

```

1 //
2 // file: strcmp.c
3 // Created by hfwei on 2022/11/29.
4 //
5
6 #include <stdio.h>
7
8 int StrCmp(const char *s1, const char *s2);
9 int StrCmpStd(const char *s1, const char *s2);
10
11 int StrNCmpStd(const char *s1, const char *s2, int n);
12
13 int main() {
14     const char *str1 = "hi, C";
15     const char *str2 = "hi, c";
16
17     printf("StrCmp(\"%s\", \"%s\") = %d\n",
18           str1, str2, StrCmp(str1, str2));
19     // printf("StrCmpStd(\"%s\", \"%s\") = %d\n",
20     //        str1, str2, StrCmpStd(str1, str2));
21     //
22     // int n = 2;
23     // printf("StrNCmpStd(\"%s\", \"%s\", %d) = %d\n",
24     //        str1, str2, n, StrNCmpStd(str1, str2, n));
25
26     return 0;
27 }
28
29 // Wrong Version
30 // hi vs. hi ('\0')
31 // int StrCmp(const char *s1, const char *s2) {
32 //     while (*s1 == *s2) {
33 //         s1++;
34 //         s2++;
35 //     }
36 //
37 //     return *s1 - *s2;
38 // }
39
40 //
41 int StrCmp(const char *s1, const char *s2) {
42     while (*s1 == *s2 && (*s1 != '\0' && *s2 != '\0')) {
43         s1++;
44         s2++;
45     }
46
47     if (*s1 == '\0' && *s2 == '\0') {
48         return 0;
49     }
50
51     // char: unsigned char, signed char
52     return (*(const unsigned char *) s1)
53            < (*(const unsigned char *) s2) ? -1 : 1;

```

```

54 }
55
56 // See https://en.cppreference.com/w/c/string/byte/strcmp
57 //
58 // Compares two null-terminated byte strings lexicographically.
59 // The sign of the result is the sign of the difference between the
    values of the first pair of characters (both interpreted as unsigned
    char) that differ in the strings being compared.
60 // The behavior is undefined if lhs or rhs are not pointers to null-
    terminated byte strings.
61 int StrCmpStd(const char *s1, const char *s2) {
62     for (; *s1 == *s2; s1++, s2++) {
63         if (*s1 == '\0') {
64             return 0;
65         }
66     }
67
68     return (*(const unsigned char *) s1)
69         < (*(const unsigned char *) s2) ? -1 : 1;
70 }
71
72 // See https://en.cppreference.com/w/c/string/byte/strncmp
73 // Compares at most count characters of two possibly null-terminated
    arrays.
74 // The comparison is done lexicographically. Characters following the
    null character are not compared.
75 int StrNCmpStd(const char *s1, const char *s2, int n) {
76     for (; 0 < n; n--, s1++, s2++) {
77         if (*s1 != *s2) {
78             return (*(const unsigned char *) s1)
79                 < (*(const unsigned char *) s2) ? -1 : 1;
80         } else if (*s1 == '\0') { // *s1 == *s2 == '\0'
81             return 0;
82         }
83     }
84
85     return 0;
86 }

```