

7.4 修改transaction()来防止死锁

```
void transaction(from, to, int amount){
    Semaphore lock1, lock2, lock3;
    wait(lock3);
    lock1 = getLock(from);
    lock2 = getLock(to);
    wait(lock1);
    wait(lock2);
    withdraw(from, amount);
    deposit(to, amount);
    signal(lock3);
    signal(lock2);
    signal(lock1);
}
```

7.6

- a. 安全
- b. 可能会造成死锁，因为系统会认定有特定数量的可用资源
- c. 会对系统产生影响，造成死锁
- d. 安全
- e. 会分配资源给不安全进程
- f. 安全

7.13

- a. 一个可行的顺序P0->P3->P1->P2->P4
- b. 不可行，因为会导致初始无可满足进程
- c. 可行，可以将P4放到最后执行

9.5

虚拟地址是 0001 0001 0001 0010 0011 0100 0101 0110

因为页大小是 2^{12} ，页表大小是 2^{20} ，所以低12位 0100 0101 0110 代表页位置，高20位 0001 0001 0001 0010 0011 是页表位置。

9.8

LRU：最近最少访问的页剔除。共两次命中，2和7命中。18次缺页

FIFO：先进先出算法，2，3，7命中。共17次缺页

最优置换：从主存中移出永远不再需要的页面，如没有这种页面，则移除最长时间不需要访问的页面。2,3,7,1,5,0,1命中。共13次缺页

9.14

- a. 安装更快的CPU：不可以提高利用率
- b. 安装更大的分页磁盘：不可以
- c. 提高多道程序：不可以
- d. 降低多道程序：可以
- e. 安装更多内存：可以，因为会有更多的页，减少了从硬盘调用的过程
- f. 安装更快的硬盘或具有多个硬盘的多个控制器：可以，因为可以加快从硬盘调用数据的速度
- g. 为页面获取算法添加预先调页：可以，因为可以进行预取
- h. 增加页面大小：可以，因为减少了 `page fault`