Worksheet 7.1: Deadlocks, Problem Solving

Q1. Given the following snapshot of a system, answer the following questions using the **avoidance** version of the **Banker's Algorithm**:

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

1. Show that the system is in a safe state by finding a safe sequence. Show your work.

	Work
	1 2 0
P_3	3 2 1
P_1	3 2 4
P_4	4 3 5
P_0	7 4 6
P_2	7 4 7

2. If a request from Process P_0 arrives for (0, 1, 0), can this request be granted immediately? Show your work.
   Allocation of P_0 becomes 3 2 1
   Need of P_0 becomes 3 1 5
   Avail becomes 1 1 0

	Work
	1 1 0
P_3	3 1 1
P_1	3 1 4
P_4	4 2 5
P_0	7 4 6
P_2	7 4 7

Since the request will put the system in a safe state, the request will be granted.

Q2. Given the following snapshot of a system, answer the following questions using the **<u>detection</u>** version of the **Banker's Algorithm**:

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

1. Show that the system is in a safe state by finding a safe sequence. When searching for a process whose request can be satisfied, check the processes in the given order, as we were doing in class. Show your work.

	Avail		
	2	1	3
P_0	2	3	5
P_4	5	3	5
P_2	6	4	5
P_1	8	5	6
P_3	9	7	7

2. How many safe sequences are there for the above snapshot? Briefly justify your answer.

   Only 1. At each step in the construction of the safe sequence, there is only one choice.

3. If P_1 requests an additional instance of each of A and C, that is, the request vector for P_1 becomes (2, 4, 6), will this request result in a deadlock? If the answer is no, give a safe state. If the answer is yes, identify the processes that will be involved in the deadlock. Remember that this is the **<u>detection</u>** version of the algorithm. Show your work.

	Avail		
	2	1	3
P_0	2	3	5
P_4	5	3	5
P_2	**6**	**4**	**5**

   Does not satisfy the need of P_1 or P_3
   P_1 and P_3 will be in a deadlock