操作系统第六次作业参考解答

1、当哲学家一次只拿一双筷子时，哲学家就餐问题会出现死锁。请讨论这种情况下的4个死锁必要条件确实存在，并讨论如何通过取消4个中的一个必要条件来避免死锁（提出一种可能的解决方案即可）

解答:

死锁必要条件:

互斥使用：筷子是互斥使用的

占有且等待：哲学家占有第一根筷子，等待第二根筷子

不可抢占：哲学家已经占有的筷子无法被抢占

循环等待：存在哲学家序列P1，P2，…，Pn，使得P1等待P2占有的筷子，P2等待P3占有的筷子，…，Pn等待P1占有的筷子

解决方案:

（破坏互斥使用：允许同时使用筷子）

破坏占有且等待：当哲学家无法获得第二根筷子时，放弃第一根筷子

破坏不可抢占：合理的情况下，允许抢占哲学家已经占有的筷子

破坏循环等待：筷子编号，哲学家先取编号小的筷子，再取编号大的筷子

2、两个进程A和B，每一个进程都需要申请资源 1，2，3。假如这两个进程都以1、2、3的次序申请，系统不会发生死锁。但如果A以3、2、1的次序申请，B以1、2、3的次序申请，则死锁可能发生。试计算当两个进程申请资源的次序

不确定的情形下，系统保证不发生死锁的概率是多少?

解答:

假设进程 A 申请资源顺序 123

A 123 B 123 不会死锁

A 123 B 132 不会死锁

A 123 B 213 可能死锁

A 123 B 231 可能死锁

A 123 B 312 可能死锁

A 123 B 321 可能死锁

A 其他情况类似

总的来说，不会死锁的概率是 1/3

3、系统有同类资源 n 个，使 n 个进程共享，如果每个进程对资源的最大需求量为 k，证明，当 $m \geq n \times (k-1)+1$ 时，系统不会发生死锁

解答:

不会发生死锁的含义就是无论何时，至少一个进程可以得到全部的资源并执行完毕。考虑极端情况，所有的进程都获得了比最大需求资源数少 1 的资源，也就是说只差 1 个资源就可以完成运行，但此时系统恰好没有空余资源，则所有进程死锁，此时 m、n、k 三个变量所满足的关系为 $m \equiv n \times (k-1)$。而此时系统只要再多 1 个资源就可以消除死锁。因此，只要能够满足 $m \geq n \times (k-1)+1$，系统就一

定不会产生死锁。

4、银行家算法中，出现以下资源分配情况

进程	资源最大数量	已分配资源
P0	7 5 3	0 1 0
P1	3 2 2	2 1 0
P2	9 0 2	3 0 2
P3	2 2 2	2 1 1
P4	4 3 3	0 0 2

系统剩余资源数量=（3，2，2）

试问：

（1）该状态是否安全？请给出详细的检查过程

（2）若进程依次有如下资源请求：

P1：资源请求 Request（1，0，2）

P4：资源请求 Request（3，3，0）

P0：资源请求 Request（0，1，0）

则系统如何进行资源分配，才能避免死锁？

解答：

（1）系统安全，因为存在安全序列（P1，P3，P0，P2，P4）。过程如下

先求出各进程剩余的需求量

P0=（7，4，3），P1=（1，1，2），P2=（6，0，0）

P3=（0，1，1），P4=（4，3，1）

根据系统的剩余资源数（3，2，2）可知，可以立即满足的进程是 P1，P1 满足后可释放占有的资源，系统剩余资源数为（5，3，2），找到可立即满足的进程是 P3，P3 满足后释放占有的资源，系统此时剩余资源数为（7，4，3），找到可立即满足的进程是 P0，P2，P4，所有进程可以依次执行完毕

（2）系统想要避免死锁，就必须保证每次分配完后都能得到安全序列，否则就拒绝分配。根据这一原则，对于进程的请求应考虑分配后是否安全，若不安全，则不能进行此次分配。题目中有 3 个请求，按照顺序来一次考虑。先考虑能否满足 P1，分配后系统处于安全状态，因此分配后可以找到安全序列（P1，P3，P2，P0，P4）。满足 P1 的请求之后，剩余资源为（2，2，0）。对于 P4 的请求，由于系统没有那么多剩余资源，因此无法满足，系统拒绝 P4 的请求。最后考虑 P0 的请求，如果满足 P0，分配后剩余资源为（2，1，0），可以找到安全序列（P1，P3，P0，P2，P4），因此可以满足 P0 的请求。总之，系统对 3 的请求的处理依次为：满足 P1，拒绝 P4，满足 P0。