

Answer on Question#50602 – Programming - C++

Make your own version of the library function `strcmp(s1, s2)`, which compares two strings and returns `-1` if `s1` comes first alphabetically, `0` if `s1` and `s2` are the same, and `1` if `s2` comes first alphabetically. Call your function `compstr()`. It should take two `char*` strings as arguments, compare them character by character, and return an `int`. Write a `main()` program to test the function with different combinations of strings. Use pointer notation throughout.

Program:

```
#include <iostream>
#include <string>
#define MAX_LEN 100

using namespace std;

int strcmp(char* s1, char *s2)
{
    int i;
    for(i=0;;i++)
    {
        if((s1[i]=='\0') && (s2[i]=='\0'))
            return 0;
        if(s1[i]=='\0')
            return 1;
        if(s2[i]=='\0')
            return -1;
        if(s1[i]<s2[i])
            return 1;
        if(s1[i]>s2[i])
            return -1;
    }
    return 0;
}

int main()
{
    char str1[MAX_LEN], str2[MAX_LEN];
    cout<<"Input first string: ";
    cin>>str1;
    cout<<"Input second string: ";
    cin>>str2;
    int res=strcmp(str1, str2);
    if(res==0)
        cout<<"Strings are same";
    else
        if(res==1)
            cout<<"First string comes first alphabetically";
        else
            cout<<"Second string comes first alphabetically";

    return 0;
}
```

Answer:

```
Input first string: alph2
Input second string: alph1
Second string comes first alphabetically
```

```
Input first string: al
Input second string: alph
First string comes first alphabetically
```

```
Input first string: alp
Input second string: alp
Strings are same
```

<https://www.AssignmentExpert.com>