**String Programs -**

1 - Taking Input in string, Sort & Reverse, Palindrome of string.

2 - WAP for Count words in string -

2.1 - Program for count Vowels & Consonants in String-

3 - For the 2 given strings check that are they both Anagram of each other. print yes or no only

4 - Program for Sum of numbers inside the string.

5 - make first letter capital of given string or user taken string, but if already capital then no need to change anything -

6 - make first letter Small of given string or user taken string, but if already Small then no need to change anything -

7 - For the entire string convert Upper string to Lower string

8 - change all the cases of string.

9 - Remove Vowels, Consonants from the given string.

10 - Distinguish Vowels and Consonants in a given string -

11 - common letters sequences in a string -

12 - For given two strings check that are rotation of each other or not?

13 - Enter the First Non-Repeating Character from the given string -

14 - Find the First Repeating element of the string -

**Strings in CPP** –

#include <bits/stdc++.h>

// #include <iostream>

// #include<algorithm>

// #include<climits>

// #include<string>

// #include<cctype>

using namespace std;

int main()

{

**// Character Array or Strings -**

    char arr[7] = {'S', 'H', 'U', 'b', 'H', 'A', 'm'};

    for (int i = 0; i < 7; i++)

    {

        cout << arr[i] << " ";

    }

    // Note - Every character array or string contains a null character

    // Character array or string can print all elements at a time no need to print one by one as int array

    cout << "i'm in Microsoft " << endl;

    cout << arr[7];

    cout << " 51LPA Package" << endl;

    // i'm in Microsoft Ç 51LPA Package ; -  b/w both the strings, here ia a null character

    // It is basiocally '\0'

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

    /\*

**String Basic Syntax - data\_type variable name : - string s1**

    In string value is given by double quotes, while for the single variable it is given by single quotyes

    String is a contiguos char array.

    In string no need to give size manualy as char array. It is dynamic and allot the size automatically

    \*/

**// Taking input in string -** If there is no space then use cin>> for input but for the sentence or line with multiple words by using getline funciton

    // string s1 = "";

    // cout<<"Now give the string value - "<<endl;

    // cin>>s1;// he, buddy what's1 app, i got placed in Microsoft

    // cout<<s1<<endl;//he,

    // getline(cin,s1);

    // cout<<s1<<endl;//he, buddy what's1 app, i got placed in Microsoft

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Now for printing the string using for loop concept  -** as we know evry string having null character at last

    // string s1;

    // getline(cin,s1);//Congratulations, Shubham..!! for getting placed in Microsoft

    // cout<<s1<<endl;//Congratulations, Shubham..!! for getting placed in Microsoft

    // for (int i = 0; s1[i] == '\0' ; i++)

    // {

    // cout<<s1[i]<<" ";

    // }

    // // C o n g r a t u l a t i o n s1 ,   S h u b h a m . . ! !   f o r   g e t t i n g   p l a c e d   i n   M i c r o s1 o f t

    // // For printing how many characters total in the string using for loop.

    // for (int i = 0; s1[i] == '\0' ; i++)

    // {

    // cout<<i;

    // }

    // 01234567891011121314151617181920212223242526272829303132333435363738394041424344454647484950515253545556575859

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// check that in entered string there is space or not ?, so using for loop**

    // string s2;

    // cout<<"what is your statement dude ..!!"<<endl;//Microsft Hyderabad,MicrosoftHyderabad

    // getline(cin, s2);

    // cout<<"So you informed about - "<<s2<<endl;//Microsft Hyderabad

    // cout<<s2.size();// 19

    // for (int i = 0; s2[i] == '\0'; i++)

    // {

    //     if (s2[i] == ' ')

    //     {

    //         cout<<"Yes Space Exist";

    //         return 0;

    //     }

    // }

    // cout<<"No space buddy";

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Sort & reverse for the string array -**

    // string s1;

    // cout<<"What's1 in your mind"<<endl;

    // cin>>s1;//MicrosoftBengaluru

**// // sort function for string -**

    // sort(s1.begin(), s1.end());

    // cout<<s1<<endl;//BMacefgilnoorrstuu

**// // reverse function for string -**

    // reverse(s1.begin(), s1.end());

    // cout<<s1<<endl;//urulagneBtfosorciM

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Palindrome string -**

    // string s1;

    // cout<<"What's1 in your mind"<<endl;

    // getline(cin,s1);

    // string original = s1;

    // reverse(s1.begin(), s1.end());

    // if (original == s1)

    // {

    //     cout<<"Yes, Palindrome";

    // }

    // else

    // {

    //     cout<<"Not a palindrome string";

    // }

    /\*

    naman

    Yes, Palindrome

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun1 -  Count words in string -**

    // string s1;

    // cout<<"What's1 in your mind"<<endl;

    // getline(cin,s1);

    // int space=0;

    // for (int i = 0; s1[i] == '\0'; i++) // OR for(int i=0; i<s1.size(); i++)

    // {

    //     if (s1[i] == ' ')

    //     {

    //         space++;

    //     }

    // }

    // cout<<"the no. of spaces here is - "<<space<<endl;

    // int words = space+1;

    // cout<<"So total words is -  "<<words<<endl;

    /\*

    What's1 in your mind

    Thank You, I accept the offer from Microsoft for hyderabad Location.

    the no. of spaces here is - 10

    So total words is -  11

    \*/

    // using functions -

    // #include <bits/stdc++.h>

    // using namespace std;

    // int count\_words(string s1)

    // {

    //     int space = 0;

    //     for (int i = 0; i < s1.size(); i++) // OR for(int i=0; s1[i] == '\0'; i++)

    //     {

    //         if (s1[i] == ' ')

    //         {

    //             space++;

    //         }

    //     }

    //     return space + 1;

    // }

    // int main()

    // {

    //     string s1;

    //     cout << "What's1 in your mind" << endl;

    //     getline(cin, s1);

    //     cout << count\_words(s1);

    // }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 2 -  Count words in string -**

**// For Vowels using function -**

    // #include <bits/stdc++.h>

    // using namespace std;

    // int count\_vowels(string s1)

    // {

    //     int count = 0;

    //     for (int i = 0; i < s1.size(); i++) // OR for(int i=0; s1[i] == '\0'; i++)

    //     {

    //         if (s1[i] == 'a' ||s1[i] == 'e' ||s1[i] == 'i' ||s1[i] == 'o' ||s1[i] == 'u' || s1[i] == 'A' ||s1[i] == 'E' ||s1[i] == 'I' ||s1[i] == 'O' ||s1[i] == 'U')

    //         {

    //             count++;

    //         }

    //     }

    //     return count;

    // }

    // int main()

    // {

    //     string s1;

    //     cout << "What's1 in your mind" << endl;

    //     getline(cin, s1);

    //     cout << count\_vowels(s1);

    // }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 2.1 -  Count Consonants in string -**

**// For Consonants using function -**

    // #include <bits/stdc++.h>

    // using namespace std;

    // int count\_Consonants(string s1)

    // {

    //     int count = 0;

    //     for (int i = 0; i < s1.size(); i++) // OR for(int i=0; s1[i] == '\0'; i++)

    //     {

    //         if (!(s1[i] == 'a' ||s1[i] == 'e' ||s1[i] == 'i' ||s1[i] == 'o' ||s1[i] == 'u' || s1[i] == 'A' ||s1[i] == 'E' ||s1[i] == 'I' ||s1[i] == 'O' ||s1[i] == 'U'))

    //         {

    //             count++;

    //         }

    //     }

    //     return count;

    // }

    // int main()

    // {

    //     string s1;

    //     cout << "What's1 in your mind" << endl;//Microsoft Hyderabad

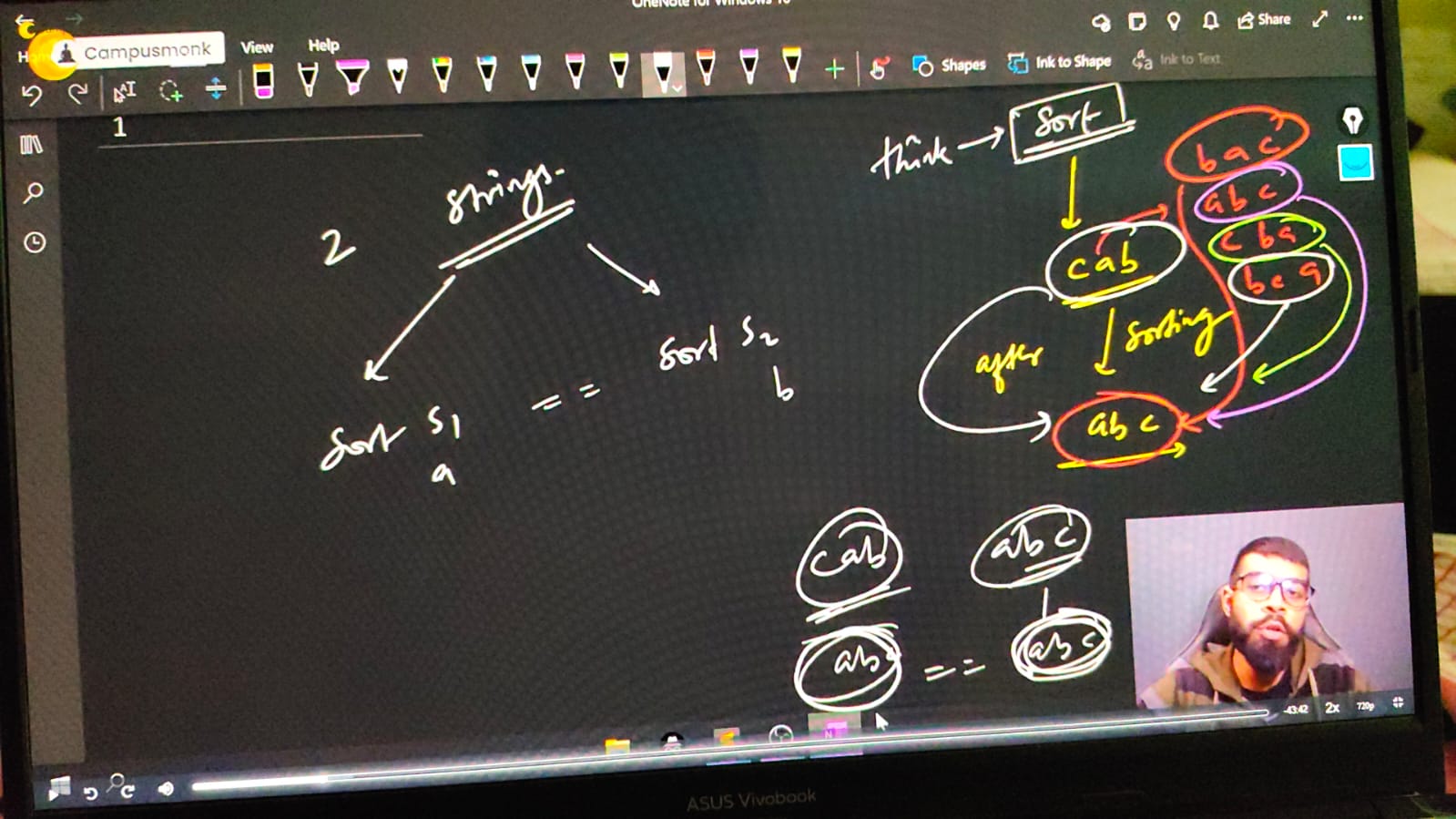
    //     getline(cin, s1);

    //     cout << count\_Consonants(s1);// 13

    // }

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// QUn 3 - For the 2 given strings check that are they both Anagram of each other. print yes or no only**



    /\*

    Anagram - ek string ke saare characters kisi bhi order me utne hi no. of times agr dusri string me aa rhe he, and no ay another character, then they are Anagram of each other.

    Ex. - S1 - abc, and S2 - cba or bac or bca any of these are Ahanagram of String -1

    \*/

    // string s1,s2;

    // cout << "What's1 in your mind" << endl;//Microsoft Hyderabad

    // cout<<"Enter the first string   - "<<" ";

    // getline(cin, s1);

    // cout<<"Now, Enter the Second string   - "<<" ";

    // getline(cin, s2);

    // sort(s1.begin(), s1.end());

    // sort(s2.begin(), s2.end());

    // cout<<"After sort string 1 is "<<s1<<" and string 2 is "<<s2<<endl;

    // if (s1==s2)

    // {

    //     cout<<"So clearly, strings are Anagram";

    // }

    // else

    // {

    //     cout<<"we can see, strings are Not an Anagram";

    // }

    /\*

    What's1 in your mind

    Enter the first string   -  shruti

    Now, Enter the Second string   -  rutish

    After sort string 1 is hirstu and string 2 is hirstu

    So clearly, strings are Anagram

    What's1 in your mind

    Enter the first string   -  Madhuri

    Now, Enter the Second string   -  ridhima

    After sort string 1 is Madhiru and string 2 is adhiimr

    we can see, strings are Not an Anagram

    What's1 in your mind

    Enter the first string   -  Muskan

    Now, Enter the Second string   -  upmuskan

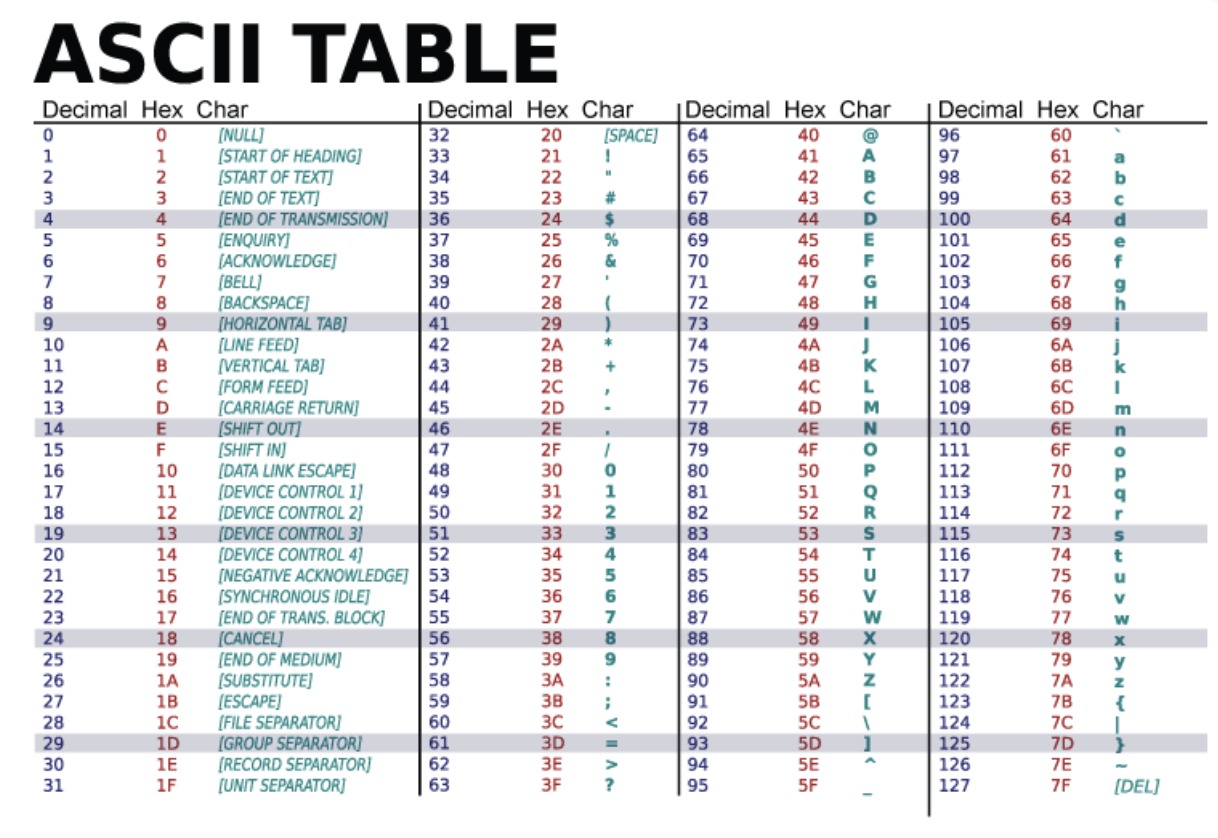
    After sort string 1 is Maknsu and string 2 is akmnpsuu

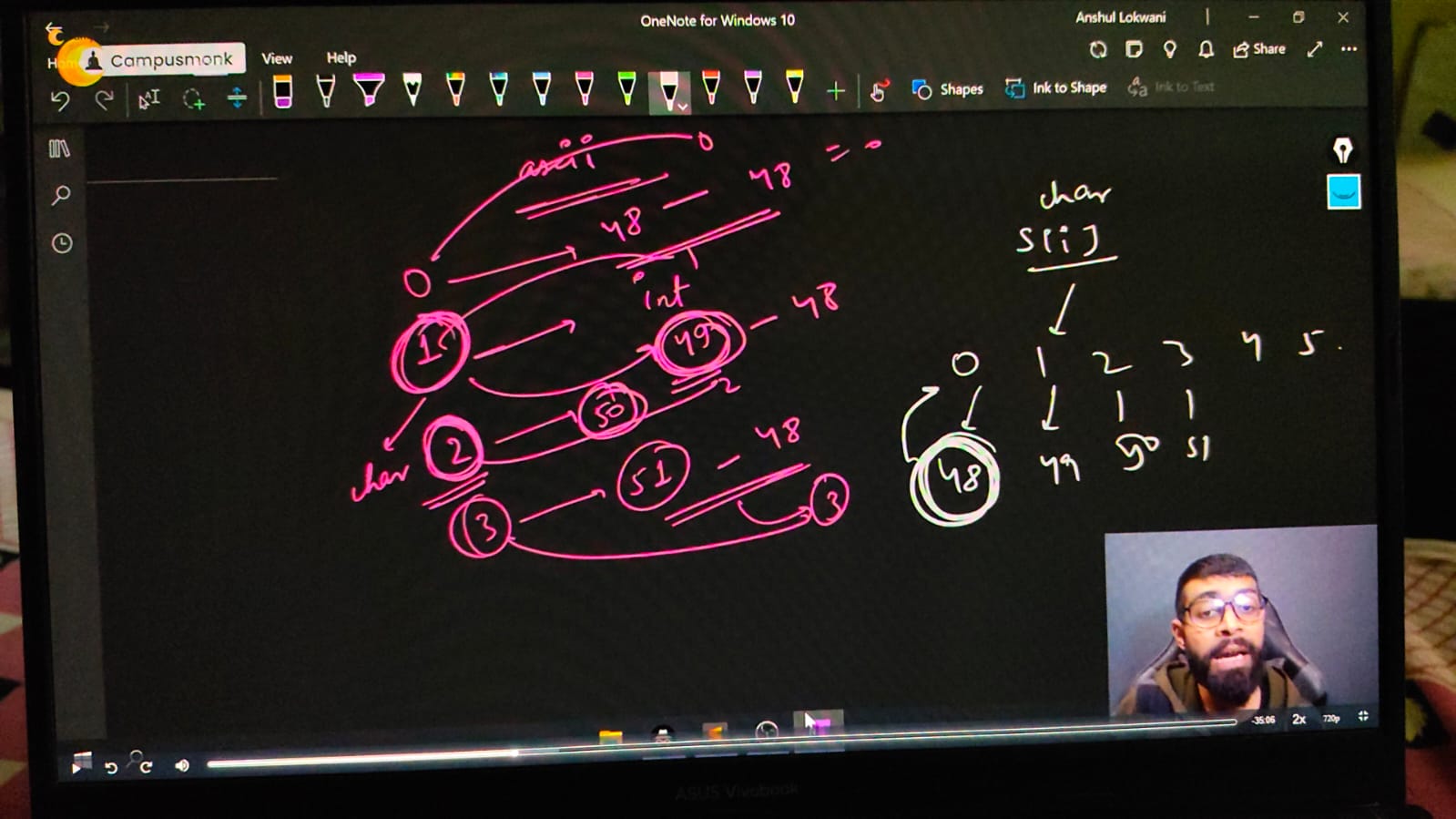
    we can see, strings are Not an Anagram

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 4 - Sum of numbers inside the string.**





    // string s1 = "1578";

    // // Sum of int in a string

    // int sum=0;

    // for (int i = 0; i <s1.size(); i++)

    // {

    //     // sum+=s1[i];

    //     sum+=s1[i]-'0'; // OR

    //     // sum+=s1[i]-48;

    // }

    // cout<<"So, the sum of int inside string is - "<<sum<<endl;

    // So, the sum of int inside string is - 213 - which is wrong because in string int deals with Ascii value as its a character in a string. So its giving the sum of ASCII values of ths entered integers

    // So, for overoming this int problem in string we need to substract 45 which is ASCII value of

    /\*

    So, the sum of int inside string is - 21

    \*/

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Capital & SMall characters OR Lower & Upper case in String -**

    // string a = "abc";

    // string b = "ABC";

    // if (a==b)

    // {

    //     cout<<"Yes"<<endl;

    // }

    // // So cleaarly it isn't printing anything as both are different because of Ascii values.

    // cout<<int('A')<<endl;//65

    // cout<<int('Z')<<endl;//90

    // cout<<int('a')<<endl;//97

    // cout<<int('z')<<endl;//122

    // cout<<int('0')<<endl;//48

    // cout<<int('9')<<endl;//57

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Lower & Upper case -**

**// Qun 5 -  make first letter capital of given string or user taken string, but if already capital then no need to change anything -**

    // string a = "shruti";

    // 📌 Logic - For making first letter capital of given small case string , simple substract the no. 32 which is ASCII Value difference of a-A : 97-65 = 32. and applicable for all the alphabets

    // a[0]-=32;

    // cout<<a<<endl;//Shruti.

    // string a1;

    // cout<<"What word you want to write and conver the first lower letter into upper letter"<<endl;

    // getline(cin, a1);

    // // cout<<"So the changes is here -  "<<a1<<endl;//So the changes is here -  Google

    // // but what if I alreaady write first letter capital in the string with this logic. then we get any another result, so the condition for avoiding that is -

    // if (a1[0]>='a'&& a1[0]<='z')

    // {

    // a1[0]-=32;

    // }

    // cout<<"So the changes is here -  "<<a1<<endl;

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 6 -  make first letter Small of given string or user taken string, but if already Small then no need to change anything -**

    // string a1;

    // cout<<"What word you want to write and conver the first lower letter into lowewr letter"<<endl;

    // getline(cin, a1);//Google

    // if (a1[0]>='A'&& a1[0]<='Z')

    // {

    //     a1[0]+=32;

    // }

    // cout<<"So the changes is here -  "<<a1<<endl;//So the changes is here -  google

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// // Qun - For the entirre string convert small string to capital string**

    //     string a1;

    //     cout<<"What's1 in your mind for upper letter converison"<<endl;

    //     getline(cin, a1);

    //     for (int i = 0; i < a1.size(); i++)//google

    //     {

    //         if (a1[i]>='a' && a1[i]<='z')

    //         {

    //             a1[i]-=32;

    //         }

    //     }

    //     cout<<"So, string we got in Upper Case is "<<a1<<endl;//GOOGLE

    /\*

    What word you want to write and conver the first lower letter into lowewr letter

    microsoft

    So, string we got in Upper Case is MICROSOFT

    What word you want to write and conver the first lower letter into lowewr letter

    MicroSoft

    So, string we got in Upper Case is MICROSOFT

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 7 - For the entirre string convert Upper string to Lower string**

    // string a1;

    // cout<<"What's1 in yor mind for lower letter strig - "<<endl;

    // getline(cin, a1);

    // for (int i = 0; i < a1.size(); i++)//google

    // {

    //     if (a1[i]>='A' && a1[i]<='Z')

    //     {

    //         a1[i]+=32;

    //     }

    // }

    // cout<<"So, string we got in Upper Case is "<<a1<<endl;//GOOGLE

    /\*

    What's1 in yor mind for lower letter strig -

    ATLASSIAN

    So, string we got in Upper Case is atlassian

    What's1 in yor mind for lower letter strig -

    AtlassIan

    So, string we got in Upper Case is atlassian

    \*/

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

    // Upper to Lower using shortcuts -

    // string s1 = "Microsoft";

    // // s1[0]= tolower(s1[0]);

    // cout<<s1<<endl; // microsoft

    // s1[3]= toupper(s1[3]);

    // cout<<s1<<endl;//MicRosoft

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 8 - change all the cases of string.**

    //     string s1 = "sHubhAm";

    //     for (int i = 0; i < s1.size(); i++)

    //     {

    //         if (s1[i]>='a' && s1[i]<='z')

    //         {

    //             s1[i]-=32;

    //         }

    //         else if (s1[i]>='A' && s1[i]<='Z')

    //         {

    //             s1[i]+=32;

    //         }

    //         else

    //         {

    //             cout<<"This is not an alphabetical character here"<<endl;

    //         }

    //     }

    //     cout<<"So, after the case changes the string we have is - "<<s1<<endl;//So, after the case changes the string we have is - ShUBHaM

    //     // Now by taking user input manualy -

    //     string s2;

    //     cout<<"What's in your minf Babes ..!!"<<endl;

    //     getline(cin, s2);

    //     for (int i = 0; i < s2.size(); i++)

    //     {

    //         if (s2[i]>='a' && s2[i]<='z')

    //         {

    //             s2[i]-=32;

    //         }

    //         else if (s2[i]>='A' && s2[i]<='Z')

    //         {

    //             s2[i]+=32;

    //         }

    //         else

    //         {

    //             cout<<"This is not an alphabetical character here"<<endl;

    //         }

    //     }

    //     cout<<"So, after the case changes the string we have is - "<<s2<<endl;

    /\*

    So, after the case changes the string we have is - ShUBHaM

    What's in your minf Babes ..!!

    nAmAn

    So, after the case changes the string we have is - NaMaN

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//  String another operations -**

**// String additioon -**

    string a = "Microsoft ", b = "Hyderabad";

    string c = a + " " + b;

    cout << c << endl; // Microsoft Hyderabad

    // w/o using any 3rd string

    a += b;

    cout << a << endl; /// Microsoft Hyderabad

**// Qun 9 - Remove Vowels, COnsonants from the given string.**

**// Remove vowels from string -**

    // string s = "Microsoft";

    // for (int i = 0; i < s.size(); i++)

    // {

    //     if (!(s[i] == 'a' ||s[i] == 'e' ||s[i] == 'i' ||s[i] == 'o' ||s[i] == 'u' || s[i] == 'A' ||s[i] == 'E' ||s[i] == 'I' ||s[i] == 'O' ||s[i] == 'U'))

    //     {

    //         cout<<s[i];

    //     }

    // }

    // o/p -  Mcrsft

    // \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Remove Consonants from string -**

    // string s = "Microsoft";

    // for (int i = 0; i < s.size(); i++)

    // {

    //     if ((s[i] == 'a' ||s[i] == 'e' ||s[i] == 'i' ||s[i] == 'o' ||s[i] == 'u' || s[i] == 'A' ||s[i] == 'E' ||s[i] == 'I' ||s[i] == 'O' ||s[i] == 'U'))

    //     {

    //         cout<<s[i];

    //     }

    // }

    // o/p - ioo

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 10 - Distinguish Vowels and Consonants in a given string -**

    // string s;

    // cout << "Buddy, what are you thinking about " << endl;

    // getline(cin, s);

    // string vwls, cnsnnts, NoALphaChar;

    // for (int i = 0; i < s.size(); i++)

    // {

    //     if ((s[i] >= 'a' && s[i] <= 'z') || (s[i] >= 'A' && s[i] <= 'Z') || s[i]==' ')

    //     {

    //         if (s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||

    //             s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U')

    //         {

    //             vwls += s[i];

    //         }

    //         else

    //         {

    //             cnsnnts += s[i];

    //         }

    //     }

    //     else

    //     {

    //         NoALphaChar+=s[i];

    //     }

    // }

    // cout << "So, all voweks of givenstring is - " << vwls << endl;

    // cout << "So, all consonants of givenstring is - " << cnsnnts << endl;

    // cout << "It's not an alphabetical character: " << NoALphaChar << endl;

    /\*

    Buddy, what are you thinking about

    Shubham Mahajan 880

    So, all voweks of givenstring is - uaaaa

    So, all consonants of givenstring is - Shbhm Mhjn

    It's not an alphabetical character: 880

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**// Qun 11 - common letters sequences in a string -**

**// Using shortcut menthod of STL -**

    // string text,pattern;

    // cin>>text>>pattern;

    // size\_t index = text.find(pattern,5);

    // if (index!=string::npos)

    // {

    //     cout<<index<<endl;

    // }

    /\*

    abcdefabcghijabc

    abc

    6

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 12 - For given two strings check that are rotation of each other or not?**

    // string two,one;

    // cout<<"write string one and two respectively "<<endl;

    // cin>>one>>two;

    // /\*

    // Rotation concept - Campus - 1st :-scampu, 2nd:-uscamp, 3rd:-puscam, 4th:-mpusca, 5th:-ampusc, 6th :=campus

    // 📌 Logic - if we write twuce a string then it can create all possible rotation of itself.

    // \*/

    // string check = one+one;

    // auto index = check.find(two);//using find function

    // if (index != string::npos)//STL Vector Part

    // {

    //     cout<<"Yes, they are rotation  of each other";

    // }

    // else

    // {

    //     cout<<"No, they are not";

    // }

    /\*

    write string one and two respectively

    abc

    bac

    No, they are not

    write string one and two respectively

    abc

    bca

    Yes, they are rotation  of each other

    \*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 13 - Enter the First Non-Repeating Character from the given string -**

//     string s3;

//     cout << "WHat's the text wants to share" << endl;

//     cin >> s3;

//     for (int i = 0; i < s3.size(); i++)

//     {

//         bool  repeating = false;

//         for (int j = 0; j < s3.size(); j++)

//         {

//             if (j!=i)// jis pr khade h usko check nhi krna h, otherwise matlb hi kya non repeatinhg check krne ka

//             {

//                 if (s3[j]==s3[i])

//                 {

//                     repeating = true;

//                     break; // agr ek baar repeating hua to agge bhui hoga hi              }

//             }

//         }

//     }

//     if (repeating == false)

//     {

//         cout<<s3[i]<<endl;

//         return 0;

//     }

// }

// cout<<"Not Found"<<endl;

/\*

WHat's the text wants to share

abdbcdaeghh

c

WHat's the text wants to share

naman

m

\*/

// \_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_

**//Qun 14- Find the First Repeating element of the string -**

// string s3;

// cout << "WHat's the text wants to share" << endl;

// cin >> s3;

// for (int i = 0; i < s3.size(); i++)

// {

//     for (int j = 0; j < s3.size(); j++)

//     {

//         if (j!=i)

//         {

//             if (s3[j]==s3[i])

//             {

//                 cout<<"First Repeating String is - "<<s3[i];

//                 return 0;

//             }

//         }

//     }

// }

/\*

WHat's the text wants to share

zxanfgung

First Repeating String is - n

WHat's the text wants to share

naman

First Repeating String is - n

WHat's the text wants to share

shubham

First Repeating String is - h

\*/

**// Method - 2**

// string s4;

// cout << "WHat's the text wants to share" << endl;

// cin >> s4;

// for (int i = 0; i < s4.size(); i++)

// {

//     int count=0;

//     for (int j = 0; j < s4.size(); j++)

//     {

//         if (s4[j]==s4[i])

//         {

//             count++;

//         }

//     }

//     if (count>1)

//     {

//         cout<<"First Repeated character is - "<<s4[i]<<"  and repeated - "<<count<<" times"<<endl;

//         return 0;

//     }

// }

// cout<<"Not found the repeated character"<<endl;

/\*

WHat's the text wants to share

shubham

First Repeated character is - h  and repeated - 2 times

WHat's the text wants to share

naman

First Repeated character is - n  and repeated - 2 times

WHat's the text wants to share

indrajeet

First Repeated character is - e  and repeated - 2 times

WHat's the text wants to share

Mahajan

First Repeated character is - a  and repeated - 3 times

WHat's the text wants to share

Pradhyumn

Not found the repeated character

\*/

}