

银行家算法解决死锁

❖ 有三类资源A(17)、B(5)、C(20)。有5个进程 P_1 — P_5 。T₀时刻系统状态如下：

	最大需求	已分配
P_1	5 5 9	2 1 2
P_2	5 3 6	4 0 2
P_3	4 0 11	4 0 5
P_4	4 2 5	2 0 4
P_5	4 2 4	3 1 4

????? :

- (1) T₀时刻是否为安全状态，给出安全序列。
- (2) T₀时刻， P_2 : Request(0,3,4)，能否分配，为什么？
- (3) 在(2)的基础上 P_4 : Request(2,0,1)，能否分配，为什么？
- (4) 在(3)的基础上 P_1 : Request(0,2,0)。能否分配。为什么？

	最大进程			已分配			需求			可利用		
	A	B	C	A	B	C	A	B	C	A	B	C
P1	5	5	9	2	1	2	3	4	7	2	3	3
P2	5	3	6	4	0	2	1	3	4			
P3	4	0	11	4	0	5	0	0	6			
P4	4	2	5	2	0	4	2	2	1			
P5	4	2	4	3	1	4	1	1	0			

	最大进程			已分配			总计		
	A	B	C	A	B	C	A	B	C
P4	2	3	3	2	0	4	4	3	7
P2	4	3	7	4	0	2	8	3	9
P3	8	3	9	4	0	5	12	3	14
P5	12	3	14	3	1	4	15	4	18
P1	15	4	18	2	1	2	17	5	20

- (1) T₀此刻是安全状态，安全序列为：P4，P2，P3，P5，P1
- (2) T₀时刻，T2: Request2 (0,3,4) 不能完成分配
 $\text{Request}_2(0,3,4) \leq \text{Need}_2(1,3,4)$ 成立
 $\text{Request}_2(0,3,4) \leq \text{Available}(2,3,3)$ 不成立，故不能分配
- (3) 在 (2) 的基础上 P_4 : $\text{Request}_4(2,0,1) \leq \text{Need}_4(2,2,1)$ 成立
 $\text{Request}_4(2,0,1) \leq \text{Available}(2,3,3)$ 成立

	最大进程			已分配			需求		
	A	B	C	A	B	C	A	B	C
P1	5	5	9	2	1	2	3	4	7
P2	5	3	6	4	0	2	1	3	4
P3	4	0	11	4	0	5	0	0	6
P4	4	2	5	4	0	5	0	2	0
P5	4	2	4	3	1	4	1	1	0

A,B,C 可利用的资源 (Available) 还剩 0, 3, 2

根据银行家算法可以得出结论：存在安全序列：P4, P2, P3, P5, P1。所以可以分配

(4) 在 (3) 的基础上 P1: Request(0,2,0)<Need(3,4,7)成立

Request(0,2,0)<Available(0,3,2)成立

	最大进程			已分配			需求		
	A	B	C	A	B	C	A	B	C
P1	5	5	9	2	3	2	3	2	7
P2	5	3	6	4	0	2	1	3	4
P3	4	0	11	4	0	5	0	0	6
P4	4	2	5	2	0	1	2	2	4
P5	4	2	4	3	1	4	1	1	0

A,B,C 可利用的资源 (Available) 0, 1, 2

这时可利用资源不能满足进程分配完成，系统处于不安全状态，找不到合适的安全序列，所以无法分配