

第2章 进程的描述与控制

2.1 简答题参考答案

1. 什么是前趋图？请画出下列4条语句的前趋图。

$S_1: a=x+y; S_2: b=z+1; S_3: c=a-b; S_4: w=c+1;$

【参考答案】本题分步解答如下。

(1) 前趋图 (precedence graph) 是一个有向无环图，记为 DAG (directed acyclic graph)，用于描述进程间执行的前后关系。

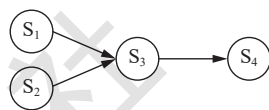


图 1-2-1 前趋图

(2) 题中4条语句对应的前趋图如图1-2-1所示。

2. 什么是进程？OS 中为什么要引入进程？它会产生什么样的影响？

【参考答案】①进程是一段可并发执行的具有独立功能的程序，是关于某个数据集的一次执行过程，也是OS进行资源分配和保护的基本单位。②在OS中引入进程，是为了实现多个程序的并发执行。传统的程序与其他程序并发执行时，执行结果不可再现，因此，传统的程序不能与其他程序并发执行，只有在为之创建进程后，其才能与其他程序（进程）并发执行。这是因为并发执行的程序“停停走走”地执行，只有在为其创建进程后，在它停下时，方能将其CPU现场信息保存在它的PCB (processing control block, 进程控制块) 中，待下次被调度执行时再从PCB中恢复CPU现场而继续执行，但传统的程序却无法违反上述要求。③建立进程所带来的好处是多个程序能并发执行，这极大地提高了资源利用率和系统吞吐量。但管理进程也须付出一定的代价，包括PCB及协调各运行机构所占用的内存空间开销，以及为进行进程间的切换、同步与通信等所付出的时间开销。

3. 进程最基本的状态有哪些？哪些事件可能会引起不同状态间的转换？

【参考答案】进程最基本的状态有3种。①运行态：进程占有处理机，正在运行。②就绪态：进程具备运行条件，等待系统分配处理机以便运行。③等待态（又称为阻塞态或睡眠态）：进程不具备运行条件，正在等待某个事件的完成。

进程不同状态间的转换及引发原因介绍如下。①运行态→等待态：等待使用资源或某事件发生；②等待态→就绪态：资源得到满足或某事件已经发生；③运行态→就绪态：运行时间片到达或出现有更高优先级的进程；④就绪态→运行态：CPU空闲时调度选中一个就绪进程需要其运行。

4. 为什么要引入进程的挂起状态？

【参考答案】所谓挂起状态，实际上就是一种静止的状态。一个进程被挂起后，不管它是否处于就绪状态，系统都不会分配给它处理机。因此，引入挂起状态是基于系统和用户的如下需要。①终端用户的需要：当终端用户在自己的程序运行期间发现问题时，希望暂停进程的运行。②父进程请求：父进程挂起自己的某个子进程，检查并修改该子进程，或者协调各子进程之间的活动。③负荷调节的需要：当实时系统中的工作负荷较重、实时任务受到影响时，挂起

一些不重要的进程。④OS的需要：OS挂起某些进程，检查或统计运行中的资源使用情况。

5. 叙述组成进程的基本要素，并说明它们的作用。

【参考答案】每个进程均由PCB、程序块、数据块和堆栈这4个基本要素组成。它们的作用分别描述如下。

(1) PCB：用来存储进程的标志信息、现场信息和控制信息。进程创建时建立PCB，进程撤销时回收PCB，其与进程一一对应。

(2) 程序块：即被执行的程序。它规定了进程一次运行应实现的功能，通常是纯代码，其作为一种系统资源可被多个进程共享。

(3) 数据块：即程序运行时须加工处理的对象，包括全局变量、局部变量和常量等的存放区以及开辟的工作区，常常为一个进程专用。

(4) 堆栈：每个进程都将捆绑一个堆栈，用来存储进程的标志信息、现场信息和控制信息。

6. (考研真题) 请给出 PCB 的主要内容。描述当进程状态发生转换(就绪→运行、运行→阻塞)时，OS 需要使用/修改 PCB 的哪些内容？

【参考答案】本题分步解答如下。

(1) PCB主要用于描述进程的基本情况以及进程的运行变化过程，是进程存在的唯一标志。PCB的内容可以分成调度信息和现场信息两部分。①调度信息供进程调度时使用，描述了进程当前所处的状况，包括进程名、进程号、存储信息、优先级、当前状态、资源清单、家族关系、消息队列指针、进程队列指针和当前打开文件的情况等。②现场信息用于保留运行进程发生状态转换时所要保存的CPU现场信息，以保证当该进程重新恢复运行时能恢复CPU现场，从断点处继续运行。常用的现场信息包括通用寄存器的内容、控制寄存器的内容、用户堆栈指针、系统堆栈指针等。

(2) OS需要使用/修改的PCB内容包括“就绪→运行”和“运行→就绪”。①就绪→运行是将PCB当前的就绪态改为运行态，修改PCB的队列指针，将PCB从就绪队列中移出等；利用PCB中的CPU现场信息，布置CPU现场并投入运行。②运行→就绪是将CPU的当前状态(各寄存器的内容)保存到PCB中，将进程状态由“运行”改为“就绪”，并修改PCB中相应的队列指针信息，然后将其放入PCB就绪队列中。

7. 试说明引起进程创建的主要事件。

【参考答案】为使程序之间能并发运行，应先为它们分别创建进程。导致一个进程去创建另一个进程的典型事件有4类：用户登录、作业调度、提供服务、应用请求。

8. (考研真题) 在创建一个进程时，OS 需要完成的主要工作是什么？

【参考答案】OS发现请求创建新进程事件后，首先，调用进程创建原语；其次，申请一个空白PCB，并向该PCB中填写用于控制和管理进程的信息；再次，为该进程分配运行时所需的资源；最后，把该进程的PCB转入就绪状态并插入就绪队列中。

9. 试说明引起进程终止的主要事件。

【参考答案】引起进程终止的主要事件介绍如下。①正常结束：指进程的任务已经完成，准备退出运行。②异常结束：指进程在运行时发生了某种异常事件，使程序无法继续运行。常见的异常事件包括越界错误、保护错、指令非法、特权指令错、运行超时、等待超时、算术运算错、I/O故障等。③外界干预：指进程应外界的请求而终止运行，这些干预有操作员干预、OS干预、父进程请求、因父进程终止等。

10. 在终止一个进程时，OS 要完成的主要工作是什么？

【参考答案】如果系统中发生了要求终止进程的事件，OS便会调用进程终止原语，按下述过程终止指定的进程：

- (1) 根据被终止进程标识符，从PCB集中检索出指定进程的PCB，读出该进程的状态；
- (2) 若被终止进程处于执行状态，则立即终止该进程的执行，置调度标志为真，用于指示该进程被终止后应重新进行调度；
- (3) 若该进程还有子孙进程，则应将其所有子孙进程也都予以终止，以防它们成为不可控的进程；
- (4) 将被终止进程拥有的全部资源或者归还给其父进程，或者归还给系统；
- (5) 将被终止进程的PCB从所在队列或链表中移出，等待其他程序来搜集信息。

11. 试说明引起进程阻塞或被唤醒的主要事件。

【参考答案】引起进程阻塞或被唤醒的主要事件介绍如下。①向系统请求共享资源失败。进程在向系统请求共享资源时，由于系统已无足够的资源分配给它，此时进程会因不能继续运行而转变为阻塞状态。②等待完成某种操作。当进程启动某种操作后，如果该进程必须在操作完成之后才能继续执行，则应先阻塞该进程。③尚未到达新数据。对于相互合作的进程，如果一个进程需要获得另一个进程提供的数据后才能处理该数据，但所需数据尚未到达，则进程阻塞。④等待到达新任务。在某些系统中，特别是在网络环境下的OS中，往往会设置一定量的系统进程，这种进程每当完成任务后便会把自己阻塞起来，等待新任务的到来。

12. 试比较进程间的低级与高级通信工具。

【参考答案】用户使用低级通信工具实现进程通信效率低，通信对用户不透明，所有操作都必须由程序员来实现；高级通信工具弥补了这些缺陷，用户可以直接利用OS提供的一组通信命令来高效地传送大量的数据。

13. 当前有哪几种高级通信机制？

【参考答案】高级通信机制可以归结为4类。①共享存储器系统通信机制：在共享存储器系统中，相互通信的进程共享某些数据结构或存储区，进程之间通过这些“空间”进行通信。②管道通信系统通信机制：发送进程和接收进程利用管道进行通信。③消息传递系统通信机制：进程不必借助任何共享存储区或数据结构，而是以格式化的消息（message）为单位，将通信的数据封装在消息中，并利用OS提供的一组通信命令在进程间进行消息传递，完成进程间的数据交换。④客户机-服务器系统通信机制：在网络环境的各种应用领域，客户机-服务器系统的通信机制已成为当前主流的通信实现机制。

14. 说明使用管道文件（pipe 文件）进行通信的优缺点。

【参考答案】在Linux、UNIX和OS/2等系统中都可以使用管道文件实现进程间的相互通信。其优点是可以在进程间不断地传递大量信息，且无须占用宝贵的内存空间。缺点是进程之间的信息传递要启动读/写磁盘，相对于消息缓冲队列而言通信速度较慢。另外，接收进程和发送进程间的同步也比较复杂，需要一定的时间开销。

15. 试比较直接通信方式和间接通信方式。

【参考答案】可以从以下几个方面来比较直接通信方式和间接通信方式。

- (1) 发送原语和接收原语。直接通信原语通常为send（receiver，message）、receive（sender，message）；间接通信原语通常为send（mailbox，message）、receive（mailbox，

message)，而且它还需要提供有关信箱创建与撤销的原语。

(2) 提供对方的标识符。直接通信要求发送双方显式地提供对方的标识符，对于接收进程，如果允许它同时接收多个进程发来的消息，则接收原语中的发送进程标识符可以是通信完成后返回的值；间接通信则不要求它们显式地提供对方的标识符，而只须提供信箱标识符。

(3) 通信链路。直接通信时，进程只须提供对方的标识符便可进行通信，在收发双方之间建立通信链路的任务由系统自动完成，此外，在收发双方之间有且仅有一条通信链路。间接通信时，仅当一对进程共享某个信箱时，它们之间才有通信链路；每对进程间可以有 multiple 链路。

(4) 实时性。直接通信通常只能提供实时的通信；间接通信既可实现实时通信，也可实现非实时通信。

16. 为什么要在 OS 中引入线程？

【参考答案】在OS中引入线程是为了减少进程在并发执行时所付出的时空开销，使OS具有更好的并发性，提高CPU的利用率。进程是分配资源的基本单位，而线程则是系统调度的基本单位。由于进程是资源的拥有者，因此在其创建、撤销、切换操作中需要较大的时空开销，限制了并发程度的进一步提高。为减少进程切换的开销，通常把进程作为资源分配单位和调度单位这两个属性进行分开处理，即进程还是作为资源分配的基本单位，但是不作为调度的基本单位（很少调度或切换），把调度执行与切换的责任交给“线程”。这样做的好处是不但可以提高系统的并发度，还能使系统适应新的对称多处理机环境，充分发挥其性能。

17. 试说明线程的属性。

【参考答案】线程是一个轻型实体，是可独立调度和分派的基本单位；线程可并发执行，可共享所属进程的资源。

18. 何谓用户级线程和内核支持线程？

【参考答案】①用户级线程是仅存在于用户空间中的线程，无须内核支持。这种线程的创建与撤销，线程间的同步与通信等功能，都无须利用系统调用实现。用户级线程的切换通常发生在一个应用进程的诸多线程之间，同样无须内核支持。②内核支持线程是在内核支持下运行的线程。无论是用户进程中的线程，还是系统中的线程，它们的创建、撤销和切换等操作都是依靠内核、在内核空间中实现的。在内核空间中还为每个内核支持线程设置了TCB（thread control block，线程控制块），内核根据该TCB感知某线程的存在并对其实施控制。

19. （考研真题）用户级线程和内核支持线程有何区别？

【参考答案】内核支持线程是在内核支持下实现的，即每个线程的TCB均设置在内核中，所有对线程的操作（如创建、撤销和切换等）都是通过系统功能调用由内核中的相应处理程序完成的。而用户级线程则仅存在于用户空间中，即每个线程的TCB均设置在用户空间中，所有对线程的操作也在用户空间中完成，而无须内核帮助。具体可从以下几个方面比较内核支持线程和用户级线程。①内核支持。用户级线程可在一个不支持线程的OS中实现，而内核支持线程则需要得到OS内核的支持。②处理机的分配。在多处机环境下，对纯粹的用户级线程来说，内核只为一个进程分配一个处理机，即进程无法享用多处理机带来的好处；而当设置有内核支持线程时，内核可调度一个应用中的多个线程同时在多个处理机上并行运行，从而提高程序的执行速度和效率。③调度和线程执行时间。对设置有内核支持线程的系统，内核的调度方式和调度算法与进程的十分相似，但调度的单位是线程；而对仅设置了用户级线程的系统，内核调度的单位为进程，当进程获得CPU时，隶属于该进程的多个线程可通过用户态下的线程调度分

享由内核分配给进程的CPU执行时间。因此，在相同条件下，内核支持线程通常会得到比用户级线程更多的CPU执行时间。④切换速度。用户级线程的切换通常发生在一个应用程序的各线程之间，由于无须陷入内核，而且切换的规则也相当简单，因此切换速度比内核支持线程至少快一个数量级。⑤系统调用。在典型的OS中，许多系统调用都会引起阻塞。当一个用户级线程执行这些系统调用时，被阻塞的是整个进程；而当一个内核支持线程执行这些系统调用时，则仅阻塞这一线程，但仍可调度其所属进程的其他线程执行。

20. 试说明用户级线程和内核支持线程的实现方法。

【参考答案】本题分步解答如下。

(1) 用户级线程在用户空间中实现，运行在“运行时系统”与“内核控制线程”的中间系统上。“运行时系统”用于管理和控制线程函数的集合。“内核控制线程”可通过系统调用获得内核提供的服务，将轻型进程（light weight process, LWP）作为中间系统。

(2) 在设置了内核支持线程的OS中，系统在创建新进程时，会为其分配一个任务数据区（per task data area, PTDA），其中包括若干个TCB空间。创建一个线程，分配一个TCB，将线程的有关信息写入TCB，并为线程分配必要的资源。当任务数据区中的TCB用完而进程又有新线程时，只要所创建的线程数目未超过系统允许值，系统即可再为之分配新的TCB。在撤销一个线程时，也应回收该线程的所有资源和TCB。

2.2 综合应用题参考答案

21. 试从调度、并发、拥有资源和系统开销这4个方面比较传统进程和线程。

【参考答案】传统进程与线程的比较如表1-2-1所示。

表1-2-1 传统进程与线程的比较

	调度	并发	拥有资源	系统开销
线程	线程是OS中调度和分派的基本单位，具有唯一的标识符和TCB	同一进程的多个线程可在一个/多个处理机上并发执行	同一进程的所有线程共享但不拥有进程的状态和资源，且驻留在进程的同一个内存地址空间中，可以访问相同的数据，通信和同步的实现也十分方便	作为系统调度和分派的基本单位，线程会被频繁地调度和切换；同一进程中的多线程共享同一地址空间，这能使线程实现快速切换
传统进程	传统进程具有独立的虚地址空间；以传统进程为单位进行任务调度时，系统必须切换地址空间，且切换时间长	许多“多任务OS”会限制用户拥有的最大进程数目，该数目对于许多并发应用来说是不够的	传统进程是系统中资源分配和保护的基本单位，也是系统调度的独立单位；每个传统进程都能以各自独立的速度在CPU上运行	对多个传统进程进行管理（如创建、调度、终止等）时，系统开销大，如在为了响应客户请求而须建立一个新的服务进程的服务器应用过程中，创建开销大

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