

Apr 13, 09 14:39

**DiningPhilosophersMonitor.java**

Page 1/2

```

/*****
/
/      filename:  DiningPhilosophersMonitor.java
/
/      description:  This is Dining Philosophers Monitor implementation w/o
/                   starvation or deadlock.
/
/      author:    Paladino, Zac
/      login id:  cps346-n1.16
/
/      class:    CPS 346
/      instructor:  Perugini
/      assignment:  PJ #3
/
/      assigned:  March 11, 2009
/      due:      April 15, 2009
/
/*****/
import java.util.concurrent.*;
import java.util.concurrent.locks.*;

class DiningPhilosophersMonitor implements DiningServer
{
    Lock lock;
    enum State {THINKING, HUNGRY, EATING};
    State[] state;
    Condition[] self;
    int[] Eaten;
    int NumPhil;

    DiningPhilosophersMonitor(int numphil){
        self = new Condition[numphil];
        state = new State[numphil];
        Eaten = new int[numphil];
        lock = new ReentrantLock();
        for(int i=0; i<numphil; i++){
            state[i] = State.THINKING;
            self[i] = lock.newCondition();
            Eaten[i] = 0;
        }
        NumPhil = numphil;
    }

    public void pickupForks(int id){
        lock.lock();
        state[id] = State.HUNGRY;
        test(id);
        if(state[id] != State.EATING){
            try{self[id].await();}catch(InterruptedException e){}
        }
        lock.unlock();
    }

    public void releaseForks(int id){
        lock.lock();
        state[id] = State.THINKING;
        test((id+1)%NumPhil);
        test((id+(NumPhil-1))%NumPhil);
        lock.unlock();
    }

    private void test(int id){
        if((state[(id+1)%NumPhil] != State.EATING)&&
           (state[(id+(NumPhil-1))%NumPhil] != State.EATING)&&
           (state[id] == State.HUNGRY)){

            if(((state[(id+1)%NumPhil] != State.HUNGRY)&&
               (state[(id+(NumPhil-1))%NumPhil] != State.HUNGRY))
               || ((Eaten[(id+1)%NumPhil] >= Eaten[id])&&

```

Apr 13, 09 14:39

**DiningPhilosophersMonitor.java**

Page 2/2

```

           (Eaten[(id+(NumPhil-1))%NumPhil] >= Eaten[id]))){
                state[id] = State.EATING;
                Eaten[id]++;
                self[id].signal();
            }
        }
    }
}

```