

7. Übungsblatt

(Ausgabe: 8. Dezember 2016 — Abgabe bis: 15. Dezember 2016, 4:00)

Aufgabe 1: Minix Scheduler

(30 Punkte)

Modifizieren Sie den Minix-Scheduler so, dass Userprozesse mit gleicher Priorität, aber weniger CPU-Zeit zuerst verwendet werden. Lesen Sie hierzu Kapitel „2.6.10. Scheduling in MINIX 3“ aus dem Buch „Operating Systems Design & Implementation 3rd Edition“. Geben Sie nur ihre Änderungen von dieser Aufgabe als `git diff`¹ in einer Datei namens `scheduler.diff` ab.

Hinweis 1: Sie können Userprozesse leichter identifizieren indem Sie ihre maximale Priorität begrenzen.

Hinweis 2: Überlegen Sie sich, an welcher Stelle es einfacher ist die Reihenfolge der Prozesse zu verändern.

Aufgabe 2: Immer wieder Pointer

(10 Punkte)

Beantworten Sie folgende Fragen rund um Pointer und schreiben Sie ihre Antworten in eine Datei namens `abgabe.md`.

1. Beschreiben Sie was die folgende Codezeile macht. (1)

```
1 char string[] = "Foobar";
```

2. Beschreiben Sie auch diese Zeile. (1)

```
1 const char *string = "Foobar";
```

3. Welche Unterschiede bzw. Vor- und Nachteile haben die vorangegangenen Codezeilen? Wann würden Sie welche einsetzen? (2)

4. Ist folgende Codezeile valide? Was ist der Vor-/Nachteil? (2)

```
1 char string[5] = "Universität";
```

5. Beschreiben Sie den Unterschied der unteren Zeile zu 1. und 4.. (2)

```
1 const char string[20] = "Foobar";
```

6. Wie unterscheidet sich diese Initialisierung zu 5.? Ist sie valide und was wäre die Ausgabe wenn man `string` printed? (2)

```
1 const char string[] = {'F', 'o', 'o', 'b', 'a', 0x74};
```

¹Da Ihr Minix-Code ein `git`-Repository sein sollte (Übungsblatt 1), reicht es `git diff > scheduler.diff` in diesem Ordner auszuführen. Stellen Sie sicher, dass das Repository keine Änderungen von vorausgegangenen Übungen enthält und *denken Sie an ihren Namen*.

Aufgabe 3: Linked List

(20 Punkte)

Ihr Arbeitgeber hat Ihnen den Code (`list.c` und `list.h`) für eine rudimentäre verlinkte Liste gegeben. Es können momentan nur Integer an das Ende der Liste angehängt werden und diese auf der Standardausgabe ausgegeben werden.

Leider ist der Code nicht kommentiert und es besteht noch keine Möglichkeit Elemente geordnet einzufügen bzw. das oberste Element zu entfernen. Ändern Sie beides!

Implementieren Sie hierzu die Funktionen, `pushComp` und `pop` wie im Headerfile angegeben und *kommentieren Sie jede Funktion und wichtige Stelle* im Code.

Da ihr Arbeitgeber mehrere Programmieren angestellt hat und Ihnen auch ihre verdiente Entlohnung zukommen lassen möchte, schreiben Sie bitte Ihren Namen an den Anfang des Quellcodes.

Um ihre Arbeit zu kontrollieren, gibt das Programm eine vordefinierte Liste (welche nicht zu ändern ist) auf der Standardausgabe aus, sollte aber selbstverständlich auch mit anderen Eingaben zurecht kommen.

Sollten Sie Fehler im Code finden, korrigieren Sie diese bitte und erklären Sie das Problem in einem Kommentar.