

STRING USER INPUT IN C++:

Method 1:

```
#include <string>

string s;
getline(cin, s);

cout << s;
: OR: for s[i] != '10'
for(int i=0; i<s.size(); i++){
    cout << s[i];
}
```

Method 2:

```
#include <cstring>
int n; cin >> n;
char s[n];
cin.getline(s, n);

for(int i=0; i<strlen(s); i++){
    cout << s[i];
}
```

1) Program to remove character from string.

```

int main() {
    int i, len, j;
    string str;
    cout << "Enter any string";
    getline(cin, str);
    char ch;
    cin >> ch;
    len = str.size();
    for (i = 0; i < len; i++) {
        if (str[i] == ch) {
            for (j = i; j < len; j++) {
                str[j] = str[j + 1];
            }
            len--;
            i--;
        }
    }
    cout << "The final string after removing"
    << ch << " is " << str;
    return 0;
}

```

2). Program to count the occurrence of given character in string.

```

int main()
{
    int i, count = 0;
    string s;
    cout << "Enter String";
    getline(cin, s);
    char ch;
    cout << "Enter character to count
           its occurrence";
    cin >> ch;

    for (i = 0; s[i] != '\0'; i++) {
        if (s[i] == ch) {
            count++;
        }
    }

    if (count == 0) {
        cout << "Given char not found";
    } else {
        cout << count;
    }
    return 0;
}

```

3). Program to check whether the given string is Anagram or not. (Ex: chirag. garchi).

```

int main() {
    string s1, s2;
    int len1, len2, i, j, found = 0, notfound = 0;
    cout << "Enter first string";
    getline (cin, s1);
    cout << "Enter second string";
    getline (cin, s2);
    len1 = s1.size();
    len2 = s2.size();

    if (len1 == len2) {
        for (i = 0; i < len1; i++) {
            found = 0;
            for (j = 0; j < len1; j++) {
                if (s1[i] == s2[j]) {
                    found = 1;
                    break;
                }
            }
            if (found == 0) {
                notfound = 1;
                break;
            }
        }
        if (notfound == 1)
            cout << "Strings are not Anagram";
        else
            cout << "Strings are Anagram";
    } else
        cout << "Character count mismatched";
}
return 0;

```

4). Program to check given string is palindrome or not.

```

int main(){
    string s;
    int i, len, temp = 0;
    int flag = 0;
    cout << "Enter a string";
    cin >> getline(cin, s);
    len = s.size();
    for (i = 0; i < len; i++) {
        if (s[i] != s[len - i - 1]) {
            temp = 1;
            break;
        }
    }
    if (temp == 0) {
        cout << "PALINDROME!";
    } else {
        cout << "NOT A PALINDROME";
    }
    return 0;
}

```

5) Program to check given character is vowel or consonant.

```

int main() {
    char ch;
    cin >> ch;

    if ((ch >='a' && ch <='z') || (ch >='A' && ch <='Z'))
        if (ch == 'a' || ch == 'A' || ch == 'e' || ch == 'E'
            || ch == 'i' || ch == 'I' || ch == 'o' || ch == 'O'
            || ch == 'u' || ch == 'U')
            cout << "VOWEL";
        else
            cout << "CONSONANT";
    else
        cout << "Neither Consonant nor vowel";
    return 0;
}

```

6) Program to check given character is digit or not.

```

int main() {
    char ch;
    cin >> ch;

    if (ch >='0' && ch <='9')
        cout << "Given char is digit";
    else
        cout << "Not a digit";
    return 0;
}

```

7). Program to replace string space with given character.

```
int main() {
    string s;
    cout << "Enter string";
    getline(cin, s);
    char ch;
    cin >> ch;
    for (i = 0; s[i] != '\0'; i++) {
        if (s[i] == ' ')
            s[i] = ch;
    }
    cout << "Resultant string" << s;
}
```

8). Program to convert lowercase to uppercase of string

```
int main() {
    string s;
    int i;
    cout << "Enter string in lowercase";
    getline(cin, s);
    for (i = 0; i <= s.size(); i++) {
        if (s[i] >= 97 && s[i] <= 122) {
            s[i] = s[i] - 32;
        }
    }
    cout << "Entered string in uppercase." << s;
    return 0;
}
```

OR: for (i = 0; i < s.size(); i++)
 s[i] = toupper(s[i]);

toupper() converts character to uppercase.

9). Program to convert lowercase vowel into uppercase of string.

```

int main()
{
    string s; int i;
    cin.getline(cin.s);
    for (i=0; i < s.size(); i++) {
        if (s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o'
            || s[i] == 'u') {
            s[i] = s[i] - 32;
        }
    }
    cout << s;
    return 0;
}

```

10). Program to delete vowel from string.

```

int main()
{
    string s;
    cin.getline(cin.s);
    int len = s.size();
    for (int i=0; i < len; 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') {
            for (int j=i; j < len; j++) {
                s[j] = s[j+1];
            }
            i--;
            len--;
        }
    }
    cout << s;
}

```

3.

- 1). Program to count sequence of vowels and consonants in string.

```

int main()
{
    string s;
    int c=0, v=0;
    cout << "Enter string ";
    getline(cin,s);
    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] <= 'Z')) v++;
        else if ((s[i] >= 'a' & s[i] <= 'z') || (s[i] >= 'A' & s[i] <= 'Z')) c++;
    }
    cout << v << " " << c;
}

```

- 2). Program to find frequency of characters of string.

```

int main()
{
    string s;
    getline(cin,s);
    char ch;
    cin >> ch;
    int count=0;
    for (int i=0; i<s.size(); i++) {
        if (s[i] == ch) count++;
    }
    cout << count;
}

```

13). Program to replace first occurrence of vowel with '-' in string.

```
int main() {
    string s;
    cin >> s;
    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] = '-'; // Replacing vowel with '-'
            break; // Once vowel is found, break the loop
        }
    }
    cout << s;
}
```

3.

14). Program to separate characters in string

```
int main() {
    string s;
    getline(cin, s);
    for (int i=0; i<s.size(); i++) {
        cout << s[i] << " ";
    }
}
```

3.

15). Program to check given two strings are equal or not.

```

int main() {
    string s1, s2;
    getline (cin, s1);
    getline (cin, s2);
    int count = 0;
    if (s1.size() == s2.size()) {
        for (int i = 0; i < s1.size(); i++) {
            if (s1[i] == s2[i]) {
                count++;
            }
        }
        if (count == s1.size())
            cout << "strings are equal";
        else
            cout << "strings are not equal";
    }
    return 0;
}

: OR:

if (strcmp(s1, s2) == 0)
    "Same";

```

16). Program to remove blank space from string.

```

int main(){
    string s;
    getline(cin,s);
    int i=0, len, j;
    cout<<"Enter String";
    for(i=0;i<s.size();i++) {
        if(s[i]==' ') {
            for(j=i;j<s.size();j++) {
                s[j]=s[j+1];
            }
            len--;
        }
    }
    cout<<s;
    return 0;
}

```

17). Program to concatenate two strings.

```

int main()
{
    // Input two strings s1 and s2; and declare s3;
    int i=0, j=0;
    while(s1[i]!='\0') {
        s3[j]=s1[i];
        i++;
        j++;
    }
    // i=0; if i=0, then it will start from 0
    while(s2[i]!='\0') {
        s3[j]=s2[i];
        i++;
        j++;
    }
    s3[j]='\0';
    for(i=0; s3[i]!='\0'; i++)
        cout<<s3[i];
}

```

18). Program to remove repeated character from string.

```

int main() {
    string s;
    getline(cin, s);
    int len = s.size();
    for (int i=0; i<len; i++) {
        for (int j=i+1; j<len; j++) {
            if (s[i] == s[j]) {
                for (int k=j; k<len; k++) {
                    s[k] = s[k+1];
                }
                len--;
            }
        }
    }
    cout << s;
    return 0;
}

```

19). Program to calculate sum of integers in string.

```

int main() {
    string s; cin >> s;
    int len = s.size(); sum = 0;
    for (int i=0; i<len; i++) {
        if (isdigit(s[i])) {
            sum = sum + (s[i] - '0');
        }
    }
    cout << sum;
    return 0;
}

```

GONA ROOPA						
S	M	T	W	T	F	S
Page No.:	Date:	11				

Program to print all non-repeating characters in string.

int main()

string s;

int i;

int freq[256] = {0};

cout << "Enter a string";

getline (cin, s);

for (i = 0; s[i] != '\0'; i++) {

freq[s[i]]++;

3

for (i = 0; i < 256; i++) {

if (freq[i] == 1) is

cout << char(i) << " ";

3

3

return 0;

3.

Ans - (1) Output

Output to be fed into TCS final

Signature

21). Program to print the highest occurrence or frequency character or highest repeated and number of times repeated in a string.

```

int main()
{
    string s;
    int i;
    int freq[256] = {0};
    cout << "Enter a string";
    getline(cin, s);

    for(i=0; s[i] != '\0'; i++)
        freq[s[i]]++;

    cout << "The Repeating characters are:";

    for (i=0; i<256; i++) {
        if (freq[i] > 1) {
            cout << char(i) << endl;
            cout << freq[i] << " ";
        }
    }

    if (freq[i] == 0)
        cout << "No Repeating character!";
}

return 0;
}

```

21) Program to copy one string to other.

```
int main(){
    string s1,s2;
    getline(cin,s1);
    s2 = s1;
    cout << s2;
}
```

23) Program to sort the characters of a string.

```
int main(){
    string s;
    getline(cin,s);
    sort(s.begin(),s.end());
    cout << s;
}
```

In Descending order;

```
for(int i=s.size(); i>=0; i--) {
    cout << s[i];
}
```

3.

Using for loop (Ascending order):

```
for (i=0; i<s.size()-1; i++) {
    for (j=i+1; j<s.size(); j++) {
        if (s[i] > s[j]) {
            temp = s[i];
            s[i] = s[j];
            s[j] = temp;
        }
    }
}
```

3

3

cout << s;

For Descending order:

```
if (s[i] < s[j]) {
    temp = s[j];
    s[j] = s[i];
    s[i] = temp;
}
```

3

?

24). Program to count alphabets, digits and special characters.

```

int main() {
    string s;
    int a,d,spl;
    a=d=spl=0;i=0;
    cout << "Enter any string";
    getline(cin,s);
    while (s[i]!='\0') {
        if ((s[i]>='A' && s[i]<='Z') || (s[i]>='a' && s[i]<='z')) i++;
        else if (s[i]>='0' && s[i]<='9') d++;
        else spl++;
        i++;
    }
    cout << " " << a << " " << d << " " << spl;
    return 0;
}

```

25). Program to toggle each character in a string.

```

int main() {
    string s;
    int i;
    cout << "Enter string";
    getline(cin, s);
    for (i = 0; s[i] != '\0'; i++) {
        if (s[i] >= 'A' && s[i] <= 'Z') {
            s[i] = s[i] + 32;
        }
        else if (s[i] >= 'a' && s[i] <= 'z') {
            s[i] = s[i] - 32;
        }
    }
    cout << s;
    return 0;
}

```

26). Program to reverse a string.

```

int main() {
    string s;
    getline(cin, s);
    for (int i = (s.size() - 1); i >= 0; i--) {
        cout << s[i];
    }
    return 0;
}

```

26) Program to capitalize the first and last character of each word of string.

```

int main(){
    string s;
    getline(cin,s);
    int len = s.size();
    for (int i=0; i<len; i++) {
        if (i==0 || i==(len-1)) { //first & last index
            s[i] = toupper(s[i]); // to uppercase
        }
        else if (s[i] == ' ') { // before & after space
            s[i+1] = toupper(s[i+1]); // space to uppercase
        }
    }
    cout << s;
    return 0;
}

```

27). Program to replace substring in a string.

```

int main()
{
    string s, s2, s3;
    cout << "Enter the main string";
    getline(cin, s);
    cout << "Enter string to be replaced";
    getline(cin, s2);
    cout << "Enter the replacing string";
    getline(cin, s3);

    int s2len = s2.length();

    s.replace(s.find(s2), s2.size(), s3);

    cout << s;
    return 0;
}
  
```

28). Program to sort words in Dictionary order.

```

int main()
{
    string s[10], temp;
    int n;
    cout << "Enter no. of words to sort";
    cin >> n;

    cout << "Enter " << n << " words:" << endl;

    for (int i = 0; i < n; i++) {
        getline(cin, s[i]);
    }
  
```

```

for (int i=0; i<n; i++) {
    for (int j=0; j<n-i; j++) {
        if (s[j] > s[j+1]) {
            temp = s[j];
            s[j] = s[j+1];
            s[j+1] = temp;
        }
    }
}

cout << "In lexicographical order: ";
for (int i=0; i<=n; i++) {
    cout << s[i] << endl;
}
return 0;

```

Q9). Program to find ASCII value of total characters in a string.

```

int main() {
    string str;
    getline(cin, str);

    for (int i=0; str[i]!='\0'; i++) {
        cout << "The ASCII value of " << str[i] << " is " << int(str[i]) << endl;
    }
    return 0;
}

```

30). Program to count lines, words and characters in a given text.

```

int main()
{
    string str;
    getline(cin, str);
    int line, word, ch;
    line = word = ch = 0;

    for (int i=0; str[i] != '\0'; i++) {
        if (str[i] == '\n') {
            line++;
            word++;
        }
        else if (str[i] == ' ' || str[i] == '\t') {
            word++;
        }
        else {
            ch++;
        }
    }

    cout << ch << word << line;
    return 0;
}

```

31) Program to search a string in the list of strings.

```

int main()
{
    char str[20][30], s1[50];
    int n, i, found = 0;
    cout << "Enter how many strings(names):";
    cin >> n;
    cout << "Enter " << n << " strings";
    for (i = 0; i < n; i++)
        cin >> str[i];
    cout << "Enter a string to search";
    cin >> s1;

    for (i = 0; i < n; i++)
        if (strcmp(s1, str[i]) == 0) {
            found = 1;
            cout << "Found in row - " << i + 1;
        }

    if (found == 0)
        cout << "Not Found";
    return 0;
}

```