Linda (or, ”Spaces”)

K. V. S. Prasad
Dept of Computer Science
Chalmers University
19 Sep 2016

Questions?

- Anything you want to say
  - Comments, questions, stray thoughts, etc.
  - Are we too fast/slow?
- Reminder – talk to your rep!
  - Feedback meeting after class
- Practical problems?
  - Don’t miss deadlines! (you’re not Douglas Adams)
  - Registration and other formalities
  - Find a lab partner!

Plan for today, and where we are

- Chap 9: Linda
  - Chap 3, 4, 5 (skipped for now)
  - Chap 6, 7, 8 – need more detail
- BUT!
  - You now know enough to try the exercises
  - in Chaps. 1, 2, 3, 6, 7, 8

Comments on message passing

- Inter-process
  - Communication
  - Coordination
  - Cooperation
  - Contention
  - Concurrency
  - Synchronisation
- We mentioned simulation and examples such as pilots, athletes, dancers, musicians, …

Examples from the book

- Critical Section
- Producer-consumer
  - Doesn't matter whether synch/asynch
- Dining philosophers
  - With synchronous channels only.
  - Each fork behaves like a semaphore
  - Both deadlock and starvation seem possible!

Rendezvous

- Like synchronous channel, except
  - Addressing asymmetric
    - Sender knows receiver's address (entry), not v-v.
  - The communication may involve computation and return of value by the receiver
  - So made for client-server

Ada

- Uses protected objects
  - Since the 1980's
    - though the concept was around earlier
  - Thus has the cleanest shared memory model
- Also has a very good communication model
  - Rendezvous
- Ada was decided carefully through the 1970s
  - Open debates and process of definition
- Has fallen away because of popularity of C, etc.
  - Use now seen as a proprietary secret!

Loosely coupled systems

- Tightly coupled systems
  - Shared memory
  - Synchronous communication
    - Whether one-to-one or broadcast
- Loosely coupled
  - Asynchronous communication
  - Persistent messages
- Linda is such a system
  - So are filing systems and databases?

Tuple space

- Large shared notice board
- Posted notes are in the form of tuples
- Can read notes matching any pattern
  - E.g., you look for a pair
    - Only singletons and triples posted
      - Block until someone posts a pair
- This blocking gives us synchronisation

Linda primitives

- Post(v1, v2, …, vn)
  - Put tuple of values out
  - Release an arbitrary proc waiting on this pattern
- Remove(X1, x2, …, Xn)
  - X's are variables and x's are constants
  - Remove an arbitrary matching note
  - Block if none available
- Read(X1, x2, …, Xn)
  - Like remove, but leave note on board

Generalisation of read and remove

- Allow patterns such as (X, 4, Y)
  - Matches only triples with middle element 4
- Allow patterns such as (X, c=, Y)
  - Where c is a variable
  - Matches only triples with middle element = c

Linda examples

- From the book
  - Slides 8.4 thru 8.7
    - Matrix multiplication using channels
  - Slides 9.1 thru 9.8
    - CS, client-server, buffer, matrix in Linda
- Given a monotonically increasing function f
    - with $f(0) < 0$ and $f(1) > 0$
    - find x where $0<x<1$ such that $f(x)=0$.
  - Can be done by binary search
  - How to use more than one process
    - Can use ability to interrupt

The matrix example

$$\begin{vmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{vmatrix} \times \begin{vmatrix} 1 & 0 & 2 \\ 0 & 1 & 2 \\ 1 & 0 & 0 \end{vmatrix} = \begin{vmatrix} 4 & 2 & 6 \\ 10 & 5 & 18 \\ 16 & 8 & 30 \end{vmatrix}$$

So element (3,3) of the result is

$$[7 \;\; 8 \;\; 9] \times \begin{vmatrix} 2 \\ 2 \\ 0 \end{vmatrix} = 7*2 + 8*2 + 9*0 = 30$$