\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Computer Programming Lab

CEN-392

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Program 6

Code :-

#include <iostream>

using namespace std;

int strlen(char str[])

{

    int i = 0;

    while (str[i] != '\0')

        i++;

    return i;

}

void strlength()

{

    cout<<"\nString Length Operation Is Selected.\n";

    char str[100];

    fflush(stdin);

    cout << "Enter The String : ";

    cin.getline(str, 100);

    int slen = strlen(str);

    cout << "\nString Length : " << slen << "\n";

}

void strrev()

{

    cout<<"\nString Reverse Operation Is Selected.\n";

    char str[100];

    fflush(stdin);

    cout << "Enter The String : ";

    cin.getline(str, 100);

    int slen = strlen(str);

    for (int i = 0; i < slen / 2; i++)

    {

        char ch = str[i];

        str[i] = str[slen - i - 1];

        str[slen - i - 1] = ch;

    }

    cout << "\nReversed String : " << str << "\n";

}

void strcpy()

{

    cout<<"\nString Copy Operation Is Selected.\n";

    char str1[100], str2[100];

    fflush(stdin);

    cout << "Enter The String : ";

    cin.getline(str2, 100);

    int s2len = strlen(str2);

    for (int i = 0; i < s2len; i++)

    {

        str1[i] = str2[i];

    }

    str1[s2len] = '\0';

    cout << "\nString Is Copied : " << str1 << "\n";

}

void strcmp()

{

    cout<<"\nString Compare Operation Is Selected.\n";

    char str1[100], str2[100];

    fflush(stdin);

    cout << "Enter The String\_1 : ";

    cin.getline(str1, 100);

    cout << "Enter The String\_2 : ";

    cin.getline(str2, 100);

    int s1len = strlen(str1);

    int s2len = strlen(str2);

    if (s1len != s2len)

    {

        cout << "\n'"<<str1<<"' And '"<<str2<<"' Are Not Same\n";

        return;

    }

    for (int i = 0; i < s1len; i++)

    {

        if (str1[i] != str2[i])

        {

            cout << "\n'"<<str1<<"' And '"<<str2<<"' Are Not Same\n";

            return;

        }

    }

    cout <<"\n '" <<str1<<"' And '"<<str2<<"' Are Same\n";

}

void strcat()

{

    cout<<"\nString Concatation Operation Is Selected.\n";

    char str1[100], str2[100];

    fflush(stdin);

    cout << "Enter The String\_1 : ";

    cin.getline(str1, 100);

    cout << "Enter The String\_2 : ";

    cin.getline(str2, 100);

    int s1len = strlen(str1);

    int s2len = strlen(str2);

    for (int i = 0; i < s2len; i++)

    {

        str1[i + s1len] = str2[i];

    }

    str1[s1len + s2len] = '\0';

    cout << "\nConcated String : " << str1 << "\n";

}

void isPalindrome()

{

    cout<<"\nString Pallindrome Operation Is Selected.\n";

    char str[100];

    fflush(stdin);

    cout << "Enter The String : ";

    cin.getline(str, 100);

    int slen = strlen(str);

    for (int i = 0; i < slen / 2; i++)

    {

        if (str[i] != str[slen - i - 1])

        {

            cout <<str<<" Is Not A Pallindrome\n";

            return;

        }

    }

    cout <<"\n"<<str<<" Is Pallindrome\n";

}

void Seach()

{

    cout<<"\nString Search Substring Operation Is Selected.\n";

    char str1[100], str2[100];

    fflush(stdin);

    cout << "Enter The String\_1 : ";

    cin.getline(str1, 100);

    cout << "Enter The String\_2 : ";

    cin.getline(str2, 100);

    int s1len = strlen(str1);

    int s2len = strlen(str2);

    if (s1len < s2len)

    {

        cout << "Substring Not Found\n";

        return;

    }

    bool check=false;

    for (int i = 0; i < s1len; i++)

    {

        int j = 0;

        for (; j < s2len && i + j < s1len; j++)

        {

            if (str1[i + j] != str2[j])

            {

                break;

            }

        }

        if (j == s2len)

        {

            if(!check)cout<<"\nSubstring Found!\n";

            cout << "Index : " << i << "\n";

            check=true;

        }

    }

    if(!check)

    cout << "\nSubstring Not Found!\n";

}

void Menu()

{

    cout << "\n\n\_\_\_\_String\_Operations\_\_\_\_\n";

    cout << "1.Length\n";

    cout << "2.Reverse\n";

    cout << "3.Copy\n";

    cout << "4.Compare\n";

    cout << "5.Concatnate\n";

    cout << "6.Pallindrome\n";

    cout << "7.Search Substring\n";

    cout << "8.Exit\n";

    cout << "Enter Your Choice : ";

}

void AnsBar()

{

    cout<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n";

}

bool Options()

{

    int opt;

    fflush(stdin);

    cin >> opt;

    AnsBar();

    switch (opt)

    {

    case 1:

        strlength();

        break;

    case 2:

        strrev();

        break;

    case 3:

        strcpy();

        break;

    case 4:

        strcmp();

        break;

    case 5:

        strcat();

        break;

    case 6:

        isPalindrome();

        break;

    case 7:

        Seach();

        break;

    case 8:

    cout<<"Exit Is Selected.\n";

    AnsBar();

        return 0;

    default:

        cout << "Invalid Entry!\n";

        break;

    }

    AnsBar();

    return 1;

}

int main()

{

    system("cls");

    cout << "\_\_\_\_Vicky\_Gupta\_20BCS070\_\_\_\_\n";

    while (true)

    {

        Menu();

        if (!Options())

            break;

    }

    cout << "Exiting...\n";

    return 0;

}

Output :-

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generated