

第一小题

进程	已获得资源 (Allocation)	最大资源数 (MAX)	还需资源数 (Need)	系统剩余 (Available)
	A B C D	A B C D	A B C D	A B C D
P0	0 0 3 2	0 0 4 4	0 0 1 2	2 6 4 4
P1	1 1 0 0	2 6 5 0	1 5 5 0	
P2	1 2 4 2	3 7 11 12	2 5 7 10	
P3	0 3 3 2	0 7 7 4	0 4 4 2	
P4	0 0 1 4	0 6 6 8	0 6 5 4	

进程	可用资源 (Work)	还需资源 (Need)	已获得资源 (Allocation)	可用资源+已获得 资源 (Work+Allocation)	是否完 成 (finish)
	A B C D	A B C D	A B C D	A B C D	
P0	2 6 4 4	0 0 1 2	0 0 3 2	2 6 7 6	T
P3	2 6 7 6	0 4 4 2	0 3 3 2	2 9 10 8	T
P1	2 9 10 8	1 5 5 0	1 1 0 0	3 10 10 8	T
P4	3 10 10 8	0 6 5 4	0 0 1 4	3 10 11 12	T
P2	3 10 11 12	2 5 7 10	1 2 4 2	4 12 15 14	T

存在安全序列：P0→P3→P1→P4→P2, 所以系统此时处于安全状态

第 2 小题

$$\text{Request}_2(1, 2, 2, 2) \leq \text{Need}_2(2, 5, 7, 10)$$

$$\text{Request}_2(1, 2, 2, 2) \leq \text{Available}_2(2, 6, 4, 4)$$

系统先假定可为 P2 分配资源，并修改已分配资源数，还需资源数，系统剩余资源数

进程	已获得资源 (Allocation)	最大资源数 (MAX)	还需资源数 (Need)	系统剩余 (Available)
	A B C D	A B C D	A B C D	A B C D
P0	0 0 3 2	0 0 4 4	0 0 1 2	1 4 2 2
P1	1 1 0 0	2 6 5 0	1 5 5 0	
P2	2 4 6 4	3 7 11 12	1 3 5 8	
P3	0 3 3 2	0 7 7 4	0 4 4 2	
P4	0 0 1 4	0 6 6 8	0 6 5 4	

进程	可用资源 (Work)	还需资源 (Need)	已获得资源 (Allocation)	可用资源+已获得资源 (Work+Allocation)	是否完成 (finish)
	A B C D	A B C D	A B C D	A B C D	T
P0	1 4 2 2	0 0 1 2	0 0 3 2	1 4 5 4	T
P3	1 4 5 4	0 4 4 2	0 3 3 2	1 7 8 6	T
P4	1 7 8 6	0 6 5 4	0 0 1 4	1 7 9 10	T
P1	1 7 9 10	1 5 5 0	1 1 0 0	2 8 9 10	T
P2	2 8 9 10	1 3 5 8	2 4 6 4	4 12 15 14	T

因为通过银行家算法分析，当进程 P2 发出 request (1, 2, 2, 2) 时，资源分配存在安全序列：

P0→P3→P4→P1→P2, 系统可以分配资源