 moniter reader\_writer{

        int write, read;                              //管程的局部变量。

        Condition r,w;                               //进行同步机制的条件变量

        Entry start\_read() {

            if(write>0) {

                  r.wait();                              //判断是否有写者在写。

             }

            read++;

            r.signal();

        }

        Entry end\_read() {

            read--;

            if(read==0)    w.signal();             //一波读者读完才能唤醒写者.

        }

        Entry start\_write() {

            write++;                                     // 一波写者进行写操作，其它读者或写者被同步阻塞

            if(write>0||read>0)

            {

                w.wait();

            }                                                //等待写者写完或者一波读者读完。

        }

        Entry end\_write() {

                write--;

            if(write==0)

            {

                w.signal();

                r.signal()                               //此时唤醒读者写者都可以，因为一波写者已经写完了。

            }

        }

    }