

Übungen zu Systemnahe Programmierung in C

Abschnitt 10.3: Dynamische Speicherverwaltung

29.06.2020

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





- `malloc(3)` allokiert Speicher auf dem Heap
 - Signatur: `void *malloc(size_t size)`
 - Reserviert mindestens `size` Byte Speicher
 - Liefert Zeiger auf diesen Speicherbereich zurück
 - Schlägt potenziell fehl
- `free(3)` gibt Speicher wieder frei

```
01 // Allokieren
02 char *s = (char *) malloc(12 * sizeof(char));
03 if(s == NULL) {
04     perror("malloc");
05     exit(EXIT_FAILURE);
06 }
07
08 // Verwenden
09 strcpy(s, "Hallo");
10 strcat(s, " Welt!");
11
12 // Freigeben
13 free(s);
```