```
type producer_consumer = MONITOR {
  int buf[9];
  int count, getptr, putptr;
  count = 0; getptr = 0; putptr = 0;
  semaphore put, get;
  int put_count, get_count;
  InterfaceModule IM;
  DEFINE put, get;
  USE enter(), leave(), wait(), signal();
```

```
process put(int x1, int x2, int x3){
  enter(IM);
    if (count > 6)
      wait(put, put_count, IM);
    count += 3;
    buf[putptr] = x1;
    putptr = (putptr + 1) % 9;
    buf[putptr] = x2;
    putptr = (putptr + 1) % 9;
    buf[putptr] = x3;
    putptr = (putptr + 1) % 9;
    signal(get, get_count ,IM);
    signal(get, get_count ,IM);
    signal(get, get_count ,IM);
  leave(IM);
}
```

```
process get(){
  enter(IM);
    if (count == 0)
      wait(get, get_count, IM);
    y = buf[getptr];
    getptr = (getptr + 1) % 9;
    count--;
    if (count < 7){
      signal(put, put_count, IM);
    }else if (count > 0){
      signal(get, get_count, IM);
    }
  leave(IM);
}
```

```
cobegin
```

```
process producer_i(){
  while(1){
    {生产3个整数};
    producer_consumer.put(a1, a2, a3);
  }
}
```

```
process consumer_j(){
  while(1){
    y = producer_consumer.get();
    {消费整数y};
  }
}
```

```
coend
```