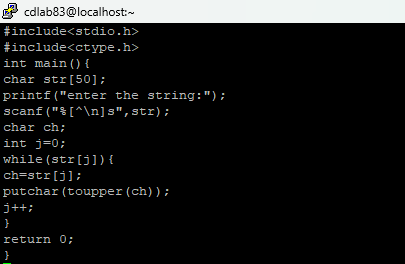
**EXP NO:1 DATE:14/2/25 DEVELOP A SIMPLE C PROGRAM TO DEMONSTRATE A BASIC STRING OPERATIONS**

**1.1 Input and Output**

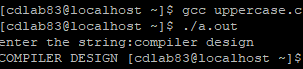
Question: Modify the program to take a string input from the user and display it in

uppercase.

Hint: Use the toupper function from <ctype.h> to convert characters to uppercase.



**OUTPUT:**

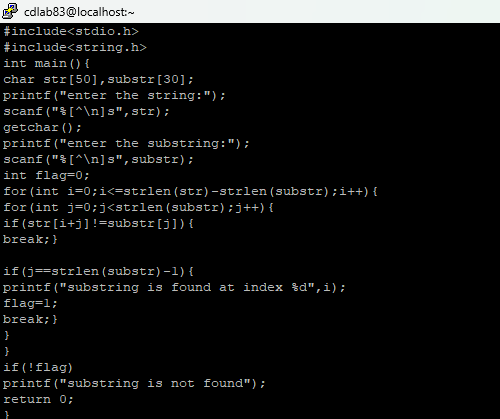


**1.2 String Length**

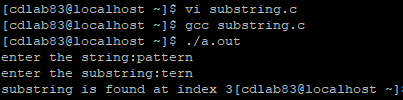
Question: Write a C program to check if a given substring exists within a string without

using the strstr() function. If the substring is found, print its starting index; otherwise,

print "Substring not found."



**OUTPUT:**

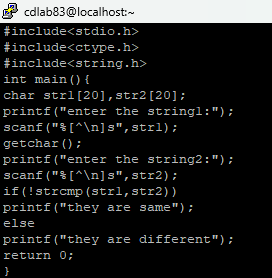


**1.3 String Comparison**

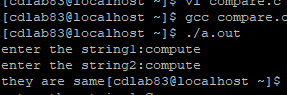
Question: Extend the program to compare two strings entered by the user and print

whether they are the same.

Hint: Use the strcmp function from <string.h> for comparison.



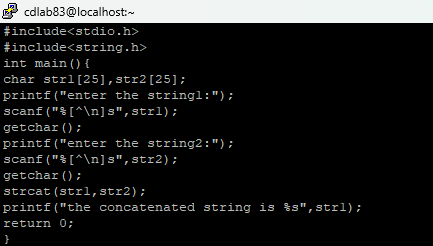
**OUTPUT:**



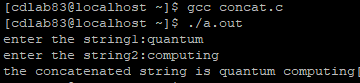
**1.4 Remove Spaces**

Question: Write a program to remove all spaces from a string entered by the user.

Hint: Use a loop to copy non-space characters to a new string.



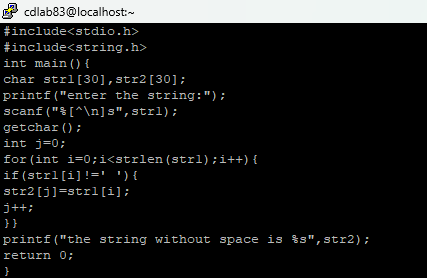
**OUTPUT:**



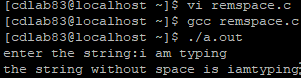
**1.5 Frequency of Characters**

Question: Modify the program to calculate the frequency of each character in the string.

Hint: Use an array of size 256 to store the count of each ASCII character.



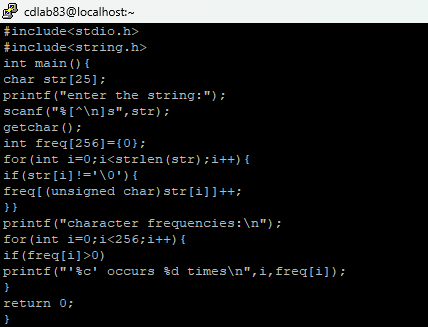
**OUTPUT:**



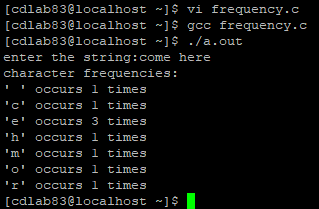
**1.6 Concatenate Strings**

Question: Extend the program to concatenate two strings entered by the user.

Hint: Use the strcat function from <string.h>.



**OUTPUT:**

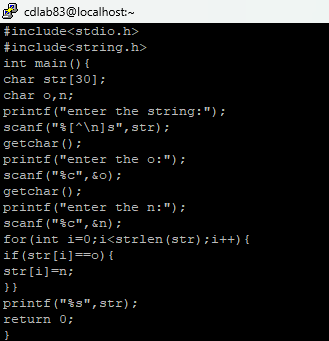


**1.7 Replace a Character**

Question: Write a program to replace all occurrences of a specific character in the

string with another character.

Hint: Traverse the string and replace the character conditionally in a loop.



**OUTPUT:**

