

文章编号: 1009-7171(2001)04-0038-03

# 进程的同步与互斥

## ——进程的同步与互斥学习

吕生荣

(集宁师专 计算机系 内蒙古 集宁 012000)

**摘要** 进程的同步与互斥是操作系统这门课的重点之一,也是一个难点。本文借教学中发现的问题,介绍了学习进程的同步与互斥时的几点体会。建议大家在掌握进程特点的基础之上来学习与掌握进程的同步与互斥关系。如:了解进程的并发性而了解同步与互斥关系的存在;掌握进程的动态性而掌握同步与互斥关系的不确定性等。

**关键词:** 多道程序设计 互斥进程 同步进程 作业 进程 P、V 操作

**中图分类号:** TP316

**文献标识码:** A

现代计算机一般都采用多道程序设计技术。其基本思想是主存中同时存放多个用户的作业(作业是指由计算机完成的一次计算。如:一次打印,程序的一次运行都可称为一个作业),使之处于运行状态,共享系统资源。每个用户的作业以及操作系统内部的某些程序模块都以进程的形式参与系统的并发执行,并发执行的诸程序之间,既有独立性,又有制约性。而相互制约的进程之间存在着某种关系——互斥关系或同步关系。一般的把具有互斥关系的进程互称互斥进程,具有同步关系的进程互称同步进程。

共享资源的两个或多个进程,为了确保运行结果的正确性,当其中的一个进程正使用共享资源时,不允许其它进程同时对该共享资源进行操作,这些进程之间的关系称为互斥关系。这当中的共享资源称为临界资源(即任何时刻只允许一个进程使用的资源,或者说资源数为1的资源)。并发进程中与共享变量有关的程序段称为临界区。比如:现在人们在火车站买车票,同时可以在不同的窗口买去同一地方的车票,我们认为是同时的。其实不然,实质是这样的:任何一个窗口售票时都要执行如下程序:

READ(X); //X是现有车票数//

IF X>=1 THEN X=X-1; //判断是否有车票,如有,卖出一张//

WRITE(X); //通知系统现有车票数//

这里X是各个窗口共同使用的,是共享变量,为了保证同一张车票售给一个用户,X也是临界资源,也就是说任何时刻只许一个窗口使用。这时,窗口之间的售票工作是互斥的。

而同步关系是指某一进程运行到某一点时,除非另一进程已完成某操作,否则就必须停下来等待这些操作结束。比如:在同一公共汽车上的司机和售票员是一组合作伙伴。司机可以看作是一个进程,其主要工作(或者说主要操作)是开车、到站停车;售票员可以看作是另一个进程,其主要工作(或者说主要操作)是售票、到站开门、关门。但司机与售票员必须互相协调,才能安全行驶。司机停车后,售票员开门,售票员关门后,司机开车。必须按这一顺序进行操作。而且只能按这一顺序进行操作。这时司机停车与售票员开门、售票员关门与司机开车之间就是两对同步关系。也就是说售票员要开门,就必须等司机停车后才能进行,否则不能开门而必须等待;司机要开车,也必须等待售票员关门后才能进行,否则也不能开车而必须等待。

但我发现,无论是我学习时,还是学生学习

时总存在一种潜意识,那就是总要确定进程间的关系,即进程间要么存在同步关系,要么存在互斥关系,而且一旦把关系确定下来就默认它是永久性成立的。其实这种关系划分是不必要的,甚至是错误的。那么该如何认知进程间的关系呢?我以为从基本概念出发,然后结合进程的特征来掌握同步与互斥。

进程的概念前面已经介绍过了,进程的特征是指进程的动态性,并发执行性,相互独立性以及结构性。

动态性是进程的最本质属性。进程是程序的一次执行过程,具体地说当要完成某项工作时,就“创建”一个进程,并由处理机调度执行这个程序所对应的进程,完成该项工作后,再由系统“撤消”这个进程。由此可见,进程是有生存期的。它随时产生,随时消亡。假定某时系统中存在 A、B 两个互斥进程,但随着时间的推移,A 或 B 可能结束,而被撤消。这时 A 或 B 进程在系统中已不复存在,当然就不能再说 A 与 B 是互斥的。所以说进程的同步或互斥只在某一时刻成立,之后,他们的关系随着进程的撤消而消失。此外,进程的同步与互斥关系还会随着时间的改变而改变。关于这一点,可以通过信号量及其上的 P、V 操作来说明。

信号量 S 是一个记录型数据结构,包括两个数据项,一个是整型变量 VALUE,表示资源的数量;另一个是指针 POINTER,指向信号量上等待进程队列的队首指针。P 操作的功能如下:

```
P(S)
BEGIN
    关中断;
    S.VALUE=S.VALUE-1;
    IF S.VALUE<0 THEN
    BEGIN
        置为等待进程;
        插入进程的信号量 S. PIONTER 所指的队列中;
        开中断;
    转进程调度;
    END
    开中断;
```

END

V 操作的功能如下:

```
V(S)
BEGIN
    关中断;
    S.VALUE=S.VALUE+1;
    IF S.VALUE<=0 THEN
    BEGIN
        从 S 的等待队列中摘下一个进程;
        置该进程为就绪状态并加入到就绪队列中;
        就绪进程数加 1;
    END
    开中断;
END
```

信号量及其上的 P、V 操作是一种典型的实现进程同步与互斥的方法。在某一进程执行过程中根据情况随时可以调用 P、V 操作。在实现互斥时(因为临界资源任何时刻只允许一个进程使用,所以我们可以理解成资源数是 1),信号量 S.VALUE 的初值为 1,当进程对临界资源进行操作时,首先检测该临界资源是否被占用,然后确定自己能否运行。这一过程是由 P 实现的,具体如下:首先信号量 S.VALUE 减 1,然后判断 S.VALUE 的值,如果 S.VALUE 小于 0,则该进程所申请的资源已被占用,因此该进程因为资源得不到满足必须等待。否则,该进程得到资源可以继续运行。当它退出临界区(对临界资源进行操作的工作区)时,就必须释放临界资源,让其它进程使用。释放资源是由 V 操作实现的。具体操作如下:首先信号量 S.VALUE 的值加 1,然后判断它的值,如果小于 0,则说明某进程正在使用这一资源时,其它进程申请过该资源,也就是说,其它进程调用过该信号量上的 P 操作,具体调用过几次,就在信号量 S.VALUE 上减过几次 1,信号量 S.VALUE 的值就是负几。这几个进程因得不到资源而位于等待队列上。这时应由 V 操作从等待队列中选择一个进程,占用它所释放的临界资源,从而处于就绪状态,随时准备占用 CPU,否则,V 结束操作。

实现同步时,信号量 S.VALUE 的初值为 0(初始时,进程运行的条件还不满足,或者说申

请的资源没有)仍用 P、V 操作来协调它们的关系。下面来分析在 P、V 控制下进程之间关系的转化。

假设有 A、B、C 三个进程,它们具有并发关系,在运行过程中都要使用打印机。而打印机的工作特点决定了它只有输出某一进程的所有信息后,才能输出另一个进程的信息。这时打印机是共享资源,而且它的工作特点决定了它也是临界资源。开始 A、B、C 同时要使用打印机,但它只能分配给其中的一个进程,所以 A、B、C 三个进程之间的关系是互斥关系。S.VALUE 的值为 1。打印机最终必将分配给某一进程,并且谁先调用 P(S)谁得到它。假设 A 是先得到这一资源的进程,这时 S.VALUE 的值已经是 0,那么 B、C 如果要输出信息,必须申请打印机而执行 P(S),P(S)把 B、C 转为等待状态。这时 A 与 B、C 间的关系是同步关系。当 A 释放打印机后(即执行 V(S)后),A 与 B、C 之间只存在并发关系,而 B、C 是互斥关系。与上述相同,一旦当中的进程 B 或 C 占用了资源(执行了 P(S)),C 或 B 又要申请打印机而执行 P(S)转为等待状态,这时 B、C 又成了同步关系。之后这种关系也必然消失。

通过上面的叙述,我们可以很明确地看到,同样是使用某一资源的进程,但随着时间的改变,进程间的关系也会发生改变,这一点不仅符合进程的动态性特征,也说明了进程间的同步与互斥关系的动态性,是一个与时间密切相关的概念。

进程的并发性是引起同步与互斥的直接原因。并发的进程具有独立性和异步性,所谓独立性是指进程是系统进行资源分配和调度运行的单位。而异步性则是指某个进程在系统中都以相互独立的不可预知的速度执行。这些进程走走停停直到最后结束。进程在系统内部的状态要由同步设施协调。进程的另一特性就是结构性,即进程由程序段、需要加工的数据段以及状态信息三部分组成。

通过进程的结构性,我们可以认识进程的同步与互斥关系的另一特点。进程的运行就是对应程序的执行。在程序的执行过程中,当对临界资源进行操作时,该进程与其它同时要对该临界资源进行操作的进程才能存在互斥关系。否则互斥关系并不成立,这些进程仍能并发运行。同步关系也是如此:若干个进程在并发执行的过程中,只有到达某一点时,同步关系才成立。从这里可以看出互斥与同步又是一个局部概念,受所执行的程序段限制。

进程是操作系统中最重要概念之一,同步与互斥又是操作系统中最关键的问题,所以要正确的认识这两种关系。在学习进程的互斥与同步时,抓住进程的动态性,掌握同步与互斥的动态性(同步与互斥关系是随着时间的推移而改变的,并不是一层不变的)。通过进程的结构性,了解进程的同步关系与互斥关系的局部性(比如说进程的互斥不是整个进程的操作都互斥,只在临界区是互斥的)。只有认识到这两个特点才能正确地掌握进程的同步与互斥。

#### 参考文献:

- [1] 尹传高,杨跃武.操作系统[M].北京:电子工业出版社.第 27、124 页
- [2] 陆丽娜,齐勇,白恩华.计算机操作系统原理与技术(上册)[M].西安交通大学出版社.第 45、46 页
- [3] 朱继生,宗大华,周虹等.最新操作系统教程[M].北京:电子工业出版社.第 126、130 页.

(上接 35 页)

#### 参考文献:

- [1] 刘智敏.误差与数据处理[M].北京:原子能出版社,1981.
- [2] 吴筑筑,谭信民,邓秀勤.计算方法[M].北京:电子工业出版社,1998.
- [3] 徐卒葳.计算方法引论[M].北京:高等教育出版社,1985.