

# Übungen zu Systemnahe Programmierung in C

## Abschnitt 5.3: Hands-On (Statistikmodul)

---

25.05.2020

Tim Rheinfels  
Benedict Herzog  
Bernhard Heinloth

Lehrstuhl für Informatik 4  
Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme  
und Betriebssysteme



FRIEDRICH-ALEXANDER  
UNIVERSITÄT  
ERLANGEN-NÜRNBERG

TECHNISCHE FAKULTÄT



- Modul `statistic` für Statistikfunktionen
  - Minimum
  - Maximum
  - Arithmetisches Mittel
- Programm um Modul zu testen



```
01 // Beschreibt Gueltigkeit des Ergebnisses
02 typedef enum {
03     STAT_OK = 0,    // Erfolg. Ergebnis gueltig.
04     STAT_ERROR = 1 // Fehler. Ergebnis ungueltig.
05 } stat_result_t;
06
07 // (Modul-) Schnittstelle
08 stat_result_t stat_min(const uint16_t *x, size_t N, uint16_t *r);
09 stat_result_t stat_max(const uint16_t *x, size_t N, uint16_t *r);
10 stat_result_t stat_avg(const uint16_t *x, size_t N, uint16_t *r);
```

## ■ Parameter:

- x: Eingabefeld
- N: Anzahl Elemente in x
- r: Zeiger auf Ergebnis

## ■ Rückgabewert:

- STAT\_OK: Das Ergebnis steht in \*r
- STAT\_ERROR: Der Wert \*r ist ungültig



```
01 // (Interne) Hilfsfunktionen
02 static uint16_t getMin(uint16_t a, uint16_t b);
03 static uint16_t getMax(uint16_t a, uint16_t b);
```

- Hilfsfunktionen für Teilergebnisse:
  - Minimum zweier Zahlen
  - Maximum zweier Zahlen
- Nicht Teil der Schnittstelle
  - ⇒ Modul-lokale Funktionen



1. Schnittstelle von `statistic` im Header `statistic.h`
  - Typdefinition für Rückgabewert
  - Funktionsdeklarationen
  - Include-Guard
2. Implementierung von `statistic` in Quelldatei `statistic.c`
  - Hilfsfunktionen
  - Modulfunktionen
3. Testprogramm in Quelldatei `test-statistic.c`
  - Definition von Testfällen
  - Ausgabe ihrer Ergebnisse