

CS3103 Operating Systems

2018-2019 Semester B

Midterm

Paper B

Note: The total points of this paper are 60

Part I Multiple choice questions (1 points each, 10 points in total)

- 1) C
- 2) D
- 3) A
- 4) A
- 5) A
- 6) A
- 7) B
- 8) D
- 9) C
- 10) C

Part II Short answer questions (3 points each, 15 points in total. Please be brief. Your answer shall be no more than 3 sentences)

1)

Yes. Mutual exclusion and interrupt are two different concepts. When a thread is in a critical section and if mutual exclusion is well designed, it can only guarantee that no other thread can access the same critical section, which, however, cannot guarantee that the current thread cannot be interrupted. For example, the current thread can be switched out due to timeout.

2)

Any three of the following events.

- Interrupt, e.g., timeout
- I/O event occurs
- Newly admitted process
- A process is activated and the state is changed from ready/suspend to ready.

3)

Interrupt, System call, exception/trap

4)

P1 will potentially get more CPU time than P2 in (b). In (b), OS is aware of two kernel threads from P1 and only one kernel thread from P2. Since there are multiple processors, each thread may be assigned to one processor. Thus, P1 can potentially get more CPU time than P2.

Or you can say that in (a), OS is not aware of two threads of P1 and only aware of one process. Since OS is only aware of one process from each of them, P1 and P2 will get the same amount of CPU time.

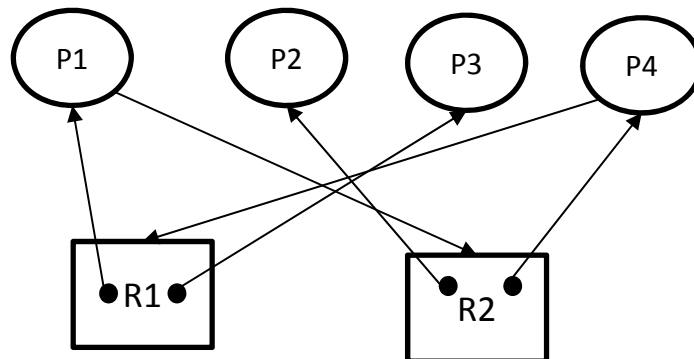
5)

No. For a sequential program, due to the dependence of instructions, even the program is multi-threaded, the execution will be sequential, it will not be executed faster. On the other hand, due to the overhead of thread switch, it could be slower than single-threaded solution.

Part III Problem solving questions (35 points in total)

1.

a)



b)

No. P2 can finish and release one unit of R2

P3 can finish and release one unit of R1

P1 can obtain one unit of R2, finish, and release one unit of R1 and one unit of R2

P4 can obtain one unit of R1, finish, and release one unit of R1 and one unit of R2.

(Note that there are several feasible sequences to finish all processes)

2.

a)

Yes. C-A is

available resource is (1, 1, 1, 1)

	A	B	C	D
P0	1	1	1	0
P1	0	2	6	3
P2	0	2	2	2
P3	0	2	1	1

P0 can finish and the available resource vector is (1, 3, 1, 3)

P3 can finish and the available resource vector is (1, 7, 3, 3)

P2 can finish and the available resource vector is (3, 7, 6, 7)

P1 can finish and the available resource vector is (4, 10, 6, 7)

Thus, it is a safe state.

b)

No. If the request from P1 is granted, the new C-A is

	A	B	C	D
P0	1	1	1	0
P1	0	1	5	3
P2	0	2	2	2
P3	0	2	1	1

Available resource vector is (1, 0, 0, 1), which cannot let any process finish. Thus, it will lead to an unsafe state. P1's request will not be granted.

3.

Process	Statement number	x
P1	1	9
P2	1	9
P1	2	8
P1	3	9
P2	2	8
P1	4	8
P2	3	9
P1	5	9
P2	4	9

4.

(a)

The consumer will be blocked when executing “Receive(mayconsume, cmsg)” as no message has been sent to mayconsume mailbox.

(b)

As initially there are some messages in mayproduce mailbox (the number of messages is the value of capacity), the producer can produce, then send a message to mayconsume mailbox. The first consumer will be waked up.