

Teil a

FCFS:

Warteschlange: 1,2,3,4,5

mittlere Wartezeit: $4 * 5 + 3 * 7 + 2 * 3 + 1 * 8 = 55 \Rightarrow 11$

LCFS:

Warteschlange: 5,4,3,2,1

mittlere Wartezeit: $4 * 2 + 3 * 8 + 2 * 3 + 1 * 7 = 45 \Rightarrow 9$

Teil b

RR (ZQ=5):

CPU	Prozess	wartend
0-5	1	2,3,4,5
5-10	2	3,4,5
10-13	3	4,5,2
13-18	4	5,2
18-20	5	2,4
20-22	2	4
22-25	4	-

mittlere Wartezeit: $4 * 5 + 3 * 5 + 3 * 3 + 2 * 5 + 2 * 2 + 1 * 2 = 60 \Rightarrow 12$

Das Programm enthaelt in Zeile 03 eine Endlosschleife.
Daher gilt:
Mutual Exclusion ist erfuehlt.
Progress Requirement und Bounded Waiting sind nicht erfuehlt.

Teil a

A ist unsicher:

	ALLOC			NEED		
0	0	2	2	7	3	1
1	3	1	1	4	1	1
2	3	2	1	6	6	0
3	1	1	0	2	1	0
TOTAL	7	6	4			
FREE	3	2	0			

WORK			
3	2	0	(P_3)
4	3	0	

B ist sicher:

	ALLOC			NEED		
0	1	2	1	6	3	2
1	3	1	2	4	1	0
2	2	2	0	7	6	1
3	1	1	0	2	1	0
TOTAL	7	6	3			
FREE	3	2	1			

WORK			
3	2	1	(P_3)
4	3	1	(P_1)
7	4	3	(P_0)
8	6	4	(P_2)

Teil b

1.Forderung: nicht zulaessig. Die Anforderung geht ueber MAX(3) hinaus.

2.Forderung:

WORK			
5	1	2	Request(2)=(1,1,0)
4	0	2	

⇒ nicht zulaessig

3.Forderung:

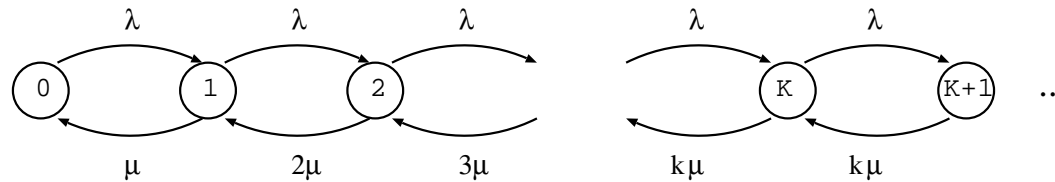
WORK			
5	1	2	Request(0)=(0,1,1)
5	0	1	

⇒ nicht zulaessig

Teil a

$M|M|K$ System mit $K = 4$

Teil b



Teil c

$$\begin{aligned}
 \lambda \cdot \Pi_0 &= \mu \cdot \Pi_1 \\
 (\lambda + \mu) \Pi_1 &= \lambda \Pi_0 + 2\mu \Pi_2 \\
 (\lambda + 2\mu) \Pi_2 &= \lambda \Pi_1 + 3\mu \Pi_3 \\
 (\lambda + 3\mu) \Pi_3 &= \lambda \Pi_2 + 4\mu \Pi_4 \\
 (\lambda + 4\mu) \Pi_4 &= \lambda \Pi_3 + 4\mu \Pi_5 \\
 &\vdots
 \end{aligned}$$

Teil a

```
init (S,1);  
  
hoch() {  
    wait(S);  
  
    if(A < 20) ++A;  
    else if(B < 20) ++B;  
    else if(C < 40) ++C;  
  
    signal(S);  
}
```

Teil b

```
runterA() {  
    wait(S);  
    if(A > 0) --A;  
    signal(S);  
}  
  
runterB() {  
    wait(S);  
    if(B > 0) --B;  
    signal(S);  
}  
  
runterC() {  
    wait(S);  
    if(C > 0) --C;  
    signal(S);  
}
```

Teil a $320+58+120+60+150+110=818$

Teil b

kleinste Adresse: 310

größte Adresse: 1639

Teil c

780	(3,30)
1436	(0,116)
580	(1,18)
984	segfault
420	(2,110)

CLIMB

2 4 1 3 5 5 2 3 3 3 1 4 0 3 3 2 1 4 2 2 4

=====

2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 2 2 2 2 2 2
 4 4 4 4 5 5 5 3 2 2 2 2 2 2 3 3 3 3 3 4
 1 3 5 4 4 3 5 5 1 4 0 0 0 0 1 4 4 4 3

* * * * * = 8 PF

2 4 1 3 5 5 2 3 3 3 1 4 0 3 3 2 1 4 2 2 4

=====

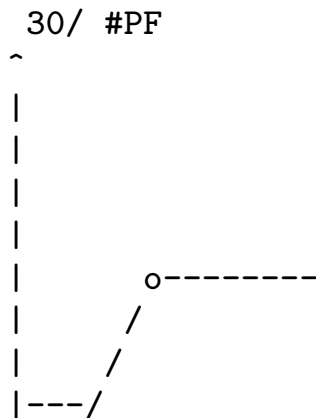
2 2 2 2 2 2 2 2 2 2 2 2 2 3 2 2 2 2 2 4
 4 4 4 4 4 4 4 4 3 3 4 4 3 2 3 3 4 4 4 2
 1 1 1 5 5 5 3 4 4 3 3 4 4 4 4 3 3 3 3
 3 5 1 1 3 5 5 1 1 0 0 0 0 1 1 1 1 1

* * * * * = 5 PF

a) Lifetime-Funktion

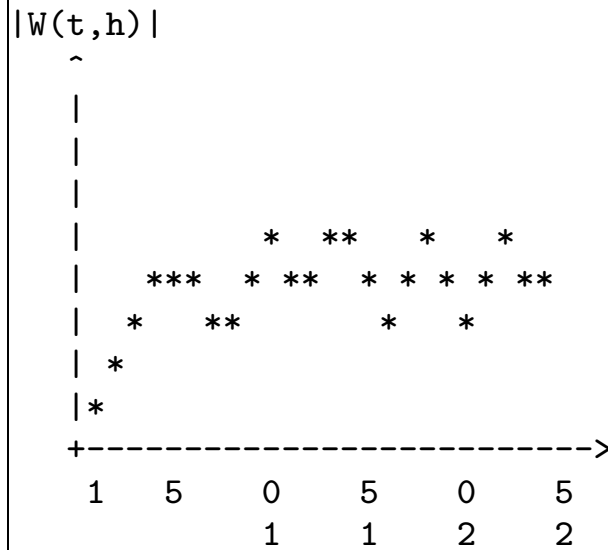
m	#PF
1	30
2	27
3	20
4	15
5	13
6	6
.	.
.	.
10	6

Graph sigmoidaler Verlauf (x-Achse m



+----->
 6 m
b) optimale Einstellung bei 6

b) Graph



c) Working Sets

- t=12: {1,2,5,6}
- t=13: {1,2,3,5,6}
- t=21: {1,2,3,5}
- t=24: {1,2,3,4}