

E03: 并发进程

进程通信

一、单项选择题

1. 临界区是指 D。
A. 并发进程中用于实现进程互斥的程序段
B. 并发进程中用于实现进程同步的程序段
C. 并发进程中用户实现进程通信的程序段
D. 并发进程中访问共享变量的程序段
2. 相关临界区是指 D。
A. 一个独占资源
B. 并发进程中与共享变量有关的程序段
C. 一个共享资源
D. 并发进程中涉及相同变量的那些程序段
3. P、V 操作是 A。
A. 两条低级进程通信原语
B. 两组不同的机器指令
C. 两条系统调用命令
D. 两条高级进程通信原语
4. 若 P、V 操作的信号量 S 初值为 2，当前值为 -1，则表示有 B 等待进程。
A. 0 个
B. 1 个
C. 2 个
D. 3 个
5. 用 V 操作唤醒一个等待进程时，被唤醒进程的状态变为 B。
A. 等待
B. 就绪
C. 运行
D. 完成
6. 在执行一次 P 操作时，信号量的值应 C。
A. 不变
B. 加 1
C. 减 1
D. 减指定数值
7. 在执行 V 操作时，当信号量的值 D 时，应释放一个等待该信号量的进程。
A. >0
B. <0
C. ≥ 0
D. ≤ 0
8. 进程间的同步是指进程间在逻辑上的相互 B 关系。
A. 联接
B. 制约
C. 继续
D. 调用
9. D 是一种只能进行 P 操作和 V 操作的特殊变量。
A. 调度
B. 进程
C. 同步
D. 信号量
10. P、V 操作这种不可被中断的过程称为 B。
A. 初始化程序
B. 原语
C. 子程序
D. 控制模块
11. 实现进程互斥时，用 C 对应，对同一个信号量调用 P、V 操作实现互斥。
A. 一个信号量与一个临界区
B. 一个信号量与一个相关临界区
C. 一个信号量与一组相关临界区
D. 一个信号量与一个消息
12. 实现进程同步时，每一个消息与一个信号量对应，进程 D 可把不同的消息发送出去。
A. 在同一信号量上调用 P 操作
B. 在不同信号量上调用 P 操作
C. 在同一信号量上调用 V 操作
D. 在不同信号量上调用 V 操作

三、填空题

13. 临界资源的概念是 同时只允许一个进程使用的软件和硬件资源，而临界区是指 访问临界资源的程序段。

14. 若一个进程已进入临界区，其他欲进入临界区的进程必须等待。
15. 用 P、V 操作管理临界区时，任何一个进程在进入临界区之前应调用P操作，退出临界区时应调用V操作。
16. 有 m 个进程共享同一临界资源，若使用信号量机制实现对临界资源的互斥访问，则信号量值的变化范围是[1-m, 1]。
17. 操作系统中，对信号量 S 的 P 原语操作定义中，使进程进入相应等待队列等待的条件是s.value < 0。
18. 进程的互斥是指当有若干进程都要使用某一共享资源时，任何时刻最多只允许一个进程去使用。
19. 利用 PV 操作管理相关临界区时，必须成对出现。在进入临界区之前要调用P 操作，在完成临界区操作后要调用V 操作。
20. 若信号量的初值为 1，用 PV 操作能限制一次只有一个进程进入临界区操作。
21. 进程的同步是指并发进程之间存在一种制约关系，一个进程的执行依赖另一个进程的消息。
22. 用 PV 操作实现进程同步时，调用P 操作测试消息是否到达，调用V 操作发送消息。

五、简答题

23. 请给出 PV 操作的定义。假设 PV 操作用信号量 s 管理某个共享资源，请问当 s.value > 0, s.value = 0 和 s.value < 0 时，它们的物理意义是什么？

答：

PV 操作是指在信号量上进行的 P 操作和 V 操作。假定信号量为 s，则 P(s) 和 V(s) 的定义如下：

```

procedure p(Var s: semaphore);
begin
    s.value := s.value - 1;
    if s.value < 0 then sleep(s);
end;

procedure V(Var s: semaphore);
begin
    s.value := s.value + 1;
    if s.value <= 0 then wakeup(s);
end;
  
```

其中，sleep(s) 表示将调用 P(s) 过程的进程置成“等待信号量 s”的状态，且将其排入等待队列。wakeup(s) 表示释放一个“等待信号量 s”的进程，该进程从等待队列退出，并加入就绪队列中。

s.value > 0 表示该资源还有冗余（s.value 表示可使用的资源数或表示可同时使用资源的进程数），s.value = 0 表示该资源恰好被进程用完，s.value < 0 表示有进程正在等待使用该资源（|s.value| 表示等待使用资源的进程个数或表示等待进入临界区的进程个数）。

24. 复印室里有一个操作员为顾客复印资料，有 5 把椅子供顾客休息等待复印。如果没有顾客，则操作员休息。当顾客来到复印室时，如果有空椅子则坐下来，并唤醒复印操作员；如果没有空椅子则必须离开复印室。

答：变量waiting记录顾客数量，初始值为0。信号量mutex用来维护waiting的访问互斥，初始值为1。信号量customers表示有等待的顾客，初始值为0。信号量operators表示有就绪的操作员，初始值为0。

<pre>main(){ cobegin(){ operator(); customer(); } }</pre>	<pre>void operator(){ while(true){ P(customers); P(mutex); waiting--; V(operators); V(mutex); copy(); } }</pre>	<pre>void customer(){ while(true){ P(mutex); if(waiting < 5){ waiting++; V(customers); V(mutex); P(operators); get_copy(); } else{ V(mutex); leave(); } } }</pre>
---	---	--

另一种方法：

<pre>main(){ cobegin(){ operator(); customer(); } }</pre>	<pre>operator(){ while(1){ P(customers); 复印; V(operator); } }</pre>	<pre>customer(){ P(mutex) if(num < 5){ num++; V(mutex); V(customer); P(operator); 复印; P(mutex); num--; V(mutex); } else{ V(mutex); } } 离开; }</pre>
---	---	---

变量: num 顾客数

信号量: mutex 保证访问num互斥

customer 有顾客等待

operator 有操作员可工作

初值: mutex=1 num=0

customer=0 operator=0

死锁

一、单项选择题

25. 在为多道程序所提供的可共享的系统资源不足时，可能出现死锁。但是，不适当的 C 也可能产生死锁。
A. 进程优先权 B. 资源的线性分配
C. 进程推进顺序 D. 分配队列优先权
26. 产生死锁的四个必要条件是：互斥、B、循环等待和不剥夺。
A. 请求与阻塞 B. 请求与保持 C. 请求与释放 D. 释放与阻塞
27. 发生死锁的必要条件有四个，要防止死锁的发生，可以破坏这四个必要条件，但破坏 B-A 条件是不太实际的。
A. 互斥 B. 不可抢占 C. 部分分配 D. 循环等待
28. 资源的按序分配策略可以破坏 B-D 条件。
A. 互斥使用资源 B. 占有且等待资源
C. 非抢夺资源 D. 循环等待资源
29. 在 C 的情况下，系统出现死锁。
A. 计算机系统发生了重大故障
B. 有多个封锁的进程同时存在
C. 若干进程因竞争资源而无休止地相互等待他方释放已占有的资源
D. 资源数大大小于进程数或进程同时申请的资源数大大超过资源总数
30. 银行家算法是一种 B 算法。
A. 死锁解除 B. 死锁避免 C. 死锁预防 D. 死锁检测
31. 当进程数大于资源数时，进程竞争资源 B 会产生死锁。
A. 一定 B. 不一定
32. 在下列解决死锁的方法中，属于死锁预防策略的是 B。
A. 银行家算法 B. 资源有序分配法
C. 死锁检测法 D. 资源分配图化简法

二、填空题

33. 银行家算法中，当一个进程提出的资源请求将导致系统从 安全状态 进入 不安全状态 时，系统就拒绝它的资源请求。
34. 如果要求所有进程一次性申请它所需要的全部资源。若系统有足够的资源分配给进程，便一次把所有的资源分配给该进程。但在分配时只要有一种资源要求不能满足，则资源全不分配，进程等待。这种死锁预防方法破坏了死锁产生必要条件中的 请求与保持 条件。
35. 对待死锁，一般应考虑死锁的预防、避免、检测和解除四个问题。典型的银行家算法是属于 避免，破坏环路等待条件是属于 预防，而剥夺资源是 解除 的基本方法。

三、应用题

36. 假设某系统中有 4 个资源（R1、R2、R3、R4），在某个时刻系统中共有 5 个进程，进程 P1，P2，P3，P4，P5 的最大资源需求数向量和此时已经分配到的资源数向量分别如表所示：

进程	当前已经分配到的资源	最大资源需求
P1	0, 0, 1, 2	0, 0, 1, 2

P2	2, 0, 0, 0	2, 7, 5, 0
P3	0, 0, 3, 4	6, 6, 5, 6
P4	2, 3, 5, 4	4, 3, 5, 6
P5	0, 3, 3, 2	0, 6, 5, 2

系统中当前可用资源向量为 (2, 1, 0, 0)，问：

- (1) 当前系统是否安全？
- (2) 如果进程 P3 发出资源请求向量 (0, 1, 0, 0)，系统能否将资源分配给它？

答：(1) 首先选择进程P1。

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 0, 3, 4	6, 6, 5, 6	6, 6, 2, 2	否
P4	2, 3, 5, 4	4, 3, 5, 6	2, 0, 0, 2	否
P5	0, 3, 3, 2	0, 6, 5, 2	0, 3, 2, 0	否

进程P1结束后，系统可用资源向量为 (2, 1, 1, 2)。此时可以选择进程P4。

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 0, 3, 4	6, 6, 5, 6	6, 6, 2, 2	否
P4				是
P5	0, 3, 3, 2	0, 6, 5, 2	0, 3, 2, 0	否

进程P4结束后，系统可用资源向量为 (4, 4, 6, 6)。此时可以选择进程P5。

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 0, 3, 4	6, 6, 5, 6	6, 6, 2, 2	否
P4				是
P5				是

进程P5结束后，系统可用资源向量为 (4, 7, 9, 8)。此时可以选择进程P2。

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2				是
P3	0, 0, 3, 4	6, 6, 5, 6	6, 6, 2, 2	否
P4				是

P5				是
----	--	--	--	---

进程P2结束后，系统可用资源向量为（6，7，9，8）。可以选择进程P3。

至此，所有进程都能完整运行完毕，因此该系统是安全状态。

(2) P3发出请求（0，1，0，0）后，系统可用资源向量为（2，0，0，0），各个进程资源向量变为：

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1	0, 0, 1, 2	0, 0, 1, 2	0, 0, 0, 0	否
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 1, 3, 4	6, 6, 5, 6	6, 5, 2, 2	否
P4	2, 3, 5, 4	4, 3, 5, 6	2, 0, 0, 2	否
P5	0, 3, 3, 2	0, 6, 5, 2	0, 3, 2, 0	否

首先进程P1上台：

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 1, 3, 4	6, 6, 5, 6	6, 5, 2, 2	否
P4	2, 3, 5, 4	4, 3, 5, 6	2, 0, 0, 2	否
P5	0, 3, 3, 2	0, 6, 5, 2	0, 3, 2, 0	否

进程P1结束后，系统可用资源向量为（2，0，1，2）。此时可以选择进程P4。

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 1, 3, 4	6, 6, 5, 6	6, 5, 2, 2	否
P4				是
P5	0, 3, 3, 2	0, 6, 5, 2	0, 3, 2, 0	否

进程P4结束后，系统可用资源向量为（4，3，6，6）。此时可以选择进程P5。

进程	当前已经分配到的资源	最大资源需求	目前所需资源	是否结束
P1				是
P2	2, 0, 0, 0	2, 7, 5, 0	0, 7, 5, 0	否
P3	0, 1, 3, 4	6, 6, 5, 6	6, 5, 2, 2	否
P4				是
P5				是

进程P5结束后，系统可用资源向量为（4，6，9，8）。此时没有可以选择的进程。

因此，系统不能分配资源给P3。