### 1.生产者-消费者（缓冲区问题）

生产者进程结构：

do{

wait(empty) ;

wait(mutex) ;

add nextp to buffer

signal(mutex) ;

signal(full) ;

}while(1) ;

消费者进程结构：

do{

wait(full) ;

wait(mutex) ;

remove an item from buffer to nextp

signal(mutex) ;

signal(empty) ;

}while(1) ;

2.桌上有一空盘，允许存放一只水果。爸爸可向盘中放苹果，也可向盘中放桔子，儿子专等吃盘中的桔子，女儿专等吃盘中的苹果。规定当盘空时一次只能放一只水果供吃者取用，请用P、V原语实现爸爸、儿子、女儿三个并发进程的同步。

**解：**设置三个信号量S、So、Sa，信号量S表示盘子是否为空，其初值为l；信号量So表示盘中是否有桔子，其初值为0；信号量Sa表示盘中是否有苹果，其初值为0。  
int S＝1;  
int Sa＝0;  
int So＝0;  
      main()  
      {  
        cobegin  
            father();      /\*父亲进程\*/  
            son();        /\*儿子进程\*/  
            daughter();    /\*女儿进程\*/  
        coend  
    ｝  
    father()  
    {  
        while(1)  
          {  
            P(S);  
            将水果放入盘中;  
            if（放入的是桔子）V(So);  
            else  V(Sa);  
           }  
     }  
    son()  
    {  
        while(1)  
          {  
             P(So);  
             从盘中取出桔子;  
             V(S);  
             吃桔子;  
            ｝  
    }  
    daughter()  
    {  
         while(1)  
            {  
              P(Sa);  
              从盘中取出苹果;  
              V(S);  
              吃苹果;  
            ｝  
｝