

Assignment - 07

Submitted by- Vineet Giri

Submitted to- Mr Naman

Chauhan

BCA 1st Year

Section- C

Program List

1. Write a program to input a string and find its length using strlen().

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100];
    cout << "Enter a string: ";
    cin.getline(str, 100);
    cout << "Length of string = " << strlen(str);
```

```
    return 0;
}
```

Output:

```
Enter a string: Hello World
Length of string = 11
```

2. Write a program to count the number of vowels as a string using `strlen()`.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100];
    int count = 0;
    cout << "Enter a string: ";
    cin.getline(str, 100);
    for (int i = 0; i < strlen(str); i++) {
        char ch = tolower(str[i]);
        if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' ||
ch == 'u')
            count++;
    }
    cout << "Number of vowels = " << count;
    return 0;
}
```

Output:

Enter a string: Education
Number of vowels = 5

3. Write a program to copy one string to another using strcpy().

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str1[100], str2[100];
    cout << "Enter a string: ";
    cin.getline(str1, 100);
    strcpy(str2, str1);
    cout << "Copied string: " << str2;
    return 0;
}
```

Output:

Enter a string: Hello
Copied string: Hello

4. Write a program to input two strings and check if they are equal using `strcpy()` and comparison.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str1[100], str2[100];
    cout << "Enter first string: ";
    cin.getline(str1, 100);
    cout << "Enter second string: ";
    cin.getline(str2, 100);
    if (strcmp(str1, str2) == 0)
        cout << "Strings are equal.";
    else
        cout << "Strings are not equal.";
    return 0;
}
```

Output:

```
Enter first string: Hello
Enter second string: Hello
Strings are equal.
```

5. Write a program to concatenate two strings using strcat().

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str1[100], str2[100];
    cout << "Enter first string: ";
    cin.getline(str1, 100);
    cout << "Enter second string: ";
    cin.getline(str2, 100);
    strcat(str1, str2);
    cout << "Concatenated string: " << str1;
    return 0;
}
```

Output:

```
Enter first string: Hello
Enter second string: World
Concatenated string: HelloWorld
```

6. Write a program to join first name and last name using strcat().

```
#include <iostream>
#include <cstring>
```

```
using namespace std;
int main() {
    char first[50], last[50];
    cout << "Enter first name: ";
    cin.getline(first, 50);
    cout << "Enter last name: ";
    cin.getline(last, 50);
    strcat(first, " ");
    strcat(first, last);
    cout << "Full name: " << first;
    return 0;
}
```

Output:

```
Enter first name: Vineet
Enter last name: Giri
Full name: Vineet Giri
```

7. Write a program to compare two strings using strcmp().

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str1[100], str2[100];
    cout << "Enter first string: ";
```

```

cin.getline(str1, 100);
cout << "Enter second string: ";
cin.getline(str2, 100);
int result = strcmp(str1, str2);
if (result == 0)
    cout << "Strings are equal.";
else if (result < 0)
    cout << "First string is smaller.";
else
    cout << "First string is greater.";
return 0;
}

```

Output:

```

Enter first string: Apple
Enter second string: Banana
The first string is smaller.

```

8. Write a program to check if two strings are equal using strcmp().

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char s1[100], s2[100];
    cout << "Enter first string: ";

```

```

cin.getline(s1, 100);
cout << "Enter second string: ";
cin.getline(s2, 100);
if (strcmp(s1, s2) == 0)
    cout << "Strings are equal.";
else
    cout << "Strings are not equal
(case-sensitive).";
return 0;
}

```

Output:

```

Enter first string: Hello
Enter second string: hello
Strings are not equal (case-sensitive).

```

9. Write a program to arrange an array of strings in alphabetical order using strcmp().

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char names[5][50], temp[50];
    cout << "Enter 5 names:\n";
    for (int i = 0; i < 5; i++)

```

```
        cin.getline(names[i], 50);
    for (int i = 0; i < 4; i++)
        for (int j = i + 1; j < 5; j++)
            if (strcmp(names[i], names[j]) > 0) {
                strcpy(temp, names[i]);
                strcpy(names[i], names[j]);
                strcpy(names[j], temp);
            }
    cout << "Names in alphabetical order:\n";
    for (int i = 0; i < 5; i++)
        cout << names[i] << endl;
    return 0;
}
```

Output:

Enter 5 names:

Vineet

Aman

Mohit

Ravi

Kunal

Names in alphabetical order:

Aman

Kunal

Mohit

Ravi

Vineet

10. Write a program to reverse a string using `strrev()`.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100];
    cout << "Enter a string: ";
    cin.getline(str, 100);
    strrev(str);
    cout << "Reversed string: " << str;
    return 0;
}
```

Output:

```
Enter a string: Hello
Reversed string: olleH
```

11. Write a program to check if a string is palindrome using `strrev()`.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100], rev[100];
```

```

    cout << "Enter a string: ";
    cin.getline(str, 100);
    strcpy(rev, str);
    strrev(rev);
    if (strcmp(str, rev) == 0)
        cout << "The string is a palindrome.";
    else
        cout << "The string is not a palindrome.";
    return 0;
}

```

Output:

```

Enter a string: level
The string is a palindrome.

```

12. Write a program to find the first occurrence of a character in a string using strchr().

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100], ch;
    cout << "Enter a string: ";
    cin.getline(str, 100);
    cout << "Enter a character: ";
}

```

```

    cin >> ch;
    char *ptr = strchr(str, ch);
    if (ptr)
        cout << "First occurrence of '" << ch << "' is at
position " << (ptr - str + 1);
    else
        cout << "Character not found.";
    return 0;
}

```

Output:

Enter a string: programming

Enter a character: g

First occurrence of 'g' is at position 4

13. Write a program to find the last occurrence of a character in a string using strrchr().

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100], ch;
    cout << "Enter a string: ";
    cin.getline(str, 100);
    cout << "Enter a character: ";
}

```

```

    cin >> ch;
    char *ptr = strrchr(str, ch);
    if (ptr)
        cout << "Last occurrence of " << ch << " is at
position " << (ptr - str + 1);
    else
        cout << "Character not found.";
    return 0;
}

```

Output:

```

Enter a string: programming
Enter a character: g
Last occurrence of 'g' is at position 11

```

14. Write a program to find a substring inside another using strstr().

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100], sub[100];
    cout << "Enter main string: ";
    cin.getline(str, 100);
    cout << "Enter substring: ";
    cin.getline(sub, 100);
}

```

```

char *ptr = strstr(str, sub);
if (ptr)
    cout << "Substring found at position " << (ptr -
str + 1);
else
    cout << "Substring not found.";
return 0;
}

```

Output:

```

Enter main string: programming
Enter substring: gram
Substring found at position 4

```

15. Write a program to check if one string is present in another using strstr().

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str1[100], str2[100];
    cout << "Enter main string: ";
    cin.getline(str1, 100);
    cout << "Enter string to search: ";
    cin.getline(str2, 100);
}

```

```
    if (strstr(str1, str2))
        cout << "String found.";
    else
        cout << "String not found.";
    return 0;
}
```

Output:

Enter main string: Hello World
Enter string to search: World
String found.

16. Write a program to count the number of words in a string using `strlen()` and spaces.

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[200];
    int count = 1;
    cout << "Enter a sentence: ";
    cin.getline(str, 200);
    for (int i = 0; i < strlen(str); i++) {
        if (str[i] == ' ')
            count++;
    }
}
```

```

    }
    cout << "Total words = " << count;
    return 0;
}

```

Output:

Enter a sentence: I love programming
Total words = 3

17. Write a program to remove all vowels from a string using string functions.

```

#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char str[100], res[100];
    cout << "Enter a string: ";
    cin.getline(str, 100);
    int j = 0;
    for (int i = 0; i < strlen(str); i++) {
        char ch = tolower(str[i]);
        if (!(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' ||
ch == 'u'))
            res[j++] = str[i];
    }
    res[j] = '\0';
}

```

```
    cout << "String without vowels: " << res;  
    return 0;  
}
```

Output:

Enter a string: Education
String without vowels: dctn

18. Write a program to copy only the first 5 characters of one string to another using `strncpy()`.

```
#include <iostream>  
#include <cstring>  
using namespace std;  
int main() {  
    char src[100], dest[100];  
    cout << "Enter a string: ";  
    cin.getline(src, 100);  
    strncpy(dest, src, 5);  
    dest[5] = '\0';  
    cout << "Copied string (first 5 chars): " << dest;  
    return 0;  
}
```

Output:

Enter a string: Blackmagic

Copied string (first 5 chars): Black

19. Write a program to join multiple strings into one sentence using strcat().

```
#include <iostream>
#include <cstring>
using namespace std;
int main() {
    char s1[50] = "I ", s2[50] = "love ", s3[50] =
"coding.";
    strcat(s1, s2);
    strcat(s1, s3);
    cout << "Final sentence: " << s1;
    return 0;
}
```

Output:

Final sentence: I love coding.

20. Write a program to check whether two strings are anagrams using string functions.

```
#include <iostream>
#include <cstring>
#include <algorithm>
using namespace std;
int main() {
    char str1[100], str2[100];
    cout << "Enter first string: ";
    cin.getline(str1, 100);
    cout << "Enter second string: ";
    cin.getline(str2, 100);
    if (strlen(str1) != strlen(str2)) {
        cout << "Not anagrams.";
        return 0;
    }
    sort(str1, str1 + strlen(str1));
    sort(str2, str2 + strlen(str2));
    if (strcmp(str1, str2) == 0)
        cout << "Strings are anagrams.";
    else
        cout << "Strings are not anagrams.";
    return 0;
}
```

Output:

```
Enter first string: listen
Enter second string: silent
Strings are anagrams.
```

