

Aufgabe 1. ▲ ▲ Lege ein Modul `mymath.c` / `mymath.h` an, in dem du die bisher geschriebenen Funktionen auslagerst. Implementiere ausserdem die folgenden Funktionen:

- a) $\log_a(x)$ mit beliebigen Argumenten $a \in \mathbb{R}$ und $x \in \mathbb{R}^+$
- b) Fakultät $n!$ für $n \in \mathbb{N}$ (falls nicht schon lange geschehen)
- c) für die k -te Wurzel aus $x \in \mathbb{R}_0^+$ mit $k \in \mathbb{R}^+$
- d) eine Funktion, die zu zwei Seiten eines Dreiecks und ihrem eingeschlossenen Winkel die Länge der dritten Seite zurück gibt

Wir brauchen im folgenden eine Potenzfunktion, die zwei Fließkommazahlen als Argumente akzeptiert. Falls du diese Funktion gestern geschrieben hast, sollte sie jetzt im `mymath`-Modul verfügbar sein. Diese Funktion wird aber vermutlich zu langsam sein, daher gibt es die Funktion

```
double pow(double x, double y);
```

in der Systemheader `<math.h>`. Im Skript findest du bei Interesse im Anhang eine Referenz einiger Systembibliotheken.

Aufgabe 2. ▲ ▲ 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 und die größere Lösung zurück gibt.

Aufgabe 3. ▲ – ▲▲

Schreiben Sie ein C-Programm, welches die n -te Zahl der Fibonacci-Folge ausgibt. Die Fibonacci-Folge ist dadurch gekennzeichnet, dass das n -te Folgenglied durch die Addition des $n-1$ + $n-2$ ergibt.

Die Folge fängt so an : 1 1 2 3 5 8 13 Implementieren sie die Funktion `int fib(n)` mit `fib(0) = 0`, `fib(1) = 1` und `fib(n) = fib(n-1) + fib(n-2)`.

- a) Implementatation mit Hilfe von einer rekursiven Funktion ▲ – ▲▲
- b) Implementatation über eine Schleife ▲ – ▲▲