

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。

饿死：饿死是一种与死锁相反的情况，它发生在当操作系统对进程分配资源有太多限制的时候。《计算机科学导论》第三版

区别：当操作系统没有对进程的资源进行限制会发生死锁，当操作系统对进程分配资源有太多限制使进程无法进行会发生饿死，两者是相反的。

2.（1）在 T0 时存在安全序列{P4,P5,P3,P2,P1}，系统是安全的。

进程\资源情况	MAX(ABC)	NEED(ABC)	ALLOCATION(ABC)	WORK&ALLOCATION	FINISH
P4	2 3 3	2 2 1	0 1 2	4 3 7	TRUE
P5	4 3 7	1 1 0	3 2 7	7 4 11	TRUE
P3	7 4 11	0 0 6	7 4 5	11 4 16	TRUE
P2	11 4 16	1 3 4	10 1 12	15 4 18	TRUE
P1	15 4 18	3 4 7	12 0 11	17 5 20	TRUE

(2)  $\text{Request}_2(0,3,4) \leq \text{Need}_2(1,3,4)$ ,  $\text{Request}_2(0,3,4) > \text{Available}(2,3,3)$  不能分配

(3)  $\text{Request}_4(2,0,1) \leq \text{Need}_4(2,2,1)$ ,  $\text{Request}_4(2,0,1) \leq \text{Available}(2,3,3)$  P4 处在安全序列第一项，因此能形成安全序列{P4,P5,P3,P2,P1}使系统处于安全状态，可分配资源。

(4)  $\text{Request}_1(0,2,0) < \text{Need}_1(3,4,7)$ ,  $\text{Request}_1(0,2,0) < \text{Available}(4,3,7)$  假定分配，能形成安全序列{P5,P3,P1,P2}，系统是安全的，可分配资源

进程\资源情况	MAX(ABC)	NEED(ABC)	ALLOCATION(ABC)	WORK&ALLOCATION	FINISH
P5	4 1 7	1 1 0	3 0 7	7 2 11	TRUE
P3	7 2 11	0 0 6	7 2 5	11 2 16	TRUE
P1	11 2 16	3 2 7	8 0 9	13 5 18	TRUE
P2	13 5 18	1 3 4	12 2 14	17 5 20	TRUE