

Section Solution

Solution: RSS News Feed Madness

```
static struct ta {
    bool available;           // all true to start
    Semaphore availLock;     // all set to 1 to start
    Semaphore requested;     // student-to-ta rendezvous, all set to 0 initially
    Semaphore finished;     // ta-to-student rendezvous, all set to 0 initially
    int numBugs;             // set by TA using Examine function
} tas[NUM_TAS];

static Semaphore numTAsAvailable;           // set to NUM_TAS
static Semaphore numMachinesAvailable;     // set to NUM_MACHINES
static int numStudentsLeft = NUM_STUDENTS;
static Semaphore studentsLeftLock;         // initially set to 1

static void TA(int id)
{
    while (true) {
        SemaphoreWait(tas[id].requested);
        if (numStudentsLeft == 0) return; // last student left
        tas[id].numBugs = Examine();
        SemaphoreSignal(tas[id].finished);
        ReadEmail();
    }
}

static void Student()
{
    int numBugs = 1;
    int ta;

    SemaphoreWait(numMachinesAvailable);

    while ((numBugs > 0) && (numBugs < 10)) {
        Debug();
        SemaphoreWait(numTAsAvailable);
        for (ta = 0; ta < NUM_TAS; ta++) {
            SemaphoreWait(tas[ta].availLock);
            if (tas[ta].available) break;
            SemaphoreSignal(tas[ta].availLock);
        }
        tas[ta].available = false;
        SemaphoreSignal(tas[ta].availLock);
        SemaphoreSignal(tas[ta].requested);
        SemaphoreWait(tas[ta].finished);
        numBugs = tas[ta].numBugs;
        tas[ta].available = true;
        SemaphoreSignal(numTAsAvailable);
    }
}
```

```
if (numBugs == 0) Rejoice();

SemaphoreWait(studentsLeftLock);
numStudentsLeft--;
bool everyoneDone = (numStudentsLeft == 0);
SemaphoreSignal(studentsLeftLock);
// thought question: why can't the two lines above be switched?

if (everyoneDone) {
    for (ta = 0; ta < NUM_TAS; ta++) {
        SemaphoreSignal(tas[ta].requested);
    }
}

SemaphoreSignal(numMachinesAvailable);
}
```