

操作系统——算法题

银行家算法

某系统有A、B、C、D四类资源可供五个进程P1、P2、P3、P4、P5共享。系统对这四类资源的拥有量为:A类3个、B类14个、C类12个、D类12个。

进程对资源的需求和分配情况如下:

进程 已占有资源 最大需求数

	A	B	C	D	A	B	C	D
P1	0	0	1	2	0	0	1	2
P2	1	0	0	0	1	7	5	0
P3	1	3	5	4	2	3	5	6
P4	0	6	3	2	0	6	5	2
P5	0	0	1	4	0	6	5	6

按银行家算法回答下列问题:

- (1) 现在系统中的各类资源还剩余多少? (4分)
- (2) 现在系统是否处于安全状态? 为什么? (6分)
- (3) 如果现在进程P2提出需要A类资源0个、B类资源4个、C类资源2个和D类资源0个, 系统能否去满足它的请求? 请说明原因。 (6)

1、系统里有 A、B、C、D 四类资源。在银行家算法中，若出现下述的资源分配情况：

Process	Max				Allocation				Available			
	A	B	C	D	A	B	C	D	A	B	C	D
P0	0	0	4	4	0	0	3	2	1	6	2	2
P1	2	7	5	0	1	7	5	0	(0	4	0	0)
P2	3	6	10	10	1	3	5	4				
P3	0	9	8	4	0	6	5	2				
P4	0	6	6	10	0	0	1	4				

试问：

- (1) 该状态是否安全？ (4 分)
- (2) 若进程 P2 提出请求 Request (1, 2, 2, 2) 后，系统能否将资源分配给它？ (3 分)
- (3) 如果系统立即满足 P2 的上述请求，系统是否立即进入死锁状态？ (3 分)

四、应用题(每题 15 分, 共 30 分)

得分	评阅人

1. 设系统中有三种类型的资源 (A, B, C) 和五个进程 (P1, P2, P3, P4, P5), A 资源的数量 17, B 资源的数量为 5, C 资源的数量为 20。在 T0 时刻系统状态如下表所示。系统采用银行家算法来避免死锁。请回答下列问题：

- (1) T0 时刻是否为安全状态？若是，请给出安全序列。
- (2) 若进程 P4 请求资源 (2, 0, 1), 能否实现资源分配？为什么？
- (3) 在 (2) 的基础上，若进程 P1 请求资源 (0, 2, 0), 能否实现资源分配？为什么？

T0 时刻系统状态

进程	最大资源需求量			已分配资源量			系统剩余资源数量		
	A	B	C	A	B	C	A	B	C
P1	5	5	9	2 3	1 4	2 7	2	3	3
P2	5	3	6	4 1	0 3	2 4			
P3	4	0	11	4 0	0 0	5 6			
P4	4	2	5	2 2	0 2	4 1			
P5	4	2	4	3 1	1 1	4 0			

P4

4 3 7

P2

8 3 9

P3

12 3 14

P5

15 4 18

https://blog.csdn.net/weixin_44841312

(1) T0 时刻为安全状态。其中的一个安全序列为 (P4, P5, P3, P2, P1)

(其他可能的安全序列为：(P4, P5, X, X, X), (P4, P2, X, X, X), (P4, P3, X, X, X), (P5, X, X, X, X))

(2) 可以为 P4 分配资源，因为分配后的状态还是安全的，其安全序列的分析如下表：

	WORK	NEED	ALLOCATION	新 WORK	FINISH
	2, 3, 3	分配给 P4: (2, 0, 1)	0, 3, 2		
P4	0, 3, 2	0, 2, 0	4, 0, 5	4, 3, 7	True
P5	4, 3, 7	1, 1, 0	3, 1, 4	7, 4, 11	True
P1	7, 4, 11	3, 4, 7	2, 1, 2	9, 5, 13	True
P2	9, 5, 13	1, 3, 4	4, 0, 2	13, 5, 15	True
P3	13, 5, 15	0, 0, 6	4, 0, 5	17, 5, 20	True

P1

17 5 20

https://blog.csdn.net/weixin_44841312

(3) 进程 P1 再请求资源 (0, 2, 0), 则不能为之分配资源。因为分配资源后，不存在安全序列，其分析如下表：

	WORK	NEED	ALLOCATION	新 WORK	FINISH
	0, 3, 2	分配给 P1: (0, 2, 0)	0, 1, 2		
P4		0, 2, 0	此时, WORK 不能满足任何 一个进程的请求使之运行结 束, 即进入了不安全状态。		False
P5		1, 1, 0			False
P1		3, 2, 7			False
P2		1, 3, 4			False
P3		0, 0, 6			False

3. 假设某系统中有 3 种资源(R1,R2,R3), 在某时刻系统中共有 4 个进程, 进程(P1,P2,P3,P4)的最大资源需求向量和此时已分配的资源数向量分别为:

进程	最大资源需求	当前已分配到资源
P1	(3,2,2)	(1,0,0)
P2	(6,1,3)	(5,1,1)
P3	(3,1,4)	(2,1,1)
P4	(4,2,2)	(0,0,2)

系统中当前可用资源向量为(1,1,2), 问:

(1) 计算还需要资源数组;

	Max	Allocation	Need	Available
P1	(3,2,2)	(1,0,0)	(2,2,2)	(1,1,2)
P2	(6,1,3)	(5,1,1)	(1,0,2)	
P3	(3,1,4)	(2,1,1)	(1,0,3)	
P4	(4,2,2)	(0,0,2)	(4,2,0)	

(2) 系统此时是否安全?

	Work	Need	Allocation	Work + Allocation	Finish
P2	(1,1,2)	(1,0,2)	(5,1,1)	(6,2,3)	true
P3	(6,2,3)	(1,0,3)	(2,1,1)	(8,3,4)	true
P4	(8,3,4)	(4,2,0)	(0,0,2)	(8,3,6)	true
P1	(8,3,6)	(2,2,2)	(1,0,0)	(9,3,6)	true

(3) 如果进程 P2 发出资源请求向量(1,0,1), 系统能否将资源分配给它?

- ① Request₂(1, 0, 1) ≤ Need₂(1, 0, 2) ✓
- ② Request₂(1, 0, 1) ≤ Available₂(1, 1, 2) ✓

	Max	Allocation	Need	Available
P1	(3,2,2)	(1,0,0)	(2,2,2)	(0,1,1)
P2	(6,1,3)	(6,1,2)	(0,0,1)	
P3	(3,1,4)	(2,1,1)	(1,0,3)	
P4	(4,2,2)	(0,0,2)	(4,2,0)	

(3) 再利用安全性算法检查此时系统是否安全。存在一个安全序列: P2, P3, P4, P1

	Work	Need	Allocation	Work + Allocation	Finish
P2	(0,1,1)	(0,0,1)	(6,1,2)	(6,2,3)	true
P3	(6,2,3)	(1,0,3)	(2,1,1)	(8,3,4)	true
P4	(8,3,4)	(4,2,0)	(0,0,2)	(8,3,6)	true
P1	(8,3,6)	(2,2,2)	(1,0,0)	(9,3,6)	true

在银行家算法中，若出现下述的资源分配情况：

Process	Allocation	Need	Available
p0	1,0,2,3	7,6,6,1	3,0,1,4
p1	2,1,0,1	6,6,5,7	
p2	1,0,4,1	3,5,1,5	
p3	0,3,2,1	3,1,2,8	
p4	3,2,1,0	3,2,2,5	
p5	0,1,2,6	2,0,1,2	

Available的变化
p5⇒3,1,3,10
p3⇒3,4,5,11
p4⇒6,6,6,11
p1⇒8,7,6,12
p0⇒9,7,8,15
p2⇒10,7,12,16
(1) 安全！

试问：(1) 该状态是否安全？

(2) 若进程p1提出请求Request(2,0,1,1)后，系统能否将资源分配给它？

need Available

P1⇒ 4,6,4,6 1,0,0,3

如果满足(2, 0, 1, 1)请求，则所有其它请求再也无法满足，造成死锁。

调度算法