

电子科技大学

计算机专业类课程

实验报告

课程名称：操作系统

学 院：计算机科学与工程学院

专 业：信息安全

学生姓名：刘汝佳

学 号：2013060202030

指导教师：薛瑞尼

电子科技大学

实验报告

实验二、银行家算法程序

学生姓名：刘汝佳

学号：2013060202030

实验地点：主楼 A2-412

实验时间：第 15 周周日

实验学时：4 学时

一、实验目的：

编程实现银行家算法程序

二、实验原理：

• 死锁的概念：

多个进程在运行过程中因争夺资源而造成的一种僵局（Deadly-Embrace），当进程处于这种僵持状态时，若无外力作用，它们都将无法再向前推进。

• 死锁的充要条件：

1. 互斥条件

进程对所分配到的资源进行排它性使用。如果此时还有其它进程申请该资源，则只能阻塞，直至占有该资源的进程释放。

2. 占有且等待（请求和保持条件）

进程已经占有了至少一个资源，又提出了新的资源要求，而该资源已被其它进程占有，此时请求进程阻塞，且对已经获得的其它资源保持不放。

3. 非抢占（非剥夺）条件

进程已获得的资源，在未使用完之前，不能被剥夺，只能在使用完时由自己释放。

4. 循环等待条件

在发生死锁时，必然存在一个进程—资源的封闭的环形链。

• 处理死锁的基本方法：

1. 死锁预防

通过限制申请资源的方法来破坏产生死锁的条件

2. 忽略

3. 避免

在系统运行过程中，对进程发出的每一个资源申请进行检查，并根据检查结果决定是否分配资源，若分配后系统可能发生死锁，则不予分配（阻塞），否则予以分配。

防止系统进入不安全状态，从而避免发生死锁。

4. 检测解除

• 安全状态：

一个进程序列 $\{P_1, \dots, P_n\}$ 是安全的，如果对于每一个进程 $P_i (1 \leq i \leq n)$ ，它尚需要的资源量不超过系统当前剩余资源量与所有进程 $P_j (j < i)$ 当前占有资源量之和。

• 银行家算法的数据结构：

设系统中有 m 种不同资源， n 个进程

$Resource[j]$ ， j 资源的数量

Available 向量：系统中尚未分配的每种资源的总量

Available[j]: 尚未分配的资源 j 的数量

Max 矩阵: 各个进程对每种资源的最大需求量(进程事先声明)

Max[i, j](Claim[i, j]): 进程 i 对资源 j 的最大需求量

Allocation 矩阵: 当前资源分配状况

Allocation[i, j]: 进程 i 获得的资源 j 的数量

Need 矩阵: 每个进程还需要的剩余资源的数量

Need[i, j]: 进程 i 尚需的资源 j 的数量

三、实验要求:

* 输入

- * `p`: 进程数量

- * `r`: 资源数量

- * 各进程的 `max`, `allocation`

* 输出

- * 若产生死锁, 打印提示: `死锁状态`。

- * 否则, 给出一种调度顺序。

四、实验截图:

```

请输入进程数目:
2
请输入资源种类数:
2
请输入各类可利用资源的数量Available[2]:
3 2

请输入各进程当前已分配的资源数量Allocation[2][2]:
3 5
4 7

请输入各进程对各类资源的最大需求数Max[2][2]:
8 9
8 9

当前状态不安全Press any key to continue

```

图2 状态不安全

```

请输入进程数目:
3
请输入资源种类数:
2
请输入各类可利用资源的数量Available[2]:
3 2

请输入各进程当前已分配的资源数量Allocation[3][2]:
3 4
3 4
3 4

请输入各进程对各类资源的最大需求数Max[3][2]:
5 5
5 5
5 5

当前状态安全, 安全序列为:
->p0->p1->p2

请输入申请资源的进程编号<0,1...,n-1>:

```

图3 状态安全

五、实验总结:

通过对算法的实际代码实现，对银行家算法中的数据结构以及各数据间的关系有了更加深刻的理解。