第二次作业 银行家算法问题

有三类资源A（17）、B（5）、C（20）。有5个进程P1-P5。T0时刻操作系统状态如下：

|  |  |  |
| --- | --- | --- |
|  | 最大需求 | 已分配 |
| P1 | 5 5 9 | 2 1 2 |
| P2 | 5 3 6 | 4 0 2 |
| P3 | 4 0 11 | 4 0 5 |
| P4 | 4 2 5 | 2 0 4 |
| P5 | 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，P2:Request2(0,3,4)，能否分配，为什么？
3. 在（2）的基础上P4:Request4(2,0,1)，能否分配，为什么？
4. 在（3）的基础上P1:Request1(0,2,0)，能否分配，为什么？

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | max | | | allocation | | | need | | | available | | |
|  | 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 |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | work | | | need | | | allocation | | | Work&allocation | | | finish |
|  | A | B | C | A | B | C | A | B | C | A | B | C |
| P4 | 2 | 3 | 3 | 2 | 2 | 1 | 2 | 0 | 4 | 4 | 3 | 7 | T |
| P2 | 4 | 3 | 7 | 1 | 3 | 4 | 4 | 0 | 2 | 8 | 3 | 9 | T |
| P3 | 8 | 3 | 9 | 0 | 0 | 6 | 4 | 0 | 5 | 12 | 3 | 14 | T |
| P5 | 12 | 3 | 14 | 1 | 1 | 0 | 3 | 1 | 4 | 15 | 4 | 18 | T |
| P1 | 15 | 4 | 18 | 3 | 4 | 7 | 2 | 1 | 2 | 17 | 5 | 20 | T |

T0时刻为安全状态

安全序列：P4-->P2-->P3-->P5-->P1

1. T0时刻

P2:Request2(0,3,4)

根据银行家算法：Request2(0,3,4)<=Need2(1,3,4)

Request2(0,3,4)>Available(2,3,3)

不能分配

1. 在（2）的基础上

P4:Request4(2,0,1)

根据银行家算法：Request4(2,0,1)<=Need4(2,2,1)

Request2(2,0,1)<=Available(2,3,3)

假定可以为P4分配资源

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | max | | | allocation | | | need | | | available | | |
|  | 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 |
|  |  |  |  |  |  |  |  |  |  | 0 | 3 | 2 |
| 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 |  |  |  |
|  |  |  |  | 4 | 0 | 5 | 0 | 2 | 0 |  |  |  |
| P5 | 4 | 2 | 4 | 3 | 1 | 4 | 1 | 1 | 0 |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | work | | | need | | | allocation | | | Work&allocation | | | finish |
|  | A | B | C | A | B | C | A | B | C | A | B | C |
| P4 | 0 | 3 | 2 | 0 | 2 | 0 | 4 | 0 | 5 | 4 | 3 | 7 | T |
| P2 | 4 | 3 | 7 | 1 | 3 | 4 | 4 | 0 | 2 | 8 | 3 | 9 | T |
| P3 | 8 | 3 | 9 | 0 | 0 | 6 | 4 | 0 | 5 | 12 | 3 | 14 | T |
| P5 | 12 | 3 | 14 | 2 | 2 | 1 | 3 | 1 | 4 | 15 | 4 | 18 | T |
| P1 | 15 | 4 | 18 | 3 | 4 | 7 | 2 | 1 | 2 | 17 | 5 | 20 | T |

安全序列：P4-->P2-->P3-->P5-->P1

处于安全状态，可以分配。

1. 在（3）的基础上

P1:Request1(0,2,0)

根据银行家算法：Request1(0,2,0)<=Need1(3,4,7)

Request1(0,2,0)<=Available(0,3,2)

假定可以为P1分配资源

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | max | | | allocation | | | need | | | available | | |
|  | 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 |
|  |  |  |  | 2 | 3 | 2 | 3 | 2 | 7 | 0 | 3 | 2 |
|  |  |  |  |  |  |  |  |  |  | 0 | 1 | 2 |
| 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 |  |  |  |
|  |  |  |  | 4 | 0 | 5 | 0 | 2 | 0 |  |  |  |
| P5 | 4 | 2 | 4 | 3 | 1 | 4 | 1 | 1 | 0 |  |  |  |

Available(0,1,2)无法满足任何一个进程的need，无法找到安全序列，不能分配。