

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Max | | | Allocation | | | Need | | |
|  | 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 | 2 | 0 | 4 | 2 | 2 | 1 |
| P5 | 4 | 2 | 4 | 3 | 1 | 4 | 1 | 1 | 0 |

T0时刻可分配资源：A：2 B：3 C：3 因此可分配资源向量为2,3,3

第一步找到P4可以分配，向量变更：<4,3,7>

第二步找到P2可以分配，向量变更：<8,3,9>

第三步找到P3可以分配，向量变更：<12,3,18>

第四步找到P5可以分配，向量变更：<15,4,18>

第五步找到P1可以分配，向量变更：<17,5,20>

T0时刻是安全状态，因为可以找到一个安全的序列(P4，P5，P1，P2，P3)。  
(2)在T0时刻 P2的request（0,3,4）<Need(1,3,4) 符合条件

但是 request（0,3,4）部分大于（2,3,3）不能分配

(3)P4的request（2,0,1）<Need(2,2,1) 符合条件且（2,0,1）<(2,3,3)成立

因此可分配资源向量为0,3,2

第一步找到P4可以分配，向量变更：<4,3,7>

第二步找到P2可分配，向量变更：<8,3,9>

第三步找到P3可以分配，向量变更：<12,3,14>

第四步找到P5可以分配，向量变更：<15,4,18>

第五步找到P1可以分配，向量变更：<17,5,20>

可找到一个安全的序列队(P4，P5，P1，P2，P3)。

(4)不能分配。若分配完成后，系统剩余的资源数量为(0，1，2)，第一步就找不到可执行的程序，这时无法找到一个安全的序列。