

Freiwillige Offline-Aufgabe 08-05 (INF & WI):

C-String-Funktion `my_strcmp()`

(geübte C++ Konstrukte: C-Strings, Funktionen mit Pointern als Parametern)

Programmieren Sie eine Funktion ...

```
int my_strcmp (const char * ptr1, const char * ptr2)
```

... welche zwei C-Strings vergleicht.

Sind die beiden C-String gleich, so gibt die Funktion den Rückgabewert 0 zurück. Dies gilt insbesondere auch dann, wenn beide C-Strings leer sind. Ansonsten werden die Zeichen an der ersten Position verglichen, an der die beiden C-Strings unterschiedlich sind: Sollte das Zeichen im ersten C-String kleiner sein, so wird der Wert -1 zurückgegeben. Sollte das Zeichen im ersten C-String größer sein, so wird der Wert 1 zurückgegeben. Dabei können Sie die Zeichen in ihrem Programm mittels < bzw. > vergleichen. (Dabei verwendet das System intern wieder die ASCII Tabelle, wodurch die Großbuchstaben kleiner sind als die Kleinbuchstaben!)

Mittels des `const` bei der Parameterdeklaration „verspricht“ die Funktion, die Werte der Parameter innerhalb des Funktionsrumpfes nicht zu ändern.

Die Benutzung jeglicher Funktionen aus der `cstring` Library ist nicht gestattet. Benutzen Sie nur den Zugriff auf die einzelnen Zeichen des C-Strings über den Array-Indexoperator `[]`. Sollten Sie die Länge eines C-Strings in ihrem Programm benötigen, so müssen Sie auch dies als Hilfsfunktion selbst programmieren.

Schreiben Sie ferner ein Hauptprogramm, welches zwei einzeilige Zeichenketten (ggfs. leer, ggfs. inklusive Leerzeichen) einliest und das Ergebnis des Vergleichs ausgibt (siehe Testläufe). Die eingegebenen Zeichenketten seien jeweils maximal 20 für den Benutzer sichtbare Zeichen lang.

Testläufe (Benutzereingaben sind unterstrichen):

```
Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? Hallo
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? Hallo
Die Texte sind identisch. Ergebnis my_strcmp(): 0
Drücken Sie eine beliebige Taste . . .
```

Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? *(leere Eingabe)*
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? *(leere Eingabe)*
Die Texte sind identisch. Ergebnis my_strcmp(): 0
Drücken Sie eine beliebige Taste . . .

Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? Hallo
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? *(leere Eingabe)*
Ergebnis my_strcmp(): 1
Drücken Sie eine beliebige Taste . . .

Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? *(leere Eingabe)*
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? Hallo
Ergebnis my_strcmp(): -1
Drücken Sie eine beliebige Taste . . .

Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? abcde
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? xyz
Ergebnis my_strcmp(): -1
Drücken Sie eine beliebige Taste . . .

Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? ABC
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? abc
Ergebnis my_strcmp(): -1
Drücken Sie eine beliebige Taste . . .

Bitte ersten Text eingeben (ggfs. mit Leerzeichen): ? ab cd
Bitte zweiten Text eingeben (ggfs. mit Leerzeichen): ? abxcd
Ergebnis my_strcmp(): -1
Drücken Sie eine beliebige Taste . . .
