Andre J Plath

Professor Yanwei Wu

CS365 Operating Systems and Networking

February 26, 2018

CS372 Operating System
HW4

7.12 Assume a multithreaded application uses only reader — writer locks for synchronization. Applying the four necessary conditions for deadlock, is deadlock still possible if multiple reader—writer locks are used?

Yes, Mutual exclusion is maintained, if there is a writer they cannot be shared. Hold-and-wait is possible, as a thread can hold one reader—writer lock while waiting to acquire another. No preemption is upheld because you cannot take a lock away. With all threads circular wait is possible.

7.19 Consider the version of the dining-philosophers problem in which the chopsticks are placed at the center of the table and any two of them can be used by a philosopher. Assume that requests for chopsticks are made one at a time. Describe a simple rule for determining whether a particular request can be satisfied without causing deadlock given the current allocation of chopsticks to philosophers.

When a philosopher makes a request for the first chopstick, do not grant the request if there is no other philosopher with two chopsticks and if there is only one chopstick remaining.

7.22 Consider the following snapshot of a system:

	Allocation	Max
	A B C D	*A B C D*
P_0	3 0 1 4	5 1 1 7
P_1	2 2 1 0	3 2 1 1
P_2	3 1 2 1	3 3 2 1
P_3	0 5 1 0	4 6 1 2
P_4	4 2 1 2	6 3 2 5

Using the banker's algorithm, determine whether or not each of the following states is unsafe. If the state is safe, illustrate the order in which the processes may complete. Otherwise, illustrate why the state is unsafe.

a) **Available** = (0, 3, 0, 1)

Processes P2, P1, and P3 are able to finish. However, P0 and P4 are unable to finish. Not Safe.

b) **Available** = (1, 0, 0, 2)

All process are able to finish. <P1, P2, P3, P4, P0>. Safe

7.23 Consider the following snapshot of a system:

	Allocation	Max	Available
	A B C D	*A B C D*	*A B C D*
P_0	2 0 0 1	4 2 1 2	3 3 2 1
P_1	3 1 2 1	5 2 5 2	
P_2	2 1 0 3	2 3 1 6	
P_3	1 3 1 2	1 4 2 4	
P_4	1 4 3 2	3 6 6 5	

.

. Answer the following questions using the banker's algorithm:

a) Illustrate that the system is in a safe state by demonstrating an order in which the processes may complete.

The processes finish as follows <P0, P3, P4, P1, P2>.

b) If a request from process P_1 arrives for (1, 1, 0, 0), can the request be granted immediately?

No, it cannot be granted immediately.

c) If a request from process P_4 arrives for (0, 0, 2, 0), can the request be granted immediately?

No, it cannot be granted immediately.