



Betriebssysteme und Sicherheit, WS 2023/24

3. Aufgabenblatt – Scheduling

Geplante Bearbeitungszeit: 30.10.2023 – 03.11.2023

Aufgabe 3.1 Untersuchen Sie inwieweit sich die Schedulingverfahren *Round Robin*, *Virtual Round Robin* und *Shortest Remaining Time First* auf die Abarbeitungsreihenfolge folgender 3 Prozesse auswirken:

$$P_1 \text{ (CPU 6, IO 2)}, P_2 \text{ (CPU 3, IO 3)}, P_3 \text{ (CPU 2, IO 4)}$$

Die 3 Prozesse werden alle zum Zeitpunkt 0 in der angegebenen Reihenfolge bereit und laufen unendlich lange. Die Schedulingverfahren Round Robin und Virtual Round Robin nutzen eine Zeitscheibe von 3 Zeiteinheiten. Es gibt in dem System nur eine CPU, auf der Prozesse ausgeführt werden können. Entwerfen Sie entsprechende Abläufe der Prozesse gemäß dem gewählten Verfahren. Wie verhalten sich entsprechend die Bewertungskriterien *durchschnittliche Wartezeit* \bar{t}_w und *durchschnittliche Verweilzeit* \bar{t}_v der jeweiligen Prozesse und allgemein für das System.

Überprüfen Sie ihre Ergebnisse auf der Website: <https://ess.cs.uos.de/software/AnimOS/CPU-Scheduling/>.

Aufgabe 3.2 In einem Ein-Prozessor-Echtzeitsystem sind zwei periodische Tasks T_1 , T_2 einzuplanen mit folgenden Werten (p_i Periodenlänge, t_i Bearbeitungszeit, Periodenende = Zeitschranke):

$$p_1 = 5 \text{ s}, t_1 = 2 \text{ s} \quad p_2 = 3 \text{ s}, t_2 = 1 \text{ s}$$

Zwischen den Tasks bestehen keine Abhängigkeiten; sie sind an beliebiger Stelle unterbrechbar. Die Tasks sollen mit einem Schedulingverfahren mit statischen Prioritäten eingeplant werden.

- Untersuchen Sie, inwieweit dies möglich ist.
- Nach einiger Zeit ist eine weitere Task T_3 einzuplanen mit $p_3 = 5 \text{ s}$, $t_3 = 1 \text{ s}$. Diskutieren Sie die entstandene Situation.
- Nachträglich stellt sich heraus, dass für T_1 ein Prozessorbedarf von 2,1 s erforderlich ist. Welche Konsequenzen hat dies für Zulassung und Einplanung der drei Tasks?

Aufgabe 3.3 In einem Ein-Prozessor-Echtzeitsystem sind drei periodische Tasks T_1 , T_2 , T_3 einzuplanen mit folgenden Werten (p_i Periodenlänge, t_i Bearbeitungszeit, Periodenende = Zeitschranke):

$$p_1 = 6 \text{ s}, t_1 = 2 \text{ s} \quad p_2 = 3 \text{ s}, t_2 = 1 \text{ s} \quad p_3 = 4 \text{ s}, t_3 = 1 \text{ s}$$

- Als Echtzeitschedulingverfahren wird EDF verwendet. Beurteilen Sie, ob die Taskmenge einplanbar ist und wenn ja, zeichnen Sie einen entsprechenden Ablaufplan.
- Die Task T_3 muss länger ausgeführt werden. Berechnen Sie die maximale mögliche Bearbeitungszeit, die das System noch verkraften kann.

Klausuraufgabe I

Ein um die Erde kreisender Satellit bestimmt seine Position mittels dreier verschiedener Lagesensoren (Sternensensor A, Magnetometer B, Gyroskop C). Zur genauen Lageregelung müssen diese periodisch ausgewertet werden, um die Antriebe entsprechend zu steuern. Die notwendigen Perioden p und Ausführungszeiten e der Auswertungsalgorithmen sind in nebenstehender Tabelle zusammengefasst. Die relative Deadline d ist gleich der Periodenlänge ($d = p$). Alle Tasks sind voneinander unabhängig und laufende Jobs zu beliebiger Zeit unterbrechbar.

	A	B	C
p	9	6	3
e	3	2	1

- Ist die Lageregelung auf einem Prozessor mittels dynamischer Prioritäten möglich, falls der Scheduling-Overhead vernachlässigbar ist? Wenn nein, begründen Sie. Falls ja, schlagen Sie einen Algorithmus für die Einplanung vor.
- Während der Entwicklung der Software wird festgestellt, dass der Scheduling-Overhead für das Einplanen eines neuen Jobs in einem System mit dynamischen Prioritäten 0,2 Zeiteinheiten beträgt. Ist die Taskmenge weiterhin auf einem Prozessor einplanbar?
- Existiert eine statische Prioritätszuordnung mit welcher die Taskmenge auf einem Prozessor eingeplant werden kann? Falls ja, geben Sie diese Zuordnung an! Ansonsten begründen Sie Ihre Aussage!
HINWEIS: Bei statischen Prioritäten ist der Scheduling-Overhead vernachlässigbar.

Als Lösung schlägt der System-Designer vor, den Sternensensor (A) durch einen Sonnensensor (A') zu ersetzen. Dessen Auswertungsalgorithmus hat eine Periode von 4 und eine Ausführungszeit von 1 Zeiteinheiten. Das System besteht damit jetzt aus den nebenstehenden Tasks.

	A'	B	C
p	4	6	3
e	1	2	1

- Ist dieses System mittels statischer Prioritäten einplanbar? Falls ja, nennen Sie eine Prioritätszuteilung! Ansonsten begründen Sie Ihre Aussage!
HINWEIS: Bei statischen Prioritäten ist der Scheduling-Overhead vernachlässigbar.
- Wäre das System mittels dynamischer Prioritäten einplanbar, wenn hier weiterhin von einem Overhead für den Jobwechsel von 0,2 Zeiteinheiten ausgegangen wird?

Klausuraufgabe II

In einem Echtzeitsystem ist eine Menge von drei periodischen Tasks so einzuplanen, dass deren Jobs in jeder Periode erfolgreich beendet werden. Das Periodenende entspricht der Deadline. Die Parameter der Tasks beschreiben jeweils die Periode p und die konstante Bearbeitungszeit e der Jobs.

$$\begin{aligned} T_1: p_1 &= 3, \quad e_1 = 1 \\ T_2: p_2 &= 12, \quad e_2 = 2 \\ T_3: p_3 &= 4, \quad e_3 = 1 \end{aligned}$$

Alle Tasks starten zum Zeitpunkt $t = 0$ und können an beliebiger Stelle unterbrochen werden. Der Scheduling-Overhead werde vernachlässigt und es stehe genau ein Prozessor zur Verfügung.

- Zunächst werden den Tasks statisch Prioritäten zugeteilt, so dass T_1 die höchste, T_2 eine mittlere und T_3 die niedrigste Priorität erhält. Weisen Sie nach, dass die gegebene Task-Menge unter den genannten Bedingungen nicht einplanbar ist!
- Ist eine Einplanung dieser Task-Menge mit statischen Prioritäten überhaupt möglich? Wenn ja, geben Sie eine entsprechende Prioritätszuteilung an!
- Die Task-Menge soll mit dynamischen Prioritäten eingeplant werden. Schlagen Sie ein passendes Einplanungsverfahren vor und geben Sie an, welcher Task im gewählten Verfahren zum Zeitpunkt $t = 3,5$ die höchste Priorität aufweist.
- Weiterhin soll die Ausführungszeit e_3 der Task T_3 erhöht werden. Bis zu welchem Maximalwert von e_3 ist eine Einplanung mit dynamischen Prioritäten möglich?

Klausuraufgabe III

Ein Einprozessor-Betriebssystem verwaltet drei Prozesse P_1 , P_2 und P_3 . Die Prozesse treffen in dieser Reihenfolge im System ein und sind alle zum Zeitpunkt $t = 0$ rechenbereit. Die Prozesse wiederholen sich unendlich lange. Nach jedem CPU-Stoß führt der Prozess einen E/A-Stoß von 20 ms durch. Die CPU-Stöße (in ms) der Prozesse sind in der folgenden Tabelle angegeben:

Prozesse	P_1	P_2	P_3
CPU-Stöße	40	80	90

Zeichnen Sie in das folgende Gantt-Diagramm ein, wie die drei Prozesse P_1 , P_2 und P_3 bearbeitet werden würden, wenn das Scheduling nach der „Virtual Round Robin“-Strategie vorgenommen wird. Die gewählte Zeitscheibe beträgt 30 ms. Hinweis: Die ersten beiden Zeiteinheiten sind bereits fertig ausgefüllt.

