

22. (考研真题) 现代 OS 一般都提供多进程 (或称多任务) 运行环境, 回答以下问题。

(1) 为支持多进程的并发执行, 系统必须建立哪些关于进程的数据结构?

(2) 为支持进程状态的变迁, 系统至少应提供哪些进程控制原语?

(3) 在执行每一个进程控制原语时, 进程状态会发生什么变化? 相应的数据结构会发生什么变化?

【参考答案】(1) 为支持多进程的并发执行, 系统必须建立关于进程的相关数据结构, 包括PCB和队列结构 (如就绪队列、等待队列、运行指针等)。

(2) 为支持进程状态的变迁, 系统应提供的进程控制原语包括: 创建原语、阻塞原语、唤醒原语、撤销原语。

(3) 在执行每一个进程控制原语时, 进程状态及相应的数据结构有4种变化情况。①创建原语: 系统为进程创建PCB, 并对它进行初始化。进程状态由无变为就绪状态, 新创建的进程加入就绪队列中。②阻塞原语: 进程状态从运行状态变为阻塞状态, 并将阻塞进程的PCB插入相应的阻塞队列中。③唤醒原语: 进程状态从阻塞状态变为就绪状态, 从阻塞队列中删除该进程, 并将其插入就绪队列中。④撤销原语: 进程状态从运行状态变为消亡状态, 系统撤销该进程的PCB。

第3章 处理机调度与死锁

3.1 简答题参考答案

1. 高级调度与低级调度的主要任务是什么? 为什么要引入中级调度?

【参考答案】①高级调度的对象是作业。它的主要任务是根据某种算法, 决定将外存上处于后备队列中的哪几个作业调入内存, 为它们创建进程、分配必要的资源, 并将它们放入就绪队列。②低级调度的对象是进程 (或内核级线程)。它的主要任务是根据某种算法, 决定就绪队列中的哪个进程获得处理机, 并由分派程序将处理机分配给被选中的进程。

引入中级调度的主要目的是提高内存利用率和系统吞吐量。为此, 应把那些暂时不能运行的进程调至外存等待, 把进程状态改为就绪驻外存状态或挂起状态。当它们已具备运行条件且内存又有空间时, 由中级调度来决定把外存上的那些已具备运行条件的就绪进程再重新调入内存, 并修改它们的状态为就绪状态, 挂在就绪队列上等待。

2. 何谓作业和 JCB (job control block, 作业控制块) ?

【参考答案】①作业包含通常所说的程序和数据, 还配有作业说明书。系统根据该说明书对程序的运行进行控制。批处理系统中是以作业为基本单位将其从外存调入内存的。②JCB是作业在系统中存在的标志。为了管理和调度作业, 为每个作业设置一个JCB, 用于记录管理和调度作业所需的全部信息。

3. 在什么情况下需要使用 JCB ? 其中包含了哪些内容?

【参考答案】当一个作业进入系统时, 便由“作业注册”程序为该作业建立一个JCB, 再

根据作业类型，将其插入相应的后备队列中等待调度。调度程序依据一定的调度算法来调度它们，被调度到的作业将被装入内存。在作业运行期间，系统会按照JCB中的信息和作业说明书来对作业进行控制。当一个作业执行结束并进入完成状态时，系统负责回收已分配给它的资源，并撤销其JCB。

JCB包含的内容通常有作业标志、用户名称、用户账户、作业类型（CPU繁忙型、I/O繁忙型、批量型、终端型）、作业状态、调度信息（优先级、作业已运行）、资源要求、入系统时间、开始处理时间、作业完成时间、作业退出时间、资源使用情况等。

4. 在作业调度中应如何确定接纳多少个作业和接纳哪些作业？

【参考答案】作业调度每次接纳进入内存的作业数取决于多道程序度，即取决于系统允许多少个作业同时在内存中运行。应选择哪些作业从外存调入内存，取决于所采用的作业调度算法，最简单的是FCFS（first come first served，先来先服务）调度算法，较常用的是SJF（short job first，短作业优先）调度算法和基于优先级的调度算法。

5. 试说明低级调度的主要功能。

【参考答案】低级调度是在就绪队列中选择某个进程，由分派程序把CPU分配给该进程。低级调度的主要功能有：①保存当前进程的CPU现场信息；②按某种算法选择投入执行的新进程；③恢复新进程的CPU现场，从而将CPU分配给新进程。

6. （考研真题）简述引起进程调度的原因。

【参考答案】引起进程调度的常见原因有：①正在执行的进程正常终止或异常终止；②正在执行的进程因某种原因而阻塞，如提出I/O请求后被阻塞、在调用wait操作时因资源不足而阻塞、因其他原因执行block原语而阻塞等；③在引入时间片的系统中，时间片用完；④在抢占式调度方式中，就绪队列中某进程的优先级比当前正在执行进程的高，或者有优先级更高的进程进入就绪队列。

7. 在抢占式调度算法中，抢占的原则是什么？

【参考答案】抢占的原则有：①优先级原则，指允许优先级高的新到进程抢占当前进程的处理机；②短进程优先原则，指允许新到的短进程抢占当前长进程的处理机；③时间片原则，即各进程按时间片轮转运行时，当正在执行的进程的一个时间片用完后，便停止该进程的执行而重新进行调度。

8. 在选择调度方式和调度算法时，应遵循哪些准则？

【参考答案】一般而言，在设计一个OS时，应如何选择调度方式和调度算法，很大程度上取决于OS的类型及其设计目标。在批处理系统、分时系统和实时系统中，通常会采用不同的调度方式和调度算法。遵循的准则包括：①处理机调度算法的共同目标，即资源利用率高、保证公平性与平衡性以及策略强制执行；②批处理系统的目标，即平均周转时间短、系统吞吐量大、处理机利用率高；③分时系统的目标，即响应时间快、保证均衡性；④实时系统的目标，即保证截至时间和可预测性。

9. 何谓静态优先级和动态优先级？确定进程优先级的依据是什么？

【参考答案】①静态优先级是指在创建进程时确定的、在进程的整个运行期间保持不变的优先级。②动态优先级是指在创建进程之初，先赋予进程一个优先级，然后其值会随进程的推进或等待时间的增加而改变，如此而为目的的是获得更好的调度性能。此外，确定进程优先级

的依据有进程类型、进程对资源的需求和用户要求等。

10. 试比较 FCFS 和 SJF 这两种调度算法。

【参考答案】本题分步解答如下。

(1) 相同点：这两种调度算法都可以用于作业调度和进程调度。

(2) 不同点：FCFS调度算法每次都从后备队列中选择一个或多个最先进入该队列的作业，将它们调入内存、分配资源、创建进程、插入就绪队列。该算法有利于长作业/长进程，不利于短作业/短进程。SJF调度算法每次调度都从后备队列中选择一个或若干个估计运行时间最短的作业，将其调入内存运行。该算法有利于短作业/短进程，不利于长作业/长进程。

11. 在基于时间片的 RR (round robin, 轮转) 调度算法中, 应如何确定时间片的大小?

【参考答案】在RR调度算法中, 时间片的大小对系统性能有很大的影响。通常, 时间片应略大于一次典型的交互所需的时间, 使大多数交互式进程能在一个时间片内完成, 从而获得很小的响应时间。在确定时间片的大小时, 一般应考虑3个因素: 系统对相应时间的要求、就绪队列中进程的数目和系统的处理能力。

12. 为什么说多级反馈队列调度算法能较好地满足各方面用户的需求?

【参考答案】本题分步解答如下。

(1) 对于终端型作业用户, 其所提交的作业大都属于较小的交互型作业, 系统只要使这些作业在第1队列规定的时间片内完成, 终端型作业用户就会感到满足。

(2) 对于短批处理作业用户, 如果其作业在第1队列中执行一个时间片即可完成, 便可获得与终端作业一样的响应时间。对于稍长批处理作业用户, 其作业通常只须在第2队列和第3队列各执行1个时间片即可完成, 周转时间仍然较短。

(3) 对于长批处理作业用户, 其将依次在第1, 2, ..., n 个队列中运行, 然后再按轮转方式运行, 用户不必担心其作业长期得不到处理。因此, 多级反馈队列调度算法能满足多用户需求。

13. 为什么在实时系统中要求系统 (尤其是 CPU) 具有较强的处理能力?

【参考答案】在实时系统中都存在着若干个实时进程或任务, 它们用来反映或控制某个 (些) 外部事件, 往往带有某种程度的紧迫性, 因而对实时系统中的调度提出了某些特殊要求。若CPU的处理能力不够强, 则有可能会因CPU忙不过来而使某些实时任务得不到及时处理, 从而导致发生难以预料的后果。

14. 按照调度方式可将实时调度算法分为哪几种?

【参考答案】按照调度方式可将实时调度算法分为非抢占式调度算法和抢占式调度算法两种。①非抢占式调度算法, 包括非抢占式RR调度算法和非抢占式优先级调度算法; ②抢占式调度算法, 根据抢占时机的不同分为基于时钟中断的抢占式优先级调度算法和立即抢占的优先级调度算法。

15. 实时系统常用的调度算法有哪些? 请分别介绍它们。

【参考答案】实时系统的调度算法主要是基于任务开始/截止时间的EDF (earliest deadline first, 最早截止时间优先) 调度算法, 以及基于任务紧急/松弛程度的LLF (least laxity first, 最低松弛度优先) 调度算法。

(1) EDF调度算法。该算法是根据任务的开始/截止时间确定任务的优先级的, 截止时间越早则优先级越高。算法要求在系统中保持一个实时任务就绪队列, 该队列按各任务截止时间的

先后顺序进行排序。

(2) LLF调度算法。该算法是根据任务的紧急/松弛程度确定任务的优先级的。任务的紧急程度越高,赋予该任务的优先级就越高,以使其优先执行。例如,一个任务在200ms时必须完成,而它本身所需的运行时间就是100ms,因此,调度程序必须在100ms或之前调度执行,该任务的紧急程度(松弛程度)为100ms。

16. 在批处理系统、分时系统和实时系统中,各采用哪几种进程(作业)调度算法?

【参考答案】①在批处理系统中可采用的调度算法有SJF、优先级、高响应比优先、多级反馈队列等调度算法。②在分时系统中常用的调度算法是RR调度算法。③在实时系统中可采用的调度算法有EDF调度算法和LLF调度算法。

17. (考研真题)什么是死锁?产生死锁的原因和必要条件是什么?如何预防死锁?

【参考答案】死锁是指多个进程在运行过程中因争夺资源而造成的一种僵局,当进程处于这种僵持状态时,若无外力作用,它们都将无法向前推进。产生死锁的原因有:①竞争不可抢占性资源;②竞争可消耗资源;③进程间推进顺序不当。产生死锁必须同时具备4个必要条件:互斥条件、请求和保持条件、不可抢占条件、循环等待条件。

预防死锁是通过破坏产生死锁的4个必要条件中的一个或几个来实现的。其中互斥条件是设备固有属性,不能改变,因此主要破坏产生死锁的其他3个必要条件。①破坏“请求和保持”条件。当一个进程在请求资源时,它不能持有不可抢占性资源。②破坏“不可抢占”条件。当一个已经保持了某些不可抢占性资源的进程提出新的资源请求而不能得到满足时,它必须释放已经保持的所有资源,待以后需要时再重新申请。③破坏“循环等待”条件。对系统所有资源类型进行线性排序,并赋予它们不同的序号,规定每个进程必须按序号递增的方式请求资源。

18. 在解决死锁问题的几个方法中,哪个方法最易于实现?哪个方法可使资源利用率最高?

【参考答案】解决死锁的方法有预防死锁、避免死锁、检测和解除死锁,其中预防死锁这一方法最容易实现,但由于所施加的限制条件过于严格,资源利用率和系统吞吐量会降低;而检测和解除死锁这一方法可使系统获得较好的资源利用率和系统吞吐量。

3.2 计算题参考答案

19. (考研真题)有5个进程(见表1-3-1)需要调度执行,若采用非抢占式优先级(短进程优先)调度算法,问这5个进程的平均周转时间是多少?

表1-3-1 进程执行时间表

进程	到达时间	执行时间
P_1	0.0	9
P_2	0.4	4
P_3	1.0	1
P_4	5.5	4
P_5	7	2

【参考答案】本题分步解答如下。

(1) 采用非抢占式优先级(短进程优先)调度算法,各进程执行次序如图1-3-1所示。在0时刻,进程 P_1 到达, P_1 获得处理机。由于不可抢占,因此 P_1 会一直运行到9时刻结束;在9时刻,所有进程到达,根据短进程优先调度, P_3 获得处理机,然后是 P_5 ; P_2 和 P_4 的预计运行时间一样,按照FCFS原则, P_2 优先获得处理机,最后是 P_4 。

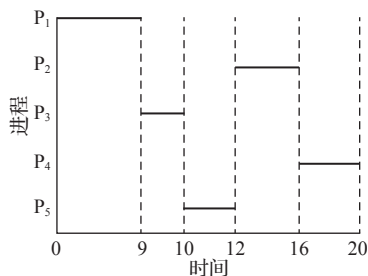


图 1-3-1 进程运行时间关系图

(2) 周转时间=完成时间-作业到达时间。因此 P_1 的周转时间是 $9-0=9$, P_2 的周转时间是 $16-0.4=15.6$, P_3 的周转时间是 $10-1=9$, P_4 的周转时间是 $20-5.5=14.5$, P_5 的周转时间是 $12-7=5$ 。这5个进程的平均周转时间为 $(9+15.6+9+14.5+5)/5=10.62$ 。

20. (考研真题) 假定要在一台处理机上执行表 1-3-2 所示的作业, 且假定这些作业在时刻 0 以 1, 2, 3, 4, 5 的顺序到达。请说明分别采用 FCFS、RR (时间片为 1)、SJF 及非抢占式优先级调度算法时, 这些作业的执行情况 (优先级的高低顺序依次为 1 到 5)。针对上述每种调度算法, 给出平均周转时间和平均带权周转时间。

表1-3-2 作业执行时间表

作业	执行时间	优先级
1	10	3
2	1	1
3	2	3
4	1	4
5	5	2

【参考答案】本题分步解答如下。

(1) 作业执行顺序如图1-3-2所示。

FCFS:

1	2	3	4	5
---	---	---	---	---

RR:

1	2	3	4	5	1	3	5	1	5	1	5	1	5	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

SJF:

2	4	3	5	1
---	---	---	---	---

非抢占式优先级:

2	5	1	3	4
---	---	---	---	---

图 1-3-2 作业执行顺序图

(2) 各个作业对应于各个算法的周转时间和带权周转时间见表1-3-3。

表1-3-3 作业运行情况比较表

算法	时间类型	P ₁	P ₂	P ₃	P ₄	P ₅	平均时间
	运行时间	10	1	2	1	5	3.8
FCFS	周转时间	10	11	13	14	19	13.4
	带权周转时间	1	11	6.5	14	3.8	7.26
RR	周转时间	19	2	7	4	14	9.2
	带权周转时间	1.9	2	3.5	4	2.8	2.84
SJF	周转时间	19	1	4	2	9	7
	带权周转时间	1.9	1	2	2	1.8	1.74
非抢占式 优先级	周转时间	16	1	18	19	6	12
	带权周转时间	1.6	1	9	19	1.2	6.36

因此，FCFS的平均周转时间为13.4，平均带权周转时间为7.26。RR的平均周转时间为9.2，平均带权周转时间为2.84。SJF的平均周转时间为7，平均带权周转时间为1.74。非抢占式优先级的平均周转时间为12，平均带权周转时间为6.36。注意：SJF的平均周转时间最短。

21. (考研真题) 将一组进程分为 4 类，如图 1-3-3 所示。各类进程之间采用优先级调度算法，而各类进程的 内部采用 RR 调度算法。请简述 P₁，P₂，P₃，P₄，P₅，P₆，P₇，P₈ 进程的调度过程。

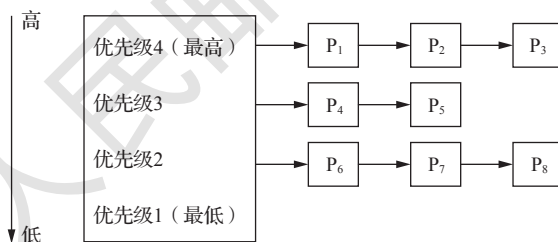


图 1-3-3 进程分类图

【参考答案】由于不同类进程间采用优先级调度算法，同类进程间采用RR调度算法，因此，系统首先对优先级为4的进程P₁、P₂、P₃采用RR调度算法进行运行；当P₁、P₂、P₃运行结束或阻塞时，再对优先级为3的进程P₄、P₅采用RR调度算法进行运行。在此期间，若P₁、P₂、P₃队列中有转为就绪状态的进程，则优先级3队列的当前时间片用完后回到优先级4队列进行进程调度。类似地，当进程P₁~P₅运行结束或阻塞时，对优先级为2的进程P₆、P₇、P₈采用RR调度算法进行运行，一旦进程P₁~P₅中有一个转为就绪状态，当前时间片用完后就立即回到相应的优先级队列进行RR调度。

22. 由 5 个进程组成进程集合 $P=\{P_0, P_1, P_2, P_3, P_4\}$ ，系统中有 3 类资源 A, B, C，假设在某时刻有表 1-3-4 所示的进程资源分配情况。

表1-3-4 进程资源分配情况

进程	Allocation	Max	Available
	A B C	A B C	A B C
P ₀	0 0 3	0 0 4	x y z
P ₁	1 0 0	1 7 5	
P ₂	1 3 5	2 3 5	
P ₃	0 0 2	0 6 4	
P ₄	0 0 1	0 6 5	

请问当 x, y, z 取下列值时，系统是否处于安全状态？

(1) 1, 4, 0; (2) 0, 6, 2; (3) 1, 1, 1; (4) 0, 4, 7。

【参考答案】由银行家算法中Need、Max、Allocation间的关系可得：

$$\text{Need} = \text{Max} - \text{Allocation} = \begin{bmatrix} 0 & 0 & 4 \\ 1 & 7 & 5 \\ 2 & 3 & 5 \\ 0 & 6 & 4 \\ 0 & 6 & 5 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 3 \\ 1 & 0 & 0 \\ 1 & 3 & 5 \\ 0 & 0 & 2 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 7 & 5 \\ 1 & 0 & 0 \\ 0 & 6 & 2 \\ 0 & 6 & 4 \end{bmatrix}。$$

(1) 当Available为(1, 4, 0)时，根据Need矩阵可知，可满足P₂的需求；P₂结束后释放资源，Available为(2, 7, 5)可满足P₀、P₁、P₃、P₄中任一进程的需求，因此系统不会出现死锁，处于安全状态。

(2) 当Available为(0, 6, 2)时，可满足进程P₀、P₃的需求；进程结束后释放资源，Available为(0, 6, 7)，可满足P₄的需求；P₄结束后释放资源，Available为(0, 6, 8)，此时不能满足其余任一进程的需求，系统出现死锁，因此当前处在非安全状态。

(3) 当Available为(1, 1, 1)时，可满足进程P₀、P₂的需求；这两个进程结束后释放资源，Available为(2, 4, 9)，此时不能满足其余任一进程的需求，系统出现死锁，处于非安全状态。

(4) 当Available为(0, 4, 7)时，可满足P₀的需求，进程结束后释放资源，Available为(0, 4, 10)，此时不能满足其余任一进程的需求，系统出现死锁，处于非安全状态。

3.3 综合应用题参考答案

23. (考研真题) 假设系统中有下述 3 种解决死锁的方法：

- (1) 银行家算法；
- (2) 检测死锁，终止处于死锁状态的进程，释放该进程所占有的资源；
- (3) 资源预分配。

简述上述哪种方法允许最大的并发性？请按“并发性”从大到小对上述3种方法进行排序。

【参考答案】①题中给出的3种方法中，检测死锁能允许更多的进程无须等待地向前推进，并发性最大。因为该方法允许进程最大限度地申请并分配资源，直至出现死锁，再由系统解决死锁。②银行家算法允许进程自由申请资源，只是在某个进程申请资源时会检查系统是否处于安全状态，若是，则可立即分配；若不是，则拒绝。其并发性的的大小次于检测死锁这一方法。③最后是资源预分配，因为此方法要求进程在运行之前申请所需的全部资源，这会使许多进程因申请不到全部资源而无法开始，得到部分资源的进程也会因未得到全部资源而不释放已占用的资源，所以此方法会导致资源浪费。因此，上述3种方法按并发性从大到小排序为：检测死锁、银行家算法、资源预分配。

24.（考研真题）某银行要实现一个电子转账系统，基本业务流程是：首先对转出方和转入方的账户进行加锁，然后办理转账业务，最后对转出方和转入方的账户进行解锁。若不采取任何措施，则系统会不会发生死锁？为什么？请设计一个能够避免死锁的方法。

【参考答案】本题分步解答如下。

（1）系统会发生死锁。原因如下：假如两个账号A和B有两个转账业务，分别是A转入B和B转入A。这两个业务在执行时可能会发生以下情况：一个业务锁定A账户，试图锁定B账户失败而等待；另一个业务锁定B账户，试图锁定A账户失败，也在等待，进而即会导致系统处于死锁状态。

（2）为了避免死锁，可采用两阶段加锁方法，即为每个账户设定一个唯一的账号，在业务执行前，必须按照账号的大小顺序依次获得所有账户的锁，业务完成后再依次按照锁定的先后次序将后锁定的账户先解锁。

25.（考研真题）设有进程 P_1 和进程 P_2 并发执行，它们都需要使用资源 R_1 和 R_2 ，使用资源情况如表 1-3-5 所示。

表1-3-5 进程使用资源情况

进程 P_1	进程 P_2
申请资源 R_1	申请资源 R_2
申请资源 R_2	申请资源 R_1
释放资源 R_1	释放资源 R_2

试判断是否会发生死锁，并解释和说明发生死锁的原因与必要条件。

【参考答案】这两个进程在不同的推进速度下，可能会产生死锁。比如：进程 P_1 先申请资源 R_1 ， P_1 得到 R_1 后进程 P_2 申请 R_2 ， P_2 得到 R_2 后， P_1 又申请资源 R_2 ，此时则因 R_2 已分配，故 P_1 阻塞。 P_1 和 P_2 两个进程因申请不到所需资源而形成死锁。如果改变进程的运行顺序，则这两个进程可能又不会发生死锁。因此，产生死锁的原因可归结为两点：①竞争资源；②进程推进顺序非法。产生死锁的必要条件：①互斥条件；②请求和保持条件；③不可抢占条件；④循环等待条件。