

# Scheduling 2 (Round Robin) - Handout

\*\*\*\*\* Nguyen

2. November 2020

## 1 Konzept

Es wird ein Quantum (eine Art Zeitslot) festgelegt, welcher üblicherweise zwischen 10ms und 100ms liegt. Wenn ein Prozess länger braucht als ein Quantum, wird es wieder hinten an die Warteschlange geschickt. Wenn ein Prozess während eines Quantums fertig wird, bekommt der nächste ein vollständiges Quantum.

## 2 Beispiel

Es sei ein Quantum von 100ms festgelegt.

Prozess	Ankunftszeit	Prozessdauer
P0	0ms	150ms
P1	30ms	250ms
P2	120ms	50ms
P3	130ms	170ms

Tabelle 1: Beispiel Prozesse

Zeitstrahl						Verbleibende Zeit			
Zeit	Slot 1	Slot 2	Slot 3	Slot 4	Event	P0	P1	P2	P3
0ms	<b>P0</b>				P0 join	<b>100</b>			
30ms	P0	<b>P1</b>			P1 join	<b>70</b>	250		
100ms	<b>P1</b>	<b>P0</b>			<b>Rotate</b>	<b>50</b>	250		
120ms	P1	P0	<b>P2</b>		P2 join		<b>230</b>	50	
130ms	P1	P0	P2	<b>P3</b>	P3 join		<b>220</b>		170
200ms	<b>P0</b>	<b>P2</b>	<b>P3</b>	<b>P1</b>	<b>Rotate</b>	50	<b>150</b>		
250ms	P2	P3	P1		P0 finish	<b>0</b>		50	
300ms	P3	P1			P2 finish			<b>0</b>	170
400ms	<b>P1</b>	<b>P3</b>			<b>Rotate</b>		150		<b>70</b>
500ms	<b>P3</b>	<b>P1</b>			<b>Rotate</b>		<b>50</b>		70
570ms	P1				P3 finish		50		<b>0</b>
620ms					P1 finish		<b>0</b>		

Tabelle 2: Ablauf im Round Robin Scheduling

### 3 Fazit

- Alle Prozesse werden hier mit der gleichen Dringlichkeit bearbeitet (kann aber auch keine Prioritäten setzen)
- Es wird sicher gegangen, dass kein Prozess verhungert
- Niedrige durchschnittliche Verweilzeit für alle Prozesse
- Algorithmus ist leicht zu implementieren
- Wichtig: Das Quantum muss gut gewählt sein
  - Wenn das Quantum zu groß ist, ähnelt es einem First Come First Serve (FCFS/FCFO)
  - Wenn das Quantum zu klein ist, ähnelt es einem Shortest Job First (SJF) und der Kontextwechsel Aufwand wird größer

### Literatur

- [1] Wolfram Burgard. *Systeme I: Betriebssysteme, Kapitel 7 Scheduling*. (letzter Zugriff: 2020-11-01). 2016. URL: <http://ais.informatik.uni-freiburg.de/teaching/ws16/systems1/slides/kap07-scheduling.pdf>.
- [2] Prof. Dr. Margarita Esponda. *Scheduling*. (letzter Zugriff: 2020-11-01). 2012. URL: [http://www.inf.fu-berlin.de/lehre/WS11/OS/slides/OS\\_V7\\_Scheduling\\_Teil\\_1.pdf](http://www.inf.fu-berlin.de/lehre/WS11/OS/slides/OS_V7_Scheduling_Teil_1.pdf).
- [3] R. Bär und G. Bischofberger und E. Dehler und N. Hammer und B. Schiemann und T. Wolf. *Informatik und Informationstechnik*. EUROPA-LEHRMITTEL, 2017. URL: [http://www.inf.fu-berlin.de/lehre/WS11/OS/slides/OS\\_V7\\_Scheduling\\_Teil\\_1.pdf](http://www.inf.fu-berlin.de/lehre/WS11/OS/slides/OS_V7_Scheduling_Teil_1.pdf).