Operating Systems

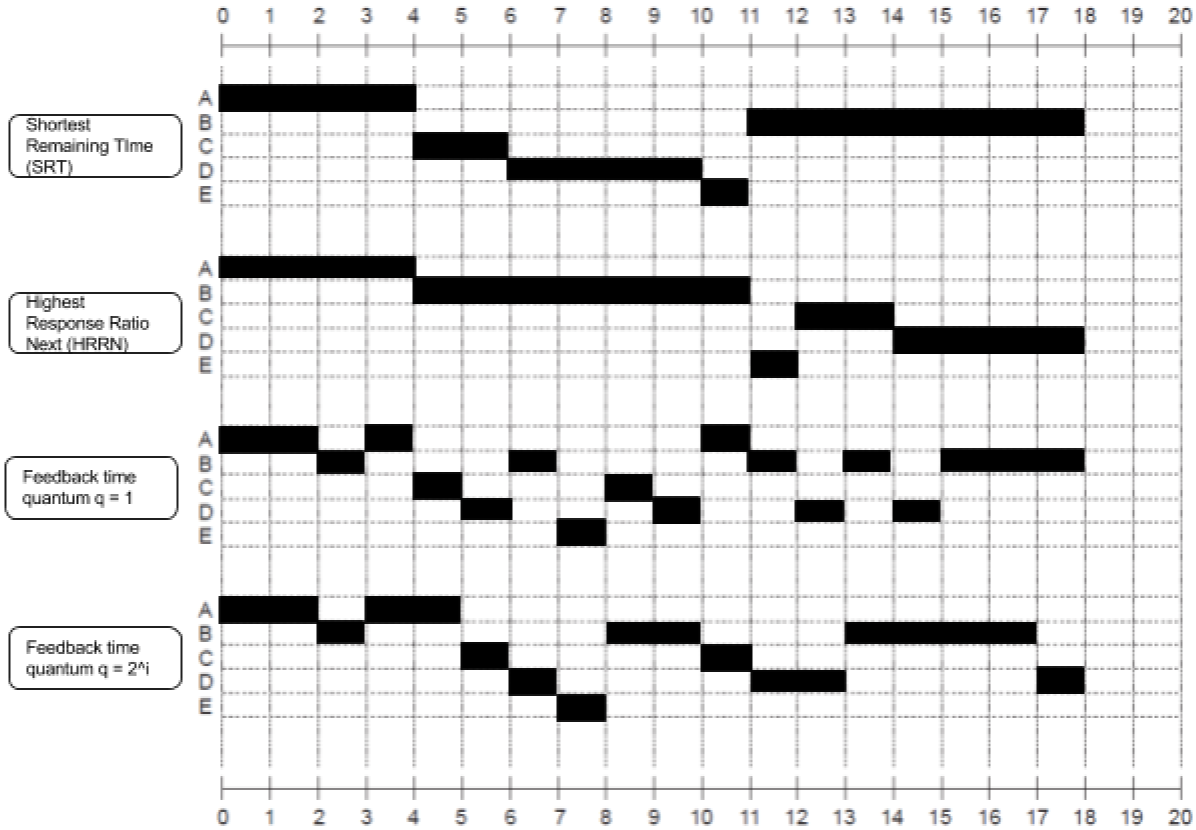
TEAM 6
Leonardo Nahra
Levindo Neto
Li-Wei Chen

Assignment 3

University of Stuttgart
04.12.2017

Assignment 3 - Answers

1.a)



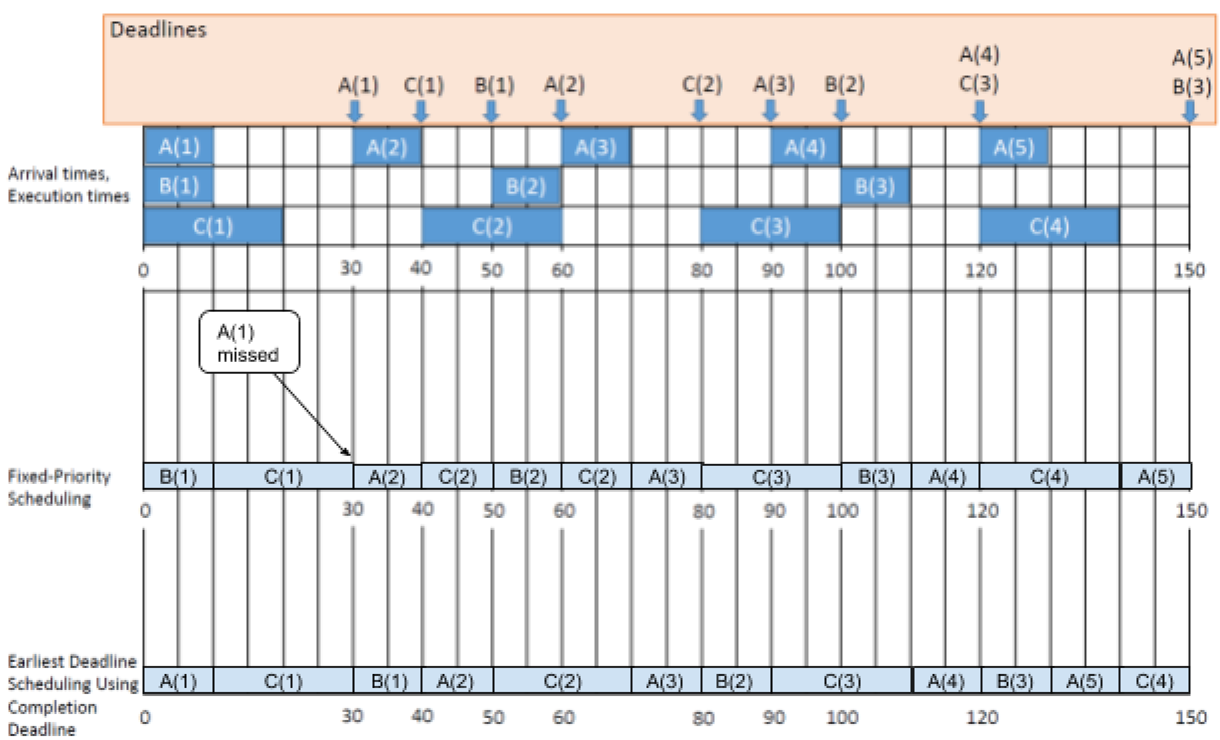
1.b)

	Process	A	B	C	D	E	Mean
	Arrival Time	0	2	4	5	7	3.6
	Service Time (Ts)	4	7	2	4	1	3.6
SRT	Response Time	1	10	1	2	4	3.6
	Turnaround Time (Tr)	4	16	2	5	4	6.2
	Relative Delay (Tr/Ts)	1	2.29	1	1.25	4	1.91
HRRN	Response Time	1	3	9	10	5	5.6

	Turnaround Time (Tr)	4	9	10	13	5	8.2
	Relative Delay (Tr/Ts)	1	1.29	5	3.25	5	3.11
Feedback Time Quantum q = 1	Response Time	1	1	1	1	1	1
	Turnaround Time (Tr)	11	16	5	10	1	8.6
	Relative Delay (Tr/Ts)	2.72	2.29	2.5	2.6	1	2.22
Feedback Time Quantum q = n^i	Response Time	1	1	2	2	1	1.4
	Turnaround Time (Tr)	5	15	7	13	1	8.2
	Relative Delay (Tr/Ts)	1.25	2.14	3.5	3.25	1	2.23

The feedback time quantum q=1 has the least average response time.
SRT has the least average relative delay.

2.a)



2.b)

Fixed-priority scheduling causes greater overhead for the dispatcher because it must switch between different process more often if one process with higher priority arrives.

2.c)

RMS is feasible if and only if the following equation holds:

C1/T1+C2/T2+...Cn/Tn<=n*(2^(1/n)-1)

For our question: 20/60 + 40/100 = 0.733 <= 0.828 (n=2), therefore these two tasks can be successfully scheduled by rate monotonic scheduling (RMS).

2.d)

20/60 + 40/100 + 20/120 = 0.9 <= 0.799 (n=3)

The equation does not hold. Hence, it is not feasible for scheduling these three tasks with Rate Monotonic Scheduling.

2.e)

- For the first deadline of three tasks to be met. The following equation holds: C1/T1+C2/T2+...Cn/Tn<1. In this case, the first deadline of three tasks will be met.
- Deadline of future repetitions of the same task will not be met since it ought to wait and may not be finished before the deadline.
- From the result of 2.d), these three tasks cannot be scheduled by RMS.

3.a)

Time	CPU 1	CPU 2	CPU 3	CPU 4
0	A0	A1	IDLE	IDLE
1	A0	A1	B0	B1
2	A0	A1	B0	B1
3	C0	C1	B0	B1
4	C0	C1	B0	B1
5	C0	C1	B0	B1
6	C2	C3	D0	D1
7	C2	C3	D0	D1
8	C2	C3	D0	E0
9	C2	C3	D0	E0
10	C2	C3	E1	E0

11	E2	E3	E1	E0
12	E2	E3	E1	IDLE
13	E2	E3	E1	IDLE
14	E2	E3	IDLE	IDLE

3.b)

Time	CPU 1	CPU 2	CPU 3	CPU 4
0	A0	A1	IDLE	IDLE
1	A0	A1	B0	B1
2	A0	A1	B0	B1
3	D0	D1	B0	B1
4	D0	D1	B0	B1
5	D0	IDLE	B0	B1
6	D0	IDLE	IDLE	IDLE
7	C0	C1	C2	C3
8	C0	C1	C2	C3
9	C0	C1	C2	C3
10	IDLE	IDLE	C2	C3
11	IDLE	IDLE	C2	C3
12	E0	E1	E2	E3
13	E0	E1	E2	E3
14	E0	E1	E2	E3
15	E0	E1	E2	E3

3.c)

I. An execution scenario in which the total processing time of both processes reaches its maximum value could be like following:

Execution time (1 unit time)	CPU 1	CPU 2
0-8	A0	B0
9-27	A0	B1
28-31	A0	A1

II. An execution scenario in which the total processing time of both processes reaches its minimum value could be like following:

Execution time (1 unit time)	CPU 1	CPU 2
0-3	A0	A1
4-12	B0	B1
13-22	A0	A1

Execution time (1 unit time)	CPU 1	CPU 2
0-8	B0	B1
9-19	A1	B1
20-21	A1	A0
22	A1	

III. Space-sharing scheduling guarantees the minimum processing time. The minimum processing time occurs when two B (or A) threads process simultaneously. And that would occur when we are using the space-sharing scheduling scheme.