**cs4414 Handout – 8 April 2014**

**Dijkstra’s Mutual Exclusion Problem:**

1. Only one thread may be in the critical section at any time.
2. Each must eventually be able to enter its critical section.
3. Must be symmetrical (all run same program).
4. Cannot make any assumptions about speed of threads.

*“We beg the challenged reader to stop here for a while, and have a try, for this seems the only way to get a feeling for the tricky consequences of the fact that each computer can only request one one-way message at a time. And only this will make the reader realize to what extent this problem is far from trivial.”*

**Initialization**

b[1:N] = [true, true, …]

c[1:N] = [true, true, …]

k = ?

**Program** for Processor i

**loop {**

b[i] := false

L1: **if** k != i

c[i] := true

if b[k]

k := i

goto L1

**else:**

c[i] := false

**for** j in [1, …, N]:

**if** j != i and not c[j]:

**goto L1**

**critical section;**

c[i] := true

b[i] := true

**}**

Lamport’s Solution

