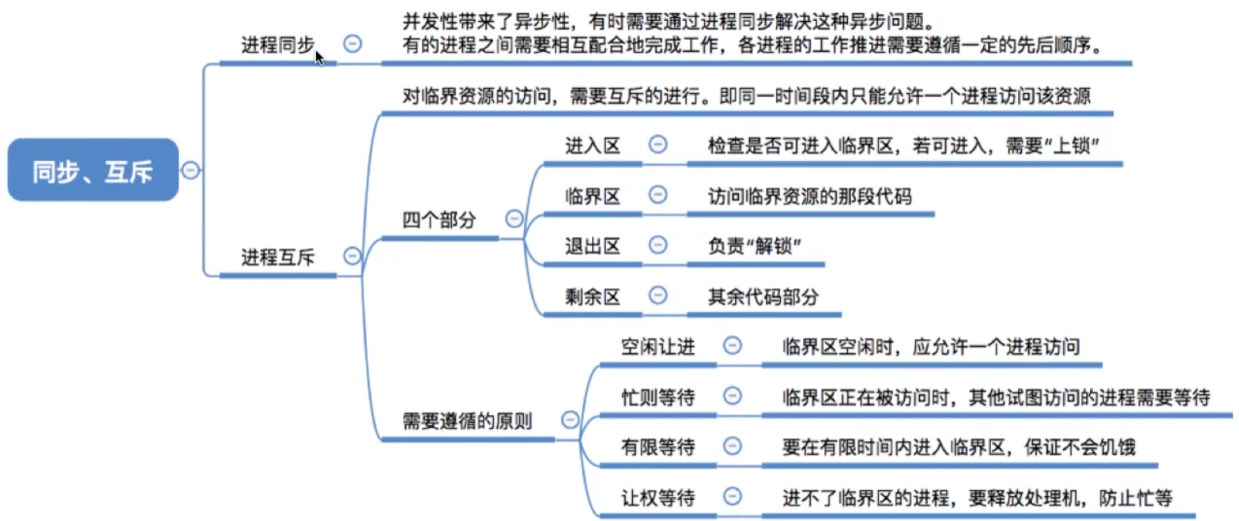


# 同步与互斥的基本概念



## 什么是进程同步

进程具有异步性的特征。异步性是指，各并发执行的进程以各自独立的、不可预知的速度向前推进。

进程通信——管道通信



读进程和写进程并发地运行，由于并发必然导致异步性，因此“写数据”和读数据两个操作执行的先后顺序是不确定的。而实际应用中，又必须按照“写数据->读数据”的顺序来执行。

如何解决这种异步问题，就是“进程同步”所讨论的内容。

操作系统要提供“进程同步机制”来实现上述需求。

同步亦称直接制约关系，它是指为完成某种任务而建立的两个或多个进程，这些进程因为需要在某些位置上协调它们的工作次序而产生的制约关系。进程间的直接制约关系就是源于它们之间的相互合作。

## 什么是进程互斥

进程的“并发”需要“共享”的支持。各个并发执行的进程不可避免的需要共享一些系统资源（比如内存，又比如打印机、摄像头这样的I/O设备）

我们把一个时间段内只允许一个进程使用的资源称为临界资源。许多物理设备（比如摄像头、打印机）都属于临界资源。此外还有许多变量、数据、内存缓冲区等都属于临界资源。

对临界资源的访问，必须互斥地进行。互斥，亦称间接制约关系。进程互斥指当一个进程访问某临界资源时，另一个想要访问该临界资源的进程必须等待。当前访问临界资源的进程访问结束，释放该资源之后，另一个进程才能去访问临界资源。

对临界资源的互斥访问，可以在逻辑上分为如下四个部分：

```
do{
    entry section;    //进入区
    critical section; //临界区
    exit section;     //退出区
    remainder section; //剩余区
}while(true)
```

- 进入区

负责检查是否可进入临界区，若可进入，则应设置正在访问临界资源的标志（可理解为“上锁”），以阻止其他进程同时进入临界区

- 临界区

访问临界资源的那段代码

- 退出区

负责解除正在访问临界资源的标志（可理解为“解锁”）

- 剩余区

做其他处理

注意：

临界区是进程中访问临界资源的代码段。

进入区和退出区是负责实现互斥的代码段。

临界区也可称为“临界段”。

如果一个进程暂时不能进入临界区，那么该进程是否应该一直占着处理机？该进程有没有可能一直进不了临界区？

为了实现对临界资源的互斥访问，同时保证系统整体性能，需要遵循以下原则：

1. 空闲让进。临界区空闲时，可以允许一个请求进入临界区的进程立即进入临界区；
2. 忙则等待。当已有进程进入临界区时，其他试图进入临界区的进程必须等待；
3. 有限等待。对请求访问的进程，应保证能在有限时间内进入临界区（保证不会饥饿）；
4. 让权等待。当进程不能进入临界区时，应立即释放处理机，防止进程忙等待。