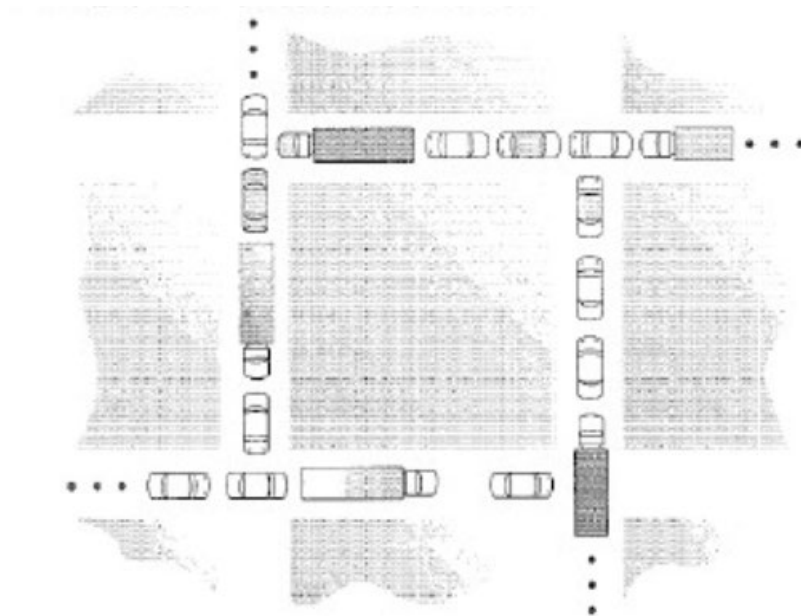


操作系统第七章作业

计试 81 白思雨 2186123935

7.1 考虑如图 7-10 所示的交通死锁：



- a, 证明这个例子中实际上包括了死锁发生的四个必要条件
- b, 给出一个简单的规则用来在这个系统中避免死锁

答：(1) 此例中导致死锁的四个条件成立

①互斥。每条道路只能被一辆车占用。

②占用并等待, 每辆车都占用了一段道路, 并等待其前方的道路被释放。

③非抢占。资源不可抢占。单行线, 汽车不能抢路超车。

④循环等待。每辆车都等待着前方的汽车把路让出来且形成了一个环路。

2) 在每个十字路口设置红绿灯, 当南北方向的路通车时, 东西方向的路上汽车等待。反之亦然。

7.3 图 7-4 所示的示例程序并不总能导致死锁。请描述 CPU 调度程序扮演什么角色, 以及如何做才能有助于程序的死锁。

答：CPU 调度应当使得线程二没有获得两个互斥锁之前线程一没有释放两个 mutex，并且应当使得两个线程的调度满足所需的四个条件。

7.11 当数组的维数减 1 时，从一般银行家算法中，容易得到只针对一种类型资源的银行家算法。通过例子来说明，多资源类型的银行家算法方案不能通过对每种资源类型单独地应用单种资源的银行家算法来实现。

答：考虑一个具有资源 A、B 和 C 的系统，它的进程 P0、P1、P2、P3 和 P4 的分配值如下：

P0	0	1	0
P1	3	0	2
P2	3	0	2
P3	2	1	1
P4	0	0	2

以及以下的需求值：

需求			
	A	B	C
P0	7	4	3
P1	0	2	0
P2	6	0	0
P3	0	1	1
P4	4	3	1

如果 Available 的值是 (2 3 0)，我们可以看到进程 P0 的 (0 2 0) 请求不能被满足，因为这将 Available 降低到 (2 10 0)，并且没有进程可以安全地完成。

然而，如果我们将这三种资源视为银行家算法的三种单资源类型，我们得到如下：

对于资源 A(我们有 2 个可用资源)，

	分配	需求
P0	0	7
P1	3	0
P2	3	6

P3	2	0
P4	0	4

流程可以按照 P1、P3、P4、P2、P0 的顺序安全地完成。对于资源 B(我们现在有 1 个可用资源，假设 2 分配给进程 P0)，

	分配	需求
P0	3	2
P1	0	2
P2	0	0
P3	1	1
P4	0	3

进程可以按照 P2、P3、P1、P0、P4 的顺序安全地完成。最后，对于资源 C(我们有 0 个可用资源)

	分配	需求
P0	0	3
P1	2	0
P2	2	0
P3	1	1
P4	2	1

流程可以按照 P1、P2、P0、P3、P4 的顺序安全地完成。如我们所见，如果我们对多种资源类型使用 banker ' s 算法，来自进程 P0 的资源请求 (0 2 0) 将被拒绝，因为它将使系统处于不安全状态。然而，如果我们考虑使用单一资源类型的三个独立资源的银行家算法，请求就会被批准。因此，如果我们有多个资源类型，我们必须对多个资源类型使用银行家算法。