Operating Systems

Homework #5
Due: 2007/6/5

1. Consider a system consisting of *m* resources of the same type being shared by *n* processes. Resources can be requested and released by processes only one at a time. Show that the system is deadlock free if the following two conditions hold:

   a. The maximum need of each process is between 1 and *m* resources.
   b. The sum of all maximum needs is less than *m*+*n*.

2. Consider the following snapshot of a system:

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

   Answer the following questions using the banker's algorithm:

   a. What is the content of the matrix *Need*?
   b. Is the system in a safe state?
   c. If a request from process P_1 arrives for (0,4,2,0), can the request be granted immediately?

3. A single-lane bridge connects the two Vermont villages of North Tunbridge and South Tunbirdge. Farmers in the two villages use this bridge to deliver their produce to the neighboring town. The bridge can become deadlocked if both 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, design an algorithm 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).