

Übung zum Rundlaufverfahren und Semaphore

- ① Man betrachte die vier zyklisch arbeitenden Fäden *A*, *B*, *C* und *D*. Die folgende Tabelle enthält die Fäden mit ihren jeweils benötigten CPU-Zeiten und E/A-Zeiten, wobei ein CPU-Stoß und ein E/A-Stoß vollständig auszuführen ist:

	CPU	E/A
A	3	6
B	6	2
C	2	6
D	5	4

Als Zeiteinheit sei hier die Einheit **ms** zu nutzen. Zum Zeitschritt **0** stehen die Fäden in der Reihenfolge *A*, *B*, *C*, *D* auf der Bereit-Liste. Vor einem E/A-Stoß muss ein CPU-Stoß erfolgen. Der E/A-Stoß muss unmittelbar nach einem CPU-Stoß ausgeführt werden. Es sei anzunehmen, dass beliebig viele E/A-Stöße gleichzeitig ausgeführt werden können. In jedem Zeitschritt kann ausschließlich jeweils nur ein Faden einen CPU-Stoß durchführen. Diese Annahme entspricht der Realität, wenn ein System nur einen Kern besitzt.

Im folgenden mittels **Rundlaufverfahren** (engl. **Round-Robin**, kurz **RR**) das Scheduling für die oben gegebenen vier Fäden für die ersten **22 ms** zu berechnen. Die Zeitscheibengänge beträgt $\tau = 3 \text{ ms}$ Dazu sei das gegebene Gantt-Diagramm für die Zeitschritte 3 bis 22 zu vervollständigen. Die Zeitschritte 0 bis 2 sind bereits vorausgefüllt:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
A	C	C	C																				
B																							
C																							
D																							

Ein CPU-Stoß soll durch **C** gekennzeichnet werden, während ein E/A-Stoß mit **E** ausgedrückt werden soll. Ein Feld bleibt leer, wenn der Prozess auf der Bereit-Liste steht.

- ② **Why did the multithread chicken crossed the road?**

Gegeben seien drei Fäden die jeweils eine von drei Funktionen ausführen (siehe unten) sowie drei Semaphore **S1**, **S2** und **S3**. Die Semaphore sollen jeweils mit **0** initialisiert werden. Der Programmtext der drei Funktionen sei derart mit den Anweisungen **P(&S1)**, **P(&S2)** und **P(&S3)** bzw. **V(&S1)**, **V(&S2)** und **V(&S3)** zu vervollständigen, so dass bei gleichzeitiger Ausführung der drei Fäden die Ausgabe **to get to the other side** entsteht.

Faden 1	Faden 2	Faden 3
<p style="text-align: center;">S1 = 0;</p> <pre> chicken1() { printf("to "); <input type="text"/> ; <input type="text"/> ; printf("to "); <input type="text"/> ; <input type="text"/> ; printf("other "); <input type="text"/> ; } </pre>	<p style="text-align: center;">S2 = 0;</p> <pre> chicken2() { <input type="text"/> ; printf("get "); <input type="text"/> ; } </pre>	<p style="text-align: center;">S3 = 0;</p> <pre> chicken3() { <input type="text"/> ; printf("the "); <input type="text"/> ; <input type="text"/> ; printf("side "); } </pre>