Readers/Writers Problem, Working in Teams

February 10, 2014
Anthony D. Joseph
http://inst.eecs.berkeley.edu/~cs162
Examples of Read-Modify-Write Processor Instructions

• test&set (&address) { /* most architectures */
   result = M[address];
   M[address] = 1;
   return result;
}

• swap (&address, register) { /* x86 */
   temp = M[address];
   M[address] = register;
   register = temp;
}

• compare&swap (&address, reg1, reg2) { /* 68000 */
   if (reg1 == M[address]) {
      M[address] = reg2;
      return success;
   } else {
      return failure;
   }
}
Synchronized (Infinite) Queue Example

AddToQueue(item) {
    lock.Acquire();
    queue.enqueue(item);
    dataready.signal();
    lock.Release();
}

RemoveFromQueue() {
    lock.Acquire();
    if (queue.isEmpty()) {
        dataready.wait(&lock);
    }
    item = queue.dequeue();
    lock.Release();
    return(item);
}
Basic Readers/Writers Solution

• Correctness Constraints:
  – Readers can access database when no writers
  – Writers can access database when no readers or writers
  – Only one thread manipulates state variables at a time
• Basic structure of a solution:
  – Reader()
    Wait until no writers
    Access database
    Check out – wake up a waiting writer
  – Writer()
    Wait until no active readers or writers
    Access database
    Check out – wake up waiting readers or writer
  – State variables (Protected by a lock called “lock”):
    » int AR: Number of active readers; initially = 0
    » int WR: Number of waiting readers; initially = 0
    » int AW: Number of active writers; initially = 0
    » int WW: Number of waiting writers; initially = 0
    » Condition okToRead = NIL
    » Condition okToWrite = NIL
Read/Writer Questions

```c
Reader() {
    // check into system
    lock.Acquire();
    while ((AW + WW) > 0) {
        WR++;
        okToRead.wait(&lock);
        WR--;
    }
    AR++;
    lock.release();

    // read-only access
    AccessDbase(ReadOnly);

    // check out of system
    lock.Acquire();
    AR--;
    if (AR == 0 && WW > 0)
        okToWrite.signal();
    lock.release();
}

Writer() {
    // check into system
    lock.Acquire();
    while ((AW + AR) > 0) {
        WW++;
        okToWrite.wait(&lock);
        WW--;
    }
    AW++;
    lock.release();

    // read/write access
    AccessDbase(ReadWrite);

    // check out of system
    lock.Acquire();
    AW--;
    if (WW > 0)
        okToWrite.signal();
    else if (WR > 0)
        okToRead.broadcast();
    lock.release();
}
```
5min Break
Tips for Programming in a Project Team

- Big projects require more than one person (or long, long, long time)
  - Big OS: thousands of person-years!
- It’s very hard to make software project teams work correctly
  - Doesn’t seem to be as true of big construction projects
    » Empire state building finished in one year: staging iron production thousands of miles away
    » Or the Hoover dam: built towns to hold workers

“You just have to get your synchronization right!”