Example (Cont'd)

- 5 processes P_0 through P_4
  - ✓ 3 resource types *A* (10 instances), *B* (5 instances), and *C* (7 instances)

	Max	Allocation	Need	Available
	A B C	*A B C*	*A B C*	*A B C*
P_0	7 5 3	0 1 0	7 4 3	3 3 2
P_1	3 2 2	2 0 0	1 2 2	
P_2	9 0 2	3 0 2	6 0 0	
P_3	2 2 2	2 1 1	0 1 1	
P_4	4 3 3	0 0 2	4 3 1	

$Available = Available - Request_i;$

$Allocation_i = Allocation_i + Request_i;$

$Need_i = Need_i - Request_i;$

- The system is in a safe state since the sequence $< P_1, P_3, P_4, P_2, P_0 >$ satisfies safety criteria

Example P_1 Request (1,0,2) (Cont'd)

- Check that Request ≤ Available (that is, (1,0,2) ≤ (3,3,2) ⇒ *true*)

	Allocation	*Need*	*Available*
	A B C	*A B C*	*A B C*
P_0	0 1 0	7 4 3	2 3 0
P_1	2 0 0->3 0 2	1 2 2 -> 0 2 0	
P_2	3 0 1	6 0 0	
P_3	2 1 1	0 1 1	
P_4	0 0 2	4 3 1	

- Executing safety algorithm shows that sequence <P_1, P_3, P_4, P_0, P_2> satisfies safety requirement

- Can request for (3,3,0) by P_4 be granted?
- Can request for (0,2,0) by P_0 be granted?

Term Project #3

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

- Is the System is safe state?
- Can request for (1,2,0,0) by P_4 be granted?

- Print the result (Safe or Unsafe) and Safe Sequence (If the system is safe)