



# 临界区问题之 Peterson 算法（双进 程）

# Peterson 算法

- ◆两进程共享数据
  - ◆int **turn**;
  - ◆Boolean **flag**[2]
- ◆变量 **turn**(只能取 i 或 j) 指示两个进程中的哪一个可以进入临界区
- ◆数组 **flag** 指示进程是否申请进入临界区。也就是说, **flag[i] = true** 表示进程  $P_i$  申请进入临界区

# Peterson 算法定义进程 $P_i$

```
while (true) {  
    flag[i] = TRUE;  
    turn = j;  
    while ( flag[j] && turn == j);  
    CRITICAL SECTION  
    flag[i] = FALSE;  
    REMAINDER SECTION  
}
```

# Peterson 算法定义进程 $P_j$

```
while (true) {  
    flag[j] = TRUE;  
    turn = i;  
    while ( flag[i] && turn == i);  
        CRITICAL SECTION  
    flag[j] = FALSE;  
        REMAINDER SECTION  
}
```

# Peterson 算法满足：

- ◆ “互斥” (Mutual Exclusion) 。为什么？
- ◆ “空闲让进” (Progress) 。为什么？
- ◆ “有限等待” (Bounded Waiting) 。为什么？



**End**