7.8 The Linux kernel has a policy that a process cannot hold a spinlock while attempting to acquire a semaphore. Explain why this policy is in place.

1. The process may take no response because of long waiting
2. risk of deadlock

8.20 In a real computer system, neither the resources available nor the demands of processes for resources are consistent over long periods (months). Resources break or are replaced, new processes come and go, and new resources are bought and added to the system. If deadlock is controlled by the banker's algorithm, which of the following changes can be made safely (without introducing the possibility of deadlock), and under what circumstances?

(a) Increase Available (new resources added).

Safety. More resources indicates processes don't have to wait allocated resources.

(b) Decrease Available (resource permanently removed from system).

Not safety.

(c) Increase Max for one process (the process needs or wants more resources than allowed)

Not safety. Increase Max for one process indicates a process can require more resouces.

(d) Decrease Max for one process (the process decides that it does not need that many resources).

Safety.

(e) Increase the number of processes.

Not safety. More process have higher chance to cause circular waiting.

(f) decrease the number of processes.

Safety.

8.27 Consider the following snapshot of a system. Use 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.

PID	**Allocation**	**Max**	**Need**
	ABCD	ABCD	ABCD
P0	1 2 0 2	4 3 1 6	3 1 1 4
P1	0 1 1 2	2 4 2 4	2 3 1 2
P2	1 2 4 0	3 6 5 1	2 4 1 1
P3	1 2 0 1	2 6 2 3	1 4 2 2
P4	1 0 0 1	3 1 1 2	2 1 1 1

(a) Available=(2,2,2,3)

safe state
p4: 3,2,2,4
P0: 4,4,2,6
P3: 5,6,2,7
P2: 6,8,6,7
P1: 6,9,7,9

(b) Available=(4,4,1,1)

safe state
p4: 5,4,1,2
p2: 5,5,2,4
p1: 5,6,3,6
p3: 6,8,3,7
p0: 7,10,3,9

8.30 A single-lane bridge connects the two Vermont villages of North Tunbridge and South Tunbridge. Farmers in the two villages use this bridge to deliver their produce to the neighbor town. The bridge can become deadlocked if a northbound and a southbound farmer get on the bridge at the same time. (Vermont farmers are stubborn and are unable to back up.) Using semaphores and/or mutex locks, design an algorithm in pseudocode that prevents deadlock. Initially, do not be concerned about starvation (the situation in which northbound farmers prevent southbound farmers from using the bridge, or vice versa).

```
int mutex = 1

void enter_bridge () {
    wait(mutex);
    enter();
    signal(mutex);
}
```

9.15 Compare the memory organization schemes of contiguous memory allocation and paging with respect to the following issues:

(a) external fragmentation

(b) internal fragmentation

(c) ability to share code across processes

contiguous memory:
It will suffer from external fragmentation only if the memory partition is fixed, otherwise it will suffer from internal fragmentation.
It don't support code sharing.

paging:
It only suffer from internal fragmentation since the page size is fixed.
It support code sharing by mapping logical memory space to same physicla frames in page table.

9.24 Consider a computer system with a 32-bit logical address and 8-KB page size. The system supports up to 1 GB of physical memory. How many entries are there in each of the following?

(a) A conventional, single-level page table

8KB = 2^13 bit
entries count = 2^(32 - 13) = 2^19

(b) An inverted page table

1GB = 2^30 bit
entries count = 2^(30 - 13) = 2^17