



Aufgabe 1. Implementiere die Funktion, die den Inhalt zweier `int`-Variablen vertauscht.

Aufgabe 2. Schreibe Funktionen `square_to` und `root_to`, die einen `double`-Pointer entgegen nehmen, die dort stehende Variable quadrieren bzw. daraus die Wurzel ziehen und das Ergebnis sowohl zurück geben als auch an die gleiche Speicherstelle schreiben.

Aufgabe 3. Erweitere das „mymath“-Modul noch um eine Funktion, die zu den drei Koeffizienten $a, b, c \in \mathbb{R}$ einer quadratischen Gleichung

$$a \cdot x^2 + b \cdot x + c = 0$$

die Lösungen berechnet.

Aufgabe 4. Schreibe ein Modul `arrayhelpers`, das einige nützliche Funktion zum `int`-Array-Handling enthält:

- a) Array zeilenweise oder mit Kommata getrennt ausgeben
- b) Array sortieren
- c) Alle Felder eines Arrays mit einem Wert initialisieren
- d) Array um 1 rotieren (d.h. das hinterste Element an erste Stelle schreiben und alle anderen Elemente um eins nach hinten schieben)
- e) Array um k rotieren
- f) Array umdrehen
- g) Ein Array in einem anderen suchen und die Position zurück geben. Sollte das Array nicht im anderen enthalten sein, so soll der Rückgabewert -1 sein.

Beispiel:

```
1 int A[10] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
2 int B[3] = {4, 5, 6}
3 int C[2] = {5, 7}
4 int D[2] = {9, 10}
```

Hier gilt: B ist an 3-ter Stelle in A enthalten und D an 8-ter. Das Array C ist garnicht in A enthalten, darum wird der Rückgabewert -1 sein.



Aufgabe 5. Implementiere eine c-Datei zu folgender Header-Datei:

```
1  /* gibt die Laenge eines Strings zurueck */
2  int str_len(char *s);
3
4  /* gibt 0 zurueck, wenn zwei strings gleich
5   * sind und 1 sonst */
6  int str_cmp(char *s1, char *s2);
7
8  /* kopiert s nach d und gibt d zurueck */
9  char *str_cpy(char *d, char *s);
10
11 /* haege s2 and s1 an und gib s1 zurueck */
12 char *str_cat(char* s1, char* s2);
```

und teste deinen Code mit folgendem Modul:

```
1  #include <stdio.h>
2  #include "mystrings.h"
3
4  int main() {
5      char p[100] = "Pepsi_";
6      char c[100] = "Coca_";
7      char suffix[10] = "Cola";
8      char out[100];
9      str_cpy(out,p);
10     str_cat(out,suffix);
11     str_cpy(p,out);
12     str_cpy(out,c);
13     str_cat(out,suffix);
14     str_cpy(c,out);
15     if (str_cmp(p,c)) {
16         printf("%s",p);
17         printf("_is_not_");
18         printf("%s",c);
19         printf("\n");
20     }
21     return 0;
22 }
```