
SPiCAufgabe #8: SPITE

(12 Punkte, Abgabe siehe unten, keine Gruppen)

Schreiben Sie ein Programm `spite` (**SPiC Timed Execution**), welches ähnlich dem Programm `watch(1)` eine bestimmten Befehl, gegebenenfalls mit Parametern, immer wieder ausführt.

```
./spite <sec> <command> [...]
```

- Als ersten Parameter bekommt das Programm die Anzahl der Sekunden, welche es zwischen den Ausführungen warten soll (`alarm(2)`).
- Nach dem Ablauf der vorgegebenen Zeit soll das Kommando in einem neu erzeugten Prozess (`fork(2)`) mit korrekt übergebenen Argumenten ausgeführt werden (`execvp(3)`).
- Ist die Ausführung beendet (`wait(2)`) wird der Beendigungsstatus (`WIFEXITED`, `WEXITSTATUS`, `WIFSIGNALED`, `WTERMSIG`) ausgegeben.
- Anschließend wird auf das `ALARM`-Signal gewartet (`sigsuspend(2)`, `sigaction(2)`).
- Sollte das Programm nicht rechtzeitig beendet sein, wird die Ausführung für die verpassten Zeitpunkte ausgesetzt. Nach Beendigung soll ein Meldung ausgegeben werden, dass mindestens eine Ausführung verpasst wurde, und anschließend auf den Ablauf des Timers gewartet werden. (Beispiel: Das Programm soll alle 2 Sekunden gestartet werden, benötigt aber 3. Folglich wird das Programm zu den Zeitpunkten 2, 6, 10, ... gestartet – 4, 8, 12, ... werden verpasst.)
- Das Drücken von `CTRL-C` (`SIGINT`) soll `spite` nur dann beenden, wenn gerade kein Programm ausgeführt wird (`sigprocmask(2)`). Das Kind soll das Signal dennoch empfangen.

Hinweis

- Übersetzen Sie das Programm mit
`gcc -pedantic -Wall -Werror -std=c99 -D_XOPEN_SOURCE=600 -o spite spite.c`
- Zum Testen kann `spic-wait` oder `sleep(1)` verwendet werden.

Abgabezeitpunkt

T00, T01	Dienstag,	09.07.2013	18:00
T02, T03, T04	Mittwoch,	10.07.2013	18:00