（1）产生死锁的必要条件是什么？解决死锁问题常用哪几种措施？

产生死锁的必要条件有：

互斥条件：在一段时间内某资源仅为一个进程所占有。

请求和保持条件：又称部分分配条件、占用并等待条件。当进程因请求资源被阻塞时，已分配资源保持不放。

不剥夺条件：进程所获得的资源在未使用完毕之前，不能被其他进程强行夺走。

循环等待条件：死锁发生时，存在一个进程资源的循环。

解决死锁问题常用的措施有：

忽略死锁。这种处理方式又称鸵鸟算法，指像鸵鸟一样对死锁视而不见。

预防死锁：设置某些限制条件，通过破坏死锁产生的四个必要条件之一来预防死锁。

避免死锁：在资源的动态分配过程中，用某种方法来防止系统进入不安全状态。

检测死锁及解除：系统定期检测是否出现死锁，若出现则解除死锁。

（2）设系统中仅有一类独占型资源，进程一次只能申请一个资源。系统中多个进程竞争该类资源。试判断下述哪些情况会发生死锁，为什么？

1）资源数为4，进程数为3，每个进程最多需要2个资源。

不会，假设每个进程都申请了最多需要资源数-1即1个资源，此时还剩余一个资源供任意一个进程申请并完成后释放供其他进程使用

2）资源数为6，进程数为2，每个进程最多需要4个资源。

会，假设每个进程都申请了最多需要资源数-1即3个资源，此时没有剩余的资源供进程再次申请使用，产生死锁

（3）单道批处理系统中，有四个作业，其有关情况如表2所示。若采用先来先服务、短作业优先、响应比高者优先调度算法，试分别计算其平均周转时间T和平均带权周转时间W。

先来先服务：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 作业 | 提交时间 | 运行时间 | 开始时间 | 完成时间 | 周转时间 | 带权周转时间 |
| J1 | 8.0 | 2.0 | 8 | 10 | 2 | 1 |
| J2 | 8.6 | 0.6 | 10 | 10.6 | 2 | 3.3 |
| J3 | 8.8 | 0.2 | 10.6 | 10.8 | 2 | 10 |
| J4 | 9.0 | 0.5 | 10.8 | 11.3 | 2.3 | 4.6 |

T=(2+2+2+2.3)/4=2.075s

W=(1+3.3+10+4.6)/4=4.725s

短作业优先：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 作业 | 提交时间 | 运行时间 | 开始时间 | 完成时间 | 周转时间 | 带权周转时间 |
| J1 | 8.0 | 2.0 | 8 | 10 | 2 | 1 |
| J2 | 8.6 | 0.6 | 10.7 | 11.3 | 2.7 | 4.5 |
| J3 | 8.8 | 0.2 | 10 | 10.2 | 1.4 | 7 |
| J4 | 9.0 | 0.5 | 10.2 | 10.7 | 1.7 | 3.4 |

T=(2+2.7+1.4+1.7)/4=1.95s

W=(1+4.5+7+3.4)/4=3.975s

响应比高者优先：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 作业 | 提交时间 | 运行时间 | 开始时间 | 完成时间 | 周转时间 | 带权周转时间 |
| J1 | 8.0 | 2.0 | 8 | 10 | 2 | 1 |
| J2 | 8.6 | 0.6 | 10.2 | 10.8 | 2.2 | 3.7 |
| J3 | 8.8 | 0.2 | 10 | 10.2 | 1.4 | 7 |
| J4 | 9.0 | 0.5 | 10.8 | 11.3 | 2.3 | 4.6 |

8s时，J1运行，J2、J3、J4依次到达

10s时，r2=1+1.4/0.6 r3=1+1.2/0.2 r4=1+1/0.5

r3最大，J3先运行

10.2s时，r2=1+1.6/0.6 r4=1+1.2/0.5

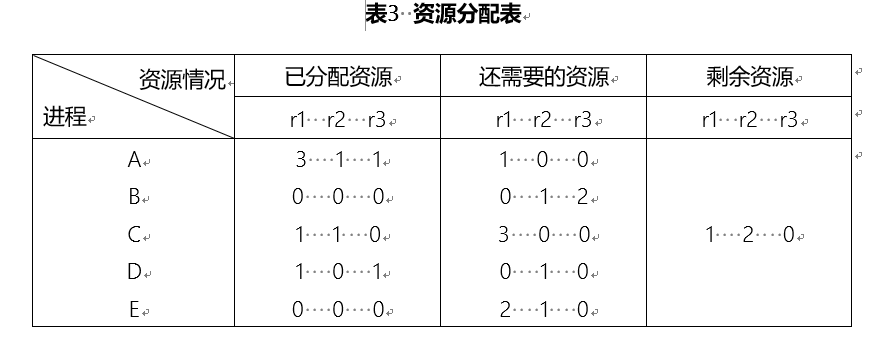
r2最大，J2先运行

10.8s时，J4运行

T=(2+2.2+1.4+2.3)/4=1.975s

W=(1+3.7+7+4.6)/4=4.075s

（4）表3给出了系统某时刻的资源分配情况：



试问：1）该状态是否安全?

安全，一个可行的安全序列为：ACDEB

2）如果进程B提出请求RequestB(0，1，0)，系统能否将资源分配给它？

可以，分配后的一个可行的安全序列为DABCE

3）如果进程E提出请求RequestE(0，1，0)，系统能否将资源分配给它？

可以，分配后的一个可行的安全序列为ACDEB