

## 专题一 进程同步

复习记录型信号量机制，复习生产者-消费者问题、读者-写者问题

1. 设有两个进程 P、Q，Q 的优先级高于 P，同时进入就绪队列。各自运行的程序段如下所示：

其中 S1、S2 为信号量，初值为 0；已知 X、Y、A 为共享变量，X 的初值为 83，Y 初值为 35，A 初值为 38；若调度程序执行的策略为抢占式优先权原则，试问：

(1) (5 分) 执行序列 (用代码  $P_i$ ,  $Q_i$  表示,  $i=1, 2, \dots, 6$ )

(2) (5 分) 变量 X、Y、A 的运行结果 (略)

进程 P

进程 Q

P1 Y=12

Q1 X=18

P2 Y=A+Y

Q2 A=X+A

P3 P (S1)

Q3 V (S1)

P4 A=Y+X

Q4 X=A+Y

P5 V (S2)

Q5 P (S2)

P6 X=Y+A

Q6 A=X-Y

解：Q1Q2Q3Q4Q5P1P2P3P4P5Q6P6

说明：题目中也可能采用非抢占式；或者优先级相同，进入就绪队列有先后次序。

2. 今有一文件 F 供 A、B 两组进程共享，规定同组的进程可以同时读文件 F；但当有 A 组 (或 B 组) 的进程在读文件 F 时就不允许 B 组 (或 A 组) 的进程读文件。

定义整型计数器变量 C1 和 C2, 分别记录 A 组和 B 组中读文件 F 的进程数, 初值均为 0. 设置 3 个初值均为 1 的信号量 S1、S2 和 SAB。

进程 A<sub>i</sub>:

\_\_P (S1) \_\_\_\_\_;

将 C1 加 1;

若 C1=1 则\_\_P (SAB) \_\_\_\_\_;

\_V (S1) \_\_\_\_\_;

读文件 F;

\_\_P (S1) \_\_\_\_\_;

将 C1 减 1;

若 C1=0 则\_\_V (SAB) \_\_\_\_\_;

V (S1) \_\_\_\_\_;

进程 B<sub>i</sub>: (类似于进程 A<sub>i</sub>, 请自行填写)

\_\_\_\_\_;

将 C2 加 1;

若 C2=1 则\_\_\_\_\_;

\_\_\_\_\_;

读文件 F;

\_\_\_\_\_;

将 C2 减 1;

若  $C2=0$  则\_\_\_\_\_;

\_\_\_\_\_;

要求: (1) 说明信号量  $S1$ 、 $S2$ 、 $SAB$  的作用;

(2) 填上合适的 P、V 操作

分析: 该问题为读者问题

解: (1)  $S1$  用于实现进程  $Ai$  互斥访问  $C1$  计数器, 初值为 1;  $S2$  用于实现进程  $Bi$  互斥访问  $C2$  计数器, 初值为 1;  $SAB$  用于实现 A 组与 B 组互斥地访问共享文件 F

(2) 请参见题干答案

3. 假定有三个并发进程  $R$ 、 $W1$ 、 $W2$  共享一个缓冲器  $B$ ,  $B$  中每次只允许存放一个数。当  $B$  中无数时,  $R$  可以从输入设备上读入数据并将数据存放到  $B$  中。若此数为偶数, 则允许  $W1$  将其取出打印, 否则  $W2$  将其取出打印。进程  $W1$  或  $W2$  对每次存入缓冲区的数据只能打印一次。 $W1$ 、 $W2$  都不能从空的缓冲区中取数。试用信号量及 PV 原语完成  $R$ 、 $W1$ 、 $W2$  的同步操作。(定义信号量时应说明其意义及初值)

分析: 该问题是典型的生产者-消费者问题, 一个生产者、两个消费者

解:  $S$ : 表示是否可以将数放入缓冲区  $B$ , 初值为 1, 表示可以放数;

$S2$ : 表示缓冲区中是否有奇数, 初值为 0, 表示无奇数;

S1: 表示缓冲区中是否有偶数，初值为 0，表示无偶数；

Begin

S, S1, S2: semaphore;

S=1; S1=0; S2=0;

Cobegin

Process R

Repeat (C 语言中: while(1))

从设备上读入一个数据;

P (S);

如果是偶数, V (S1); 否则, V (S2)

Until false

Process W1

Repeat

P(S1);

从缓冲区中取偶数;

V (S);

打印;

Until false

Process W2

Repeat

P(S2);

从缓冲区中取奇数;

V (S);

打印;

Until false

coend

End

4. ( 2007 年 专 升 本 )

生产围棋的工人不小心把相等数量的黑子和白子混合装在一个盒子里，现在要用自动分拣系统把黑子和白子分开，该系统由两个并发执行的进程 PA 和 PB 组成，系统功能如下：

PA 专拣黑子，PB 专拣白子；

每个进程每次只拣一个子，当一个进程拣子时，不允许另一个进程去拣子；

当一个进程拣了子（黑子或白子）后，必须让另一个进程去拣一个（白子或黑子）。

请回答：写出用 PV 操作时应定义的信号量和初值；

根据定义的信号量，写出用 PV 操作管理两个并发进程的程序。

分析：类似单缓冲的生产者-消费者问题

解：假设先让 PA 拣黑子，设两个信号量 S1，S2，初值为 1 和 0

PA

Repeat

P (S1);

拣黑子;

V (S2);

Until false

PB

Repeat

P (S2);

捡白子;

V (S1);

Until false

#### 5. (2008 年专升本)

某高校机房面向全校师生开放, 机房有 40 台计算机供学生上机使用。

如果机房有空机位的话, 要上机的学生在登记簿上登记班级、姓名等信息, 则可上机; 否则的话在门外等待。学生上机结束后, 在登记簿上登记离开的时间。使用 PV 操作描述进程的并发过程。

分析: 类似于读者问题。

解: mutex: 互斥信号量, 用于实现学生互斥地使用登记簿, 初值为 1, ;

S: 资源信号量, 用于表示计算机的数量, 初值为 40

Studenti

P (S);

P (mutex);

登记上机信息;

V (mutex);

上机;

.

.

V (S);

下机;

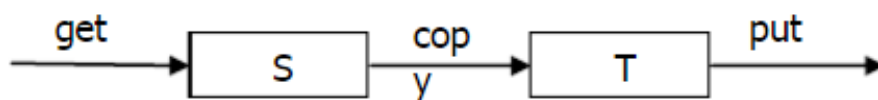
P (mutex);

登记下机信息;

V (mutex);

#### 6. (2010 年专升本考试)

如下图，有三个并发进程 get、copy、put，三个进程共用两个缓冲区 S、T（其大小每次存放一个数据），get 将数据放入 S，copy 将数据从 S 取出放入 T，put 从 T 中取出数据。在将缓冲区中的上一个数据取走之前不能放入新数据，缓冲区初始时空。试用 P、V 操作实现三个进程之间的同步，并写出程序描述。



分析：简单的生产者-消费者问题

解：S11：记录 S 缓冲区是否为空，初值为 1；

S12：记录 S 缓冲区是否为满，初值为 0；

S21：记录 T 缓冲区是否为空，初值为 1；

S22：记录 T 缓冲区是否为满，初值为 0；

Get 进程：

Repeat

生产数据；

P (S11);

往 S 中放数据；

V (S12);

Until false

Copy 进程:

Repeat

P (S12);

从 S 中取数据;

V (S11);

P (S21);

往 T 中放数据;

V (S22);

Until false

Put 进程

Repeat

P (S22);

从 T 中取数据;

V (S21);

Until false

7. 在一个只允许单向行驶的十字路口, 分别有若干由东向西, 由南向北的车辆在等待通过十字路口。为了安全, 每次只允许一辆车通过 (东向西或南向北)。当有车辆通过时其它车辆等待, 当无车辆在路口行驶时则允许一辆车进入。请用 PV 操作实现能安全行驶的自动管理系统。(西安电子科技大学 2001 研究生入学考试)

分析: 本题为简单的进程对临界资源的互斥问题



解： S：用于实现两个方向的车辆对十字路口的互斥使用，初值为 1  
东西方向车辆：

Repeat

P (S);

行驶;

V (S);

Until false

南北方向车辆：

Repeat

P (S);

行驶;

V (S);

Until false

8. 某火车订票系统，提供给多个用户同时共享一个订票数据库。规定多个用户允许同时查询该数据库，有查询者时，用户不能订票；有用户订票而需要更新数据库时，不可以有其它用户使用数据库。请用 PV 操作描述查询和订票者的同步执行程序。

分析：本题是典型的读者—写者问题。

9. 吃水果问题

爸爸为儿子和女儿切水果，切好的水果放在一个盘子里，每次只放一种。儿子爱吃苹果，若盘子里是苹果则取走；女儿爱吃橘子，只能取走盘子里的橘子。请用 PV 操作实现上述的同步过程，并说明信号量

的物理含义及初值

分析：本题为生产者-消费者问题，类似于第 3 题，请自行完成。

参考答案：

S：用于记录盘子是否为空的，初值为 1；

S1：记录盘子中是否有苹果，初值为 0；

S2：记录盘子中是否有橘子，初值为 0；

Father 进程：

Repeat

P (S)；

如果是苹果，V (S1)；否则，V (S2)；

Until false

Son 进程：

Repeat

P (S1)；

取走苹果；

V(S)；

Until false

daughter 进程：

Repeat

P (S2)；

取走橘子；

V(S)；

Until false