南京大學
NANJING UNIVERSITY

计算机与操作系统
第十一讲 并发程序设计
习题讲解

南京大学软件学院

16:03

8、独木桥问题1

- 独木桥问题1：东西向汽车过独木桥，为了保证安全，只要桥上无车，则允许一方的汽车过桥，待一方的车全部过完后，另一方的车才允许过桥。请用信号量和PV操作写出过独木桥问题的同步算法。

独木桥问题1

```
var wait,mutex1,mutex2:semaphore;
mutex1:=mutex2:=1;wait:=1;
counter1,counter2:integer;
```

```
process P东(){

   P(mutex1);
     count1++;
     if (count1==1) P(wait);
   V(mutex1);
     过独木桥;
   P(mutex1);
     count1--;
     if (count1=0) V(wait);
  V(mutex1);

}
```

```
process P西(){

   P(mutex2);
     count2++;
     if (count1==1) P(wait);
   V(mutex2);
     过独木桥;
   P(mutex1);
    count2--;
    if (count1==0) V(wait);
  V(mutex2);

}
```

独木桥问题 2

- 独木桥问题2：在独木桥问题1中，限制桥面上最多可以有k辆汽车通过。试用信号量和P，V操作写出过独木桥问题的同步算法

独木桥问题2

```
semaphore wait,mutex1,mutex2,bridge;
 mutex1=mutex2=1;bridge=k;wait=1;
 int count1,count2; count1=0;count2=0;
```

```
process P东( ){
     P(mutex1);
       count1++;
       if (count1==1)  P(wait);
     V(mutex1);
     P(bridge);
     {过桥};
     V(bridge);
     P(mutex1);
       count1--;
       if (count1==0)  V(wait);
     V(mutex1);
   }
```

```
process P西( ){
     P(mutex2);
       count2++;
       if (count2==1)  P(wait);
     V(mutex2);
     P(bridge);
      {过桥};
      V(bridge);
     P(mutex2);
       count2--;
      if (count2==0)  V(wait);
     V(mutex2);
     }
```

独木桥问题3

- 独木桥问题3：在独木桥问题1中，以叁辆汽车为一组，要求保证东方和西方以组为单位交替通过汽车。试用信号量和P，V操作写出汽车过独木桥问题的同步算法

独木桥问题3

```
semaphore wait,mutex1,mutex2;
mutex1=mutex2=1;wait=1;
int counter1,counter2; counteru1=0; countd1=0; counteru2=0; counterd2=0;
semaphore S1,S2;S1=3;S2=0;
```

```
Process P东(){

  P(S1)
    P(mutex1);
      countu1++;
      if (countu1==1) P(wait);
    V(mutex1);
      过独木桥;
  V(S2)
   P(mutex1);
    countu1--;    countd1++
    if ((countu1==0)&(countd1==3))
     {countd1=0;   V(wait); }
   V(mutex1);

}
```

```
Process P西(){

  P(S2)
    P(mutex2);
      countu2++;
      if (countu2==1) P(wait);
    V(mutex2);
      过独木桥;
  V(S1)
    P(mutex2);
     countu2--;  countd2++
    if ((countu2==0)&(countd2==3))
     {countd2=0;    V(wait); }
    V(mutex2);

}
```

独木桥问题4

- 独木桥问题4：在独木桥问题1中，要求各方向的汽车串行过桥，但当另一方提出过桥时，应能阻止对方未上桥的后继车辆，待桥面上的汽车过完桥后，另一方的汽车开始过桥。试用信号量和P，V操作写出过独木桥问题的同步算法

独木桥问题4

```
semaphore stop,wait,mutex1,mutex2;
stop=mutex1=mutex2=1;wait=1;
int count1,count2; count1=0;count2=0;
```

```
process P东( ) {
  P(stop);
    P(mutex1);
      count1++;
      if (count1==1)  P(wait);
    V(mutex1);
  V(stop);
    {过桥};
   P(mutex1);
      Count1--;
     if (count1==0)  V(wait);
    V(mutex1);
  }
```

```
process P西( ) {
   P(stop);
    P(mutex2);
      count2++;
      if (count2==1)  P(wait);
    V(mutex2);
   V(stop);
    {过桥};
     P(mutex2);
       count2--;
       if (count2==0)  V(wait);
    V(mutex2);
    }
```