

Erarbeitet von: M.Eng. Michael Finsterbusch
Modulverantwortlicher: Prof. Dr. rer. nat. Matthias Krause
Stand: 3. Oktober 2017

Ziel der Übung ist das Kennenlernen und Vertiefen von:

- Strings
- Arrays
- Schleifen
- Kontrollstrukturen
- Bitmasken

Abgabe: • <http://praktomat.hft-leipzig.de> unter „DKMI/DAI-17 C-Progr. Übung 7“
• sämtliche Abgabemodalitäten sind im Praktomat hinterlegt

Aufgaben

Gegeben ist die Header-Datei *uebung7.h*, in der Funktionsdeklarationen enthalten sind, sowie die Datei *uebung7.c*, in der die Funktionen implementiert werden. Kopieren Sie diese in Ihr Arbeitsverzeichnis. Die Funktionen sollen entsprechen den folgenden Vorgaben implementiert werden. Um die korrekte Funktionsweise Ihrer Implementation zu testen, verwenden Sie die Funktionen in der *main()*-Funktion, die in der Datei *main.c* implementiert werden soll.

Verwenden Sie in dieser Übung nicht die standard String-Funktionen aus **string.h**!

1. Implementieren Sie die Funktion `my_strlen()` in der Datei *uebung7.c*. Diese Funktion soll die Länge eines Null-terminierten String zurückgeben. Der Rückgabewert ist -1, wenn der übergebene String-Pointer NULL ist.
2. Implementieren Sie die Funktion `my_toupper()` in der Datei *uebung7.c*. Die Funktion erhält einen Null-terminierten String. In diesem sollen alle Kleinbuchstaben durch Großbuchstaben ersetzt werden.

Hinweis: Ob ein Zeichen (`char`) ein Kleinbuchstabe, lässt sich z.B. durch `if(c>='a' && c<='z')` ermitteln.

Hinweis: Die Kodierung eines Klein- bzw. Großbuchstaben unterscheidet sich nur in Bit 6. Dass heißt, wird Bit 6 auf Null gesetzt, wird aus einem Kleinbuchstaben ein Großbuchstabe. Prüfen Sie ob der String-Pointer NULL ist!

3. Implementieren Sie die Funktion `my_tolower()` in der Datei *uebung7.c*. Die Funktion soll im gegebenen String alle Großbuchstaben durch Kleinbuchstaben ersetzen. Prüfen Sie ob der String-Pointer NULL ist!
4. Implementieren Sie die Funktion `my_strstr()` in der Datei *uebung7.c*. Der Funktion werden zwei Strings übergeben. Im ersten String soll der zweite String gefunden werden. Wird der String gefunden, soll ein Zeiger auf die Startposition im ersten String zurückgegeben werden. Wird der zweite String nicht gefunden, soll NULL zurückgegeben werden. Ist einer der beiden übergebenen String-Pointer NULL, soll ebenfalls NULL zurückgegeben werden.

5. Implementieren Sie die Funktion `palindrome_test()` in der Datei `uebung7.c`. Diese soll ermitteln, ob der übergebene String ein Palindrom ist. Ein Palindrom ist eine Zeichenkette, die von vorn und von hinten gelesen dasselbe ergibt.
6. Implementieren Sie die Funktion `passwd_gen()` in der Datei `uebung7.c`, mit der (pseudo-) zufällige Passwörter generiert werden können. Der Funktion wird ein Puffer übergeben, der das Passwort aufnehmen kann, die Länge des zu generierenden Passworts sowie verschiedene Zeichenklassen, die in dem Passwort enthalten sein sollen. Die Zeichenklassen werden als Bitmaske übergeben.

Die Funktion soll einen nullterminierten String zurückliefern. Dass heißt, nach dem letzten Zeichen des Passworts muss das `'\0'`-Zeichen gesetzt werden.

Zu Beginn soll ein Array (z.B. `char CharSet[256]`) entsprechend den übergebenen Zeichenklassen mit Buchstaben, Zahlen und Sonderzeichen initialisiert werden. Mit Hilfe eines Zufallsgenerators (die Funktion `rand()`, siehe Dokumentation) wird dann der Index für die einzelnen Zeichen des Passworts ausgewählt. Beispiel:

```
1  enum CharSet {
2      LOWER_LETTERS    = (1<<0),    /* a-z */
3      CAPITAL_LETTERS  = (1<<1),    /* A-Z */
4      DIGITS           = (1<<2),    /* 0-9 */
5      SPECIAL_LETTERS  = (1<<3)     /* #+*-_... */
6  };
7  [...]
8  char passwd[32];

10  passwd_gen(passwd, 10, LOWER_LETTERS|CAPITAL_LETTERS|DIGITS|
11      ↳SPECIAL_LETTERS);
12  printf("Passwort: %s\n", passwd);
13  //Ausgabe: Passwort: fm3Le#PVkir

14  passwd_gen(passwd, 12, DIGITS);
15  printf("Passwort: %s\n", passwd);
16  //Ausgabe: Passwort: 164881573678

18  passwd_gen(passwd, 8, LOWER_LETTERS|CAPITAL_LETTERS);
19  printf("Passwort: %s\n", passwd);
20  //Ausgabe: Passwort: fvAkPart
```