第七章 死锁

1. Consider the traffic deadlock depicted in Figure 1.

a. Show that the four necessary conditions for deadlock indeed hold in this example.

b. State a simple rule for avoiding deadlocks in this system.

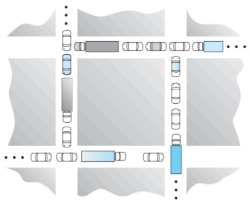

Figure 1. Traffic deadlock

a.

①互斥：路口的每个区域只能被一辆车占用，

②持有并等待：每个车都已经"持有"了路口的一块区域，同时"等待"下一个区域

③不可抢占：没有外力可以把车从占据的路口区域"挪出"，只能自己离开。

④循环等待：车辆形成了循环的等待链，A车等待B车的区域，B等C，C等D，D等A，互相等待对方的资源。

b.

打破持有并等待

车辆在进入交叉路口，必须能一次性通过整个路口。

即执行前申请所有资源

2. Consider the following snapshot of a system:

	Allocation	*Max*	*Available*
	A B C D	A B C D	A B C D
P_0	0012	0012	1520
P_1	1000	1750	
P_2	1354	2356	
P_3	0632	0652	
P_4	0014	0656	

Answer the following questions using the banker's algorithm:

a. What is the content of the matrix Need?

b. Is the system in a safe state?

c. If a request from process P_1 arrives for (0,4,2,0), can the request be granted immediately?

a. $need_0 = [0,0,0,0]$

$need_1 = [0,7,5,0]$

$need_2 = [1,0,0,2]$

$need_3 = [0,0,2,0]$

$need_4 = [0,6,4,2]$

b. ① $need_0 \le available$

P_0 释放, $available = [1,5,2,0] + [0,0,1,2] = [1,5,3,2]$

② $need_2 \le available$

P_2 释放, $available = [1,5,3,2] + [1,3,5,4] = [2,8,8,6]$

③ $need_3 \le available$

P_3 释放, $available = [2,8,8,6] + [0,6,3,2] = [2,14,11,8]$

④ $need_4 \le available$

P_4 释放, $available = [2,14,11,8] + [0,0,1,4] = [2,14,12,12]$

⑤ $need_1 \le available$

P_1 释放, $available = [2,14,12,12] + [1,0,0,0] = [3,14,12,12]$

系统处于安全状态 序列为 $P_0 \to P_2 \to P_3 \to P_4 \to P_1$

c.

$R_1 \le need_1$ $[0,4,2,0] < [0,7,5,0]$

$R_1 \le Available$ $[0,4,2,0] < [1,5,2,0]$

$Allocation_1 = [1,0,0,0] + [0,4,2,0] = [1,4,2,0]$

$need_1 = [0,7,5,0] - [0,4,2,0] = [0,3,3,0]$

$Available = [1,5,2,0] - [0,4,2,0] = [1,1,0,0]$

$need_0 < Available \Rightarrow [0,0,0,0] < [1,1,0,0]$

$Available = [1,1,0,0] + [0,0,1,2] = [1,1,1,2]$

$need_2 < Available \Rightarrow [1,0,0,2] < [1,1,1,2]$

$Available = [1,1,1,2] + [1,3,5,4] = [2,4,6,6]$

$need_1 < Available \Rightarrow [0,3,3,0] < [2,4,6,6]$

$Available = [2,4,6,6] + [1,4,2,0] = [3,8,8,6]$

$need_3 < Available \quad [0,0,2,0] < [3,8,8,6]$

$Available = [3,8,8,6] + [0,6,3,2] = [3,14,11,8]$

$need_4 < Available \quad [0,6,4,2] < [3,14,11,8]$

$Available = [3,14,11,8] + [0,0,1,4] = [3,14,12,12]$

是安全的序列为 $P_0 \rightarrow P_2 \rightarrow P_1 \rightarrow P_3 \rightarrow P_4$

请求可以立即授予