**操作系统习题集之死锁**

**一．选择题**

**1．C**

**2．D**

**3. (1) C (2) B**

**4. (1) B (2)A**

**5.C**

**6.B**

**7.D**

**8.B**

**9.B**

**二．填空题**

1.环路条件。

2.死锁

3.多个

**三．综合题**

**1. 何谓死锁？产生死锁的原因和必要条件是什么？**

*a*.死锁是指多个进程因竞争资源而造成的一种僵局，若无外力作用，这些进程都将永远不能再向前推进；

*b*.产生死锁的原因有二，一是竞争资源，二是进程推进顺序非法；

*c*.必要条件是: 互斥条件，请求和保持条件，不剥夺条件和环路等待条件。

**2．在解决死锁问题的几个方法中，哪种方法最易于实现？哪种方法是资源利用率最高？**

解决/处理死锁的方法有预防死锁、避免死锁、检测和解除死锁，其中预防死锁方法最容易实现，但由于所施加的限制条件过于严格，会导致系统资源利用率和系统吞吐量降低；而检测和解除死锁方法可是系统获得较好的资源利用率和系统吞吐量。

**3. 请详细说明可通过哪些途径预防死锁?**

*a*.摒弃"请求和保持"条件：系统规定所有进程开始运行之前，都必须一次性地申请其在整个运行过程所需的全部资源，但在分配资源时，只要有一种资源不能满足某进程的要求，即使其它所需的各资源都空闲，也不分配给该进程，而让该进程等待；

*b*.摒弃"不剥夺"条件：系统规定，进程是逐个地提出对资源的要求的。当一个已经保持了某些资源的进程，再提出新的资源请求而不能立即得到满足时，必须释放它已经保持了的所有资源，待以后需要时再重新申请；

*c*.摒弃"环路等待"条件：系统将所有资源按类型进行线性排序，并赋予不同的序号，且所有进程对资源的请求必须严格按序号递增的次序提出，这样，在所形成的资源分配图中，不可能再出现环路，因而摒弃了"环路等待"条件。

**4. 在银行家算法中，若出现下述资源分配情：**

|  |  |  |  |
| --- | --- | --- | --- |
| **Process** | **Allocation** | **Need** | **Available** |
| **P0** | **0032** | **0012** | **1622** |
| **P1** | **1000** | **1750** |
| **P2** | **1354** | **2356** |
| **P3** | **0333** | **0652** |
| **P4** | **0014** | **0656** |

**试问：**

**⑴ 该状态是否安全？**

**⑵ 若进程P2提出请求Request(1，2，2，2)后，系统能否将资源分配给它？**

⑴该状态是安全的，因为存在一个安全序列< P0P3P4P1P2>。下表为该时刻的安全序列表。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **资源情况**  **进程** | **Work** | **Need** | **Allocation** | **Work+Allocation** | **Finish** |
| **P0**  **P3**  **P4**  **P1**  **P2** | **1 6 2 2**  **1 6 5 4**  **1 9 8 7**  **1 9 9 11**  **2 9 9 11** | **0 0 1 2**  **0 6 5 2**  **0 6 5 6**  **1 7 5 0**  **2 3 5 6** | **0 0 3 2**  **0 3 3 3**  **0 0 1 4**  **1 0 0 0**  **1 3 5 4** | **1 6 5 4**  **1 9 8 7**  **1 9 9 11**  **2 9 9 11**  **3 12 14 15** | **true**  **true**  **true**  **true**  **true** |

⑵若进程P2提出请求Request(1，2，2，2)后，系统不能将资源分配给它，若分配给进程P2，系统还剩的资源情况为（0，4，0，0），此时系统中的资源将无法满足任何一个进程的资源请求，从而导致系统进入不安全状态，容易引起死锁的发生。

**5．设系统有三种类型的资源，数量为(4，2，2)，系统中有进程A，B，C按如下顺序请求资源：**

**进程A申请(3，2，1)**

**进程B申请(1，0，1)**

**进程A申请(1，0，0)**

**进程C申请(2，0，0)**

**请你给出一和防止死锁的资源剥夺分配策略，完成上述请求序列，并列出资源分配过程，指明哪些进程需要等待，哪些资源被剥夺。(10分)**

**解：(10分)**

**① 分配策略为：当进程Pi申请ri类资源时，检查ri中有无可分配的资源：有则分配给Pi；否则将Pi占有的资源全部释放而进入等待状态。(Pi等待原占有的所有资源和新申请的资源)**

**② 资源分配过程：　　　　　　　剩余资源**

**进程A：(3，2，1)　　　　　　(1，0，1)**

**进程B：(1，0，1)　　　　　　(0，0，0)**

**进程A：(1，0，0)(不满足)　　(3，2，1)**

**A的所有资源被剥夺，A处于等待**

**进程C：(2，0，0)　　　　　　(1，2，1)**

**C，B完成之后，A可完成。**