**线程之间的同步与互斥**

多线程共享同一个进程的地址空间；优点是：线程间很容易通信，可以通过全局变量实现数据的共享和交换；缺点是：多个线程同时访问共享对象时需要引入同步和互斥机制。

这里主要介绍Posix中的两种线程同步机制，分别为信号量和互斥锁。这两个同步机制可以通过互相调用对方来实现，但互斥锁更适用于同时可用的资源是唯一的情况；信号量更适用于同时可用的资源为多个的情况。

信号量线程控制：

信号量：

信号量代表某一类资源，其值表示系统中该资源的数量；本质上，信号量是一个非负的整数计数器，它被用来控制对公共资源的访问。

信号量是一个受保护的变量，只能通过以下三种操作来访问：

1. 初始化(sem\_init)；
2. P操作(sem\_wait)；
3. V操作(sem\_post)。

PV原子操作是对整数计数器信号量sem的操作：

1. P(S)含义：if(信号量的值>0) {申请资源的任务继续运行；

信号量的值减一}

else {申请资源的任务阻塞；}

1. V(S)含义：if(没有任务在等待该资源) {信号量的值加一；}

else {唤醒第一个等待的任务，使其继续运行；}

PV原子操作主要用于进程或线程间的同步和互斥这两种典型的情况。

1. 若用于互斥，几个进程或线程往往只设置一个信号量；
2. 当信号量用于同步操作时，往往会设置多个信号量，并安排不同的初始值来实现它们之间的顺序执行。

sem\_init()：

sem\_init()用于创建一个信号量，并初始化它的值。

|  |  |
| --- | --- |
| 头文件 | #include <semaphore.h> |
| 原型 | int sem\_init(sem\_t \*sem, int pshared, unsigned int value); |
| 参数 | sem：信号量指针 |
| pshared：信号量共享的范围(0：表示线程间使用) |
| value：信号量初值 |
| 返回值 | 成功：0 |
| 失败：-1(设置errno) |

sem\_wait()：

sem\_wait()即相当于P(S)操作，当无资源可申请时，会使调用线程阻塞(睡眠)。

|  |  |
| --- | --- |
| 头文件 | #include <semaphore.h> |
| 原型 | int sem\_wait(sem\_t \*sem); |
| 参数 | sem：信号量指针 |
| 返回值 | 成功：0 |
| 失败：-1(设置errno) |

sem\_post()：

sem\_post()即相当于V(S)操作。

|  |  |
| --- | --- |
| 头文件 | #include <semaphore.h> |
| 原型 | int sem\_post(sem\_t \*sem); |
| 参数 | sem：信号量指针 |
| 返回值 | 成功：0 |
| 失败：-1(设置errno) |

互斥锁线程控制：

互斥锁主要用来保护临界资源。每个临界资源都有一个互斥锁来保护，任何时刻最多只能有一个线程能访问该资源。

线程必须先获得互斥锁才能访问临界资源，访问完资源后释放该锁。如果无法获得锁，线程就会阻塞直到获得锁为止。

互斥锁机制主要包括以下基本操作：

pthread\_mutex\_init()：

初始化互斥锁，初始化后，资源默认为可用状态。

|  |  |
| --- | --- |
| 头文件 | #include <pthread.h> |
| 原型 | int pthread\_mutex\_init(pthread\_mutex\_t \*mutex, pthreadmutexattr\_t \*attr); |
| 参数 | mutex：互斥锁 |
| attr：互斥锁属性(NULL表示使用缺省属性) |
| 返回值 | 成功：0 |
| 失败：-1 |

pthread\_mutex\_lock()：

上锁，即相当于P(S)操作。

|  |  |
| --- | --- |
| 头文件 | #include <pthread.h> |
| 原型 | int pthread\_mutex\_lock(pthread\_mutex\_t \*mutex); |
| 参数 | mutex：互斥锁 |
| 返回值 | 成功：0 |
| 失败：-1 |

pthread\_mutex\_unlock()：

解锁，即相当于V(S)操作。

|  |  |
| --- | --- |
| 头文件 | #include <pthread.h> |
| 原型 | int pthread\_mutex\_unlock(pthread\_mutex\_t \*mutex); |
| 参数 | mutex：互斥锁 |
| 返回值 | 成功：0 |
| 失败：-1 |