银行家算法

1. 尝试分配：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 最大需求 | 已分配 | 需求 | 分配+工作 | Finish |
| P4 | (2,3,3) | (2,0,4) | (2,2,1) | (4,3,7) | TRUE |
| P2 | (4,3,7) | (4,0,2) | (1,3,4) | (8,3,9) | TRUE |
| P3 | (8,3,9) | (4,0,5) | (0,0,6) | (12,3,14) | TRUE |
| P5 | (12,3,14) | (3,1,4) | (1,1,0) | (15,4,18) | TRUE |
| P1 | (15,4,18) | (2,1,2) | (3,4,7) | (17,5,20) | TRUE |

安全{p4,p2,p3,p5,p1}

2. 不能

T0时刻，Available(2,3,3)

Request(0,3,4)>Available(2,3,3) 请求不合理

3. 能

Request (2,0,1) < Available (2,3,3)

Request (2,0,1) < Need (2,2,1) 分配资源进行安全性检测,可得安全序列（p4,p2,p3,p5,p1） ,系统处在安全状态所以资源能分配给 p4。

4. 不能

Request (0,2,0) < Available (0,3,2) Request (0,2,0) < Need (3,4,7) 为 p1 分配资源后可用资源向量变为（0，1，2），不能满足任意一个进程的需要，所以不能。