```
Script started on 2024-04-11 17:58:46-05:00 [TERM="xterm" TTY="/dev/pts/4" COLUMNS=
ee43254@ares:~$ pwd
/home/students/ee43254
ee43254@ares:~$ cat strextra.info
Name: Kyle Enkhzul

Class: CSC122-W01



Activity: "Hi" == "hi"?

Level: 1.5, 1.5 (base program),



Description:



This lab provides practice with string manipulation and libraries. We are to

write a function that allows case-in sensitive comparisons of two strings and

should be similar to the external library functions of strcasecmp or stricmp.

ee43254@ares:~$ show-code strextra.cpp


strextra.cpp:


```

```cpp
#include <iostream>
#include "strextra.h"

using namespace std;

int main() {
    const char* str1 = "Hello";
    const char* str2 = "hello";

        const char* str3 = "Bye";
    const char* str4 = "Goodbye";

        const char* str5 = "Zinc";
    const char* str6 = "Thanks";

        if(strcmp_ncase(str1, str2) == 0) {
                cout << str1 << " and " << str2 << " are the same!\n";
        }
        else {
                cout << str1 << " and " << str2 << " are not the same!\n";
        }

        if(strcmp_ncase(str3, str4) == 0) {
                cout << str3 << " and " << str4 << " are the same!\n";
        }
        else {
                cout << str3 << " and " << str4 << " are not the same!\n";
        }

        if(strcmp_ncase(str5, str6) == 0) {
                cout << str5 << " and " << str6 << " are the same!\n";
        }
        else {
                cout << str5 << " and " << str6 << " are not the same!\n";
        }

    return 0;
}
```

```
ee43254@ares:~$ show-code strextra.h


strextra.h:


```

```cpp
#ifndef STREXTRA_H_INC
#define STREXTRA_H_INC

#include <iostream>
#include <cctype>

short strcmp_ncase(const char *s1, const char *s2) {
        while (*s1 && *s2) {
     char c1 = static_cast<char>(std::tolower(static_cast<unsigned char>(*s
     char c2 = static_cast<char>(std::tolower(static_cast<unsigned char>(*s
        if (c1 != c2)
            return c1 - c2;
        s1++;
        s2++;
    }
    return *s1 - *s2;
}

#endif
```

```
ee43254@ares:~$ CPP strextra
strextra.cpp***


ee43254@ares:~$ ./strextra.out
Hello and hello are the same!
Bye and Goodbye are not the same!
Zinc and Thanks are not the same!
ee43254@ares:~$ cat strextra.tpq
1. How do you compare two characters without reference to case?

How might you do this without destroying the character variable(s) contents?
```

To compare two characters without regard to case, you can convert each character to lowercase using std::tolower and then compare them. To avoid modifying the original characters, cast them to unsigned char before converting.

2. How can you compare two strings in a case-insensitive way without destroying their contents? (You should not change the strings in order to compare them!)

You can compare two strings in a case-insensitive way by converting each character to lowercase before comparing them. This can be done without changing the original strings by using temporary variables for comparison.

3. What kind of arguments should your string comparison function take? (Value, reference, constant?)

A string comparison function should take constant pointers to the strings being compared to indicate that the function does not modify the strings.

4. How do you get that weird return value for your function?
Is it always -1, 0, 1? Or is there a reason it was defined as simply less than 0, 0, or greater than 0?

The return value of the string comparison function indicates whether the first string is less than, equal to, or greater than the second string. It is not necessarily limited to -1, 0, or 1; any negative value typically indicates the first string is "less than" the second string, while a positive value indicates the first string is "greater than" the second string.

5. How many times will you need to call your function to test it thoroughly? How many times should you have to run the driver to do this testing?

To test the function thoroughly, you would need to test it with different combinations of strings to cover all possible cases, including equal strings, strings with different lengths, and strings with different character cases. You should run the driver at least three different times to showcase the three different outcomes that could happen. ee43254@ares:~$ exit
exit

Script done on 2024-04-11 17:59:10-05:00 [COMMAND_EXIT_CODE="0"]