0;

while(a[index] != '\0' || b[index] != '\0'){

if(a[index] == b[index]){ //compares each charater of both string

index = index + 1;

}

else{

break;

}

}

int diff = a[index] - b[index];

return diff;

}

int Substring(char a[], char b[]){

int len\_str = 0, len\_sub = 0;

while(a[len\_str] != '\0'){

len\_str++; //calculate length of string

}

while(b[len\_sub] != '\0'){

len\_sub++; //calculate length of substring

}

if(len\_sub > len\_str) return 0; //not a substring (0 occurences)

int count = 0;

for(int i = 0; i <= len\_str - len\_sub; i++){

int j;

for(j = 0; j <= len\_sub - 1; j++){

if(a[i+j] != b[j]){

break;

}

}

if(j == len\_sub){

count++; //increment substring count

}

}

return count;

}

int main(){

cout << "\n\*\*\*MENU FOR STRING OPERATIONS\*\*\*\n" << endl;

cout << "1. String Length\n2. String Copy\n3. String Concatenation\n4. String Reverse\n";

cout << "5. String Palindrome\n6. String Compare\n7. Substring\n" << endl;

cout << "Select which operation you want to perform : ";

int choice;

cin >> choice;

char str1[50], str2[50];

cout << "\nEnter a string : ";

cin >> str1;

switch (choice)

{

case 1:

cout << "The length of String : " << stringLen(str1); //Calculates string length

break;

case 2:

cout << "Copied String(str2) : " <<stringCopy(str1,str2); //Copies String to another variable

break;

case 3:

cout << "Enter another string : ";cin >> str2;

cout << "Concatenated string : " << stringConcatenation(str1, str2); //Concntenates two strings

break;

case 4:

cout << "The reverse of string : " << stringReverse(str1); //Reverses the String

break;

case 5:

if(stringPalindrome(str1)) cout << "The string entered is a Palindrome."; //Checks a string is palindrome or not

else cout << "The string entered is not a Palindrome.";

break;

case 6:

cout << "Enter another string : ";cin >> str2;

cout << "The differnece of strings is : " << stringCompare(str1, str2) << endl; //Compares two strings

if(stringCompare(str1,str2) == 0) cout << "Strings are equal";

if(stringCompare(str1,str2) < 0) cout << "String 1 is greater than String 2";

if(stringCompare(str1,str2) > 0) cout << "String 1 is less than String 2";

break;

case 7:

cout << "Enter sub string : ";cin >> str2;

cout << "The substring occurred " <<Substring(str1, str2); //Counts occurence of Substring

cout << " times in main string";

break;

default:

cout << "Enter a valid choice";

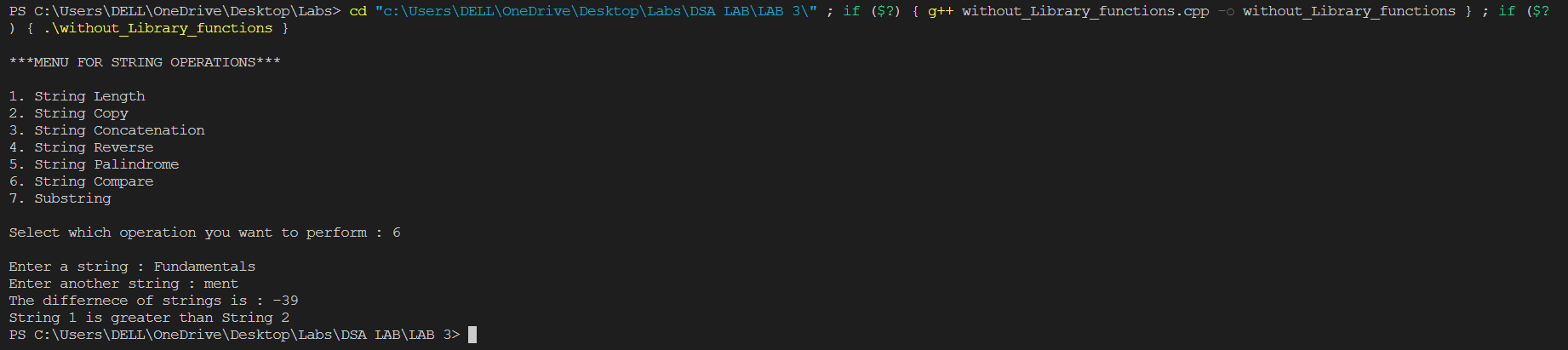
break;

}

return 0;

}

**Input & Output:**

****