**STL-Strings**

**Character Array**

* Need to know size beforehand.
* Larger size required for operation (concatenates or append).
* No terminating extra character.

**String**

* Need to know size beforehand.
* Performing operation like concatenating and append is easier.
* Terminated with a special character **‘\0’.**

To use strings in a program, you need to include a header called string.

**#include<string>**

# Declaring a string :-

**string str = 'rishabh';**

It declares a string of value “rishabh”

**string str(10);**

It declares a string of size 10.

**string s(5, 'N');**

It declares a string of size 5 with all characters ‘N’.

**string abc(str);**

It declares a copy of the string str.

**Different functions of string:-**

**getline()** function:- To **input the string** with space we use **getline**() function instead of cin.

string s;

getline(cin, s);

cout << "Rishabh is"<< s << endl;

**Input** : **a peaceful soul**

**Output**: Rishabh is **a peaceful soul**

**append() function**:- Inserts additional characters at the end of the string (can also be done using ‘+’ or ‘+=’ operator). Its time complexity is O(N) where N is the size of the new string. **Output**: *family*

**string s1 = "fam";**

**string s2 = "ily";**

**s1.append(s2);**

**cout << s1 << endl;**

**string s1 = "fam";**

**string s2 = "ily";**

**s1 = s1 + s2;**

**cout << s1 << endl;**

**stoi()** function:- Returns the strings converted to int datatype.

string s = "786";

int x = stoi(s);

cout << x+2 <<endl;

**Output**: 788

**Note**:- To convert an integer to a string, we use to\_string() function.

int x = 786;

cout << to\_string(x)+"2" <<endl;

**Output**: *7862*

**Sorting a string**:

We need to include a header file known as algorithm in our code like this.

#include<algorithm>

Then we can use **sort()** function that is present in above header file on our string. **Sort()** function takes 2 arguments viz. iterator to start of the string and iterator to end of the string.

string s = "ahfkhhfyuirkdswu786";

sort(s.begin(), s.end());

cout<<s<<endl;

**Output** : 678adffhhhikkrsuuwy

**13.2 String Challenges**

**UpperCase-LowerCase interconversion:-**

**Function used: Inbuilt function**

#include<bits/stdc++.h>

using namespace std;

int main(){

    string str = "ahwbdcgkeiozxybcd";

    sort(str.begin(), str.end());  //sorting string

    // convert to upper case

    transform(str.begin(), str.end(), str.begin(), ::toupper);

    cout<<"upper case : "<<str<<endl;

    // convert to lower case

    transform(str.begin(), str.end(), str.begin(), ::tolower);

    cout<<"lower case : "<<str<<endl;

    return 0;

}

Normal used:-

    // convert into upper case

    for(int i=0; i<str.size(); i++){

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

            str[i] -= 32;

    }

    cout<<"upper case: "<<str<<endl;

    // convert to lower case

    for (int i=0; i<str.size(); i++){

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

            str[i] += 32;

    }

    cout<<"lower case: "<<str<<endl;

# Q. Max Frequency:-

int main(){

    string s = "akjahhsahuhvizsw";

    int freq[26];

    for(int i=0; i<26; i++)

        freq[i] = 0;

    for(int i=0; i<s.size(); i++)

        freq[s[i]-'a']++;

    char ans = 'a';

    int maxF = 0;

    for(int i=0; i<26; i++){

        if(freq[i] > maxF){

            maxF = freq[i];

            ans = i+'a';

        }

    }

    cout<<maxF<<" "<<ans<<endl;

    return 0;

} **Output**: 4 h

**In-Build Function:-**

getline(cin, s);        // input string

s1.append(s2);          // insert s1+s2

s.assign("Ribh");       // s = "Ribh"

cout<<s.at(3)<< endl;   // particular position

s.begin();              // first character

s.end();                // last character

s.clear();              // Erases all the contents

s2.compare(s1)          // s2 - s1

s.c\_str();              // C-style string

s.empty();              // string is empty

s.erase();              // Deletes a substring

s.find();               // Searches the string

s.insert();             // Inserts additional characters

s.length();             // Returns the length of the string

s.size();               // Returns the length of the string

s.resize();             // Resize the string to the new length

s.substr(3, 4);         // copy of the string 3 to 4 is 3456

stoi(s);                // strings convert to a integer

to\_string(s);           // convert integer to a string

sort(s.begin(),s.end());// use sort() function

transform(s.begin(), s.end(), s.begin(), ::toupper); //convert upper

transform(s.begin(), s.end(), s.begin(), ::tolower); //convert lower

sort(s.begin(), s.end(), greater<int>()); //sorting greater to lower

sort(s.begin(), s.end(), greater<int>()); // 999887443 sort