

CDS

Praktische Übungsserie 1 **Prozesse und Threads**

Aufgabe 1 (Prozesse):

Schreibe ein C-Programm auf einem *nix-System, welches mittels *fork* einen *child process* erstellt, der schliesslich zu einem Zombie-Prozess wird. Dieser Zombie-Prozess soll für mindestens 10 Sekunden im System bleiben.

Du kannst Prozesse mittels `ps -u` anzeigen lassen. Prozesse mit Status `z` sind Zombie-Prozesse. Um den Zombie-Prozess los zu werden, musst du den zugehörigen `parent process` beenden.

Benutze das Skeleton `zombies.c`

Befehl zum Kompilieren: `gcc -Wall -std=c99 ./zombies.c -o zombies`

Zombie-Processes: https://en.wikipedia.org/wiki/Zombie_process

Aufgabe 2 (Threads):

Schreibe ein Programm mit mehreren Threads, das verschiedene statistische Werte berechnet zu einer gegebenen Liste von Zahlen berechnet. Das Programm nimmt als Kommandozeilen-Argumente zwei oder mehr Parameter entgegen und dann drei arbeitende Threads erstellen: Ein Thread berechne das arithmetische Mittel der zahlen, ein weiterer Thread bestimme das Maximum und ein dritter Thread bestimme das Minimum. Die Eingabe muss überprüft werden (mindestens zwei Argumente und nur Zahlen).

Beispiel:

Angenommen, die Eingabewerte seien: `90 81 78 95 79 72 85`

Dann könnte eine mögliche Ausgabe wie folgt aussehen:

```
Average: 82.857140
Maximum: 95
Minimum: 72
```

Benutze das Skeleton `statistics.c`

Befehl zum Kompilieren: `gcc -Wall -std=c99 -pthread ./statistics.c -o statistics`

Abgabefrist: 12. März 2019, 10h00