# 操作系统作业

### 2016011977 蔡蕊蕊

1. 解释死锁和饥饿的定义，并标明出自哪本资料书，然后举例子来解释死锁和饥饿的区别，或者用通俗易懂的语言解释。

答：（1）**死锁**：如果一组进程中的每一个进程都在等待仅由该组进程中的其他进程才能引发的事件，那么该组进程是死锁的。***-----第四版《计算机操作系统》3.5.3 P107***

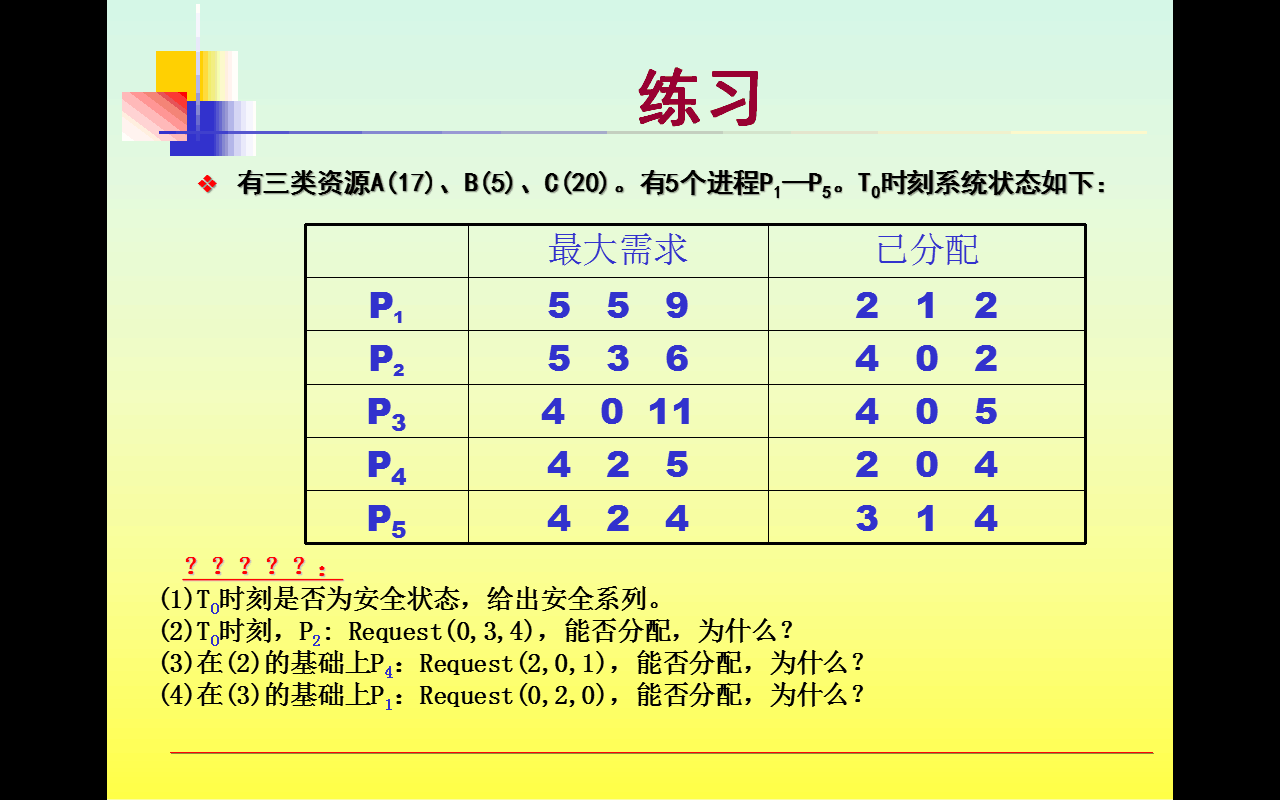
**饥饿**：饥饿是指系统不能保证某个进程的等待时间上界，从而使该进程长时间等待，当等待时间给进程推进和响应带来明显影响时，称发生了**进程饥饿**。当饥饿到一定程度的进程所赋予的任务即使完成也不再具有实际意义时称该进程被**饿死**。-***------来源于百度百科***

**饥饿：**进程在其生存周期中需要很多不同类型的资源。由于进程往往是动态创建的，这样，在任何时候系统中都会出现资源申请。何时、为哪个进程、分配什么资源、以及分配多少资源，是系统分配资源的策略问题。在这种策略下，系统会出现这样一种情况：在可以预计的时间内，某个或某些进程永远得不到完成工作的机会，因为他们所需的资源总是被别的进程占有或抢占，这种状态称作“饥饿”或者“饿死”（Starvation）  ***-----来源于第三版《计算机操作系统》***

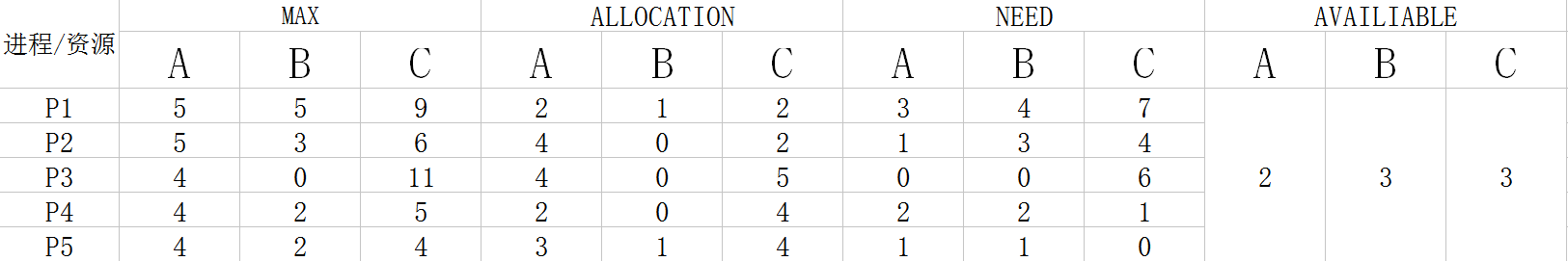
1. **死锁的例子**可以参考哲学家进餐问题，有五个哲学家和五只筷子，使每个哲学家先拿左手的筷子，再拿右手的筷子。当每个哲学家都先拿左手的筷子并且都拿到了，此时五只筷子都被哲学家拿在手中，右手的筷子都拿不到，没有人释放“筷子”这一资源，致使没有任何一个进程被唤醒，只能无限期地等待下去，故陷入了死锁的状态。

**饥饿的例子**如考虑一台打印机分配的问题，当有多个进程需要打印文件时，系统按照短文件优先的策略排序，该策略具有平均等待时间短的优点，似乎非常合理，但当短文件打印任务源源不断时，长文件的打印任务将被无限期地推迟，导致饥饿以至饿死。

1. 所以区别就是：
2. 死锁就是进程等待永远不会被释放的资源，饿死进程等待会被释放但却不会分配给自己的资源，表现为等待时限没有上界(排队等待或忙式等待)；
3. 死锁一定涉及多个进程，而饥饿或被饿死的进程可能只有一个。
4. 死锁就是有大家都互相占着某一样东西，都不愿意让步，但是想要达到目的，必须再拥有一个甚至更多，由于都不愿意放手，就一直僵着谁都没有办法完成任务。
5. 饥饿就是类似于闹饥荒，假如我是个男人，我正在排队领碗筷，但是有个规定就是遇见老人小孩和妇女，必须让他们先领，但是老人小孩妇女源源不断，我就一直要等着，然后我可能饿死了。



（1）此时为T0时刻的状态



1. 由表可知，此时A、B、C三者可用资源为（2,3,3）。此时满足P4、P5的需求，那么现将资源分配给P4，则在一段时间内，P4完成任务，释放资源（4,2,5），此时可用资源为（4,3,7）。
2. 此时可用资源满足P2（1,3,4）、P3（0,0,6）、P5（1,1,0），选择P2，将资源分配给P2，则在一定时间内，P2完成任务，释放资源（5,3,6），此时可用资源为（8,3,9）。
3. 此时可用资源满足P1、P3、P5，选择P1，将资源分配给P1，则在一定时间内，P1完成任务，释放资源（5,5,9），此时可用资源为（10,4,11）。
4. 此时可用资源满足P3、P5，选择P3，将资源分配给P3，则在一定时间内，P3完成任务，释放资源（4,0,11），此时可用资源为（14,4,16）。
5. 此时可用资源满足P5，将资源分配给P5，则在一定时间内，P5完成任务，释放资源（4,2,4），

可用资源为（17,5,20）

最后P1-P5全部完成，没有产生死锁状态，故在T0时是安全状态，安全序列为P4-P2-P1-P3-P5。

1. T0时刻，若P2 Request（0,3,4），此时在T0时刻，P2所需要资源为（1,3,4），比较可知，满足P2所请求的资源，故第一步骤得到满足。紧接着转向第二步骤，由于T0时刻可用资源为（2,3,3），比较可知，P2所请求的资源大于可用资源，**故尚无足够资源分配给P2，P2需要等待。**
2. 在（2）的基础上，T0时刻，若P4 Request（2,0,1），此时在T0时刻，P4所需要资源为（2,2,1），比较可知，满足P4所请求的资源，故第一步骤得到满足。紧接着转向第二步骤，由于T0时刻可用资源为（2,3,3），比较可知，P4所请求资源小于可用资源，故转向第三步骤，所以**系统可试探着把资源分配给P4，并进行安全性检查，检查系统是否安全。**
3. 在（3）的基础上，T0时刻，若P1 Request（0,2,0）,此时在T0时刻，P1所需要资源为（3,4,7），比较可知，满足P1所请求的资源，故第一步骤得到满足。紧接着转向第二步骤，由于T0时刻可用资源为（0,3,2），比较可知，P1所请求资源小于可用资源，故转向第三步骤，所以**系统可试探着把资源分配给P1，并进行安全性检查，检查系统是否安全。**