ECE 428/CS 425
Spring 2016
Distributed Systems
Homework 6

Due: March 15, 2016 by 5:00 p.m.

Total points: 30

Each question is worth 10 points.
Consider the Bakery algorithm modified to delete 3 lines, as shown on the next slide. In the modified algorithm, variable Choosing is not used at all.

Present a scenario for $n = 2$ or $3$ in which the algorithm may violate the mutual exclusion property.
Bakery Algorithm

Code for entry section:

Choosing[i] := true
Number[i] := max{Number[0], ..., Number[n-1]} + 1
Choosing[i] := false
for j := 0 to n-1 (except i) do
    wait until Choosing[j] = false
    wait until Number[j] = 0 or (Number[j],j) > (Number[i],i)
endfor

Code for exit section:

Number[i] := 0
Question 2

Consider the 2-processor algorithm modified to delete 1 line in the code for processor \( p_1 \), as shown on the next slides.

Present a scenario in which the algorithm either violates the mutual exclusion property, or results in a deadlock.
2-Processor Algorithm for $p_0$

Code for $p_0$'s entry section:

```
1 .
2 .
3 W[0] := 1
4 .
5 .
6 wait until W[1] = 0
```

Code for $p_0$'s exit section:

```
7 .
8 W[0] := 0
```
2-Processor Algorithm for $p_1$

Code for $p_1$'s entry section:

1. $W[1] := 0$
2. wait until $W[0] = 0$
3. $W[1] := 1$
4. ...
5. if ($W[0] = 1$) then goto Line 1
6. ...

Code for $p_1$'s exit section:

7. ...
8. $W[1] := 0$
Suppose that process P1 performs the following write operations in the specified order:

\[
\begin{align*}
  x & := 1 \\
  y & := 2 \\
  z & := 3 \\
\end{align*}
\]

Assume that, initially, all variables have value 0.

Suppose that the above operations have been completed (i.e., P1 has received acknowledgements that these operations are completed).

At a later time, process P1 performs the following operation:

\[
a := x + y + z
\]

Assume that no other operations performed. **Under each of the consistency models below**, what are the different values possible for variable \(a\), after the above update of variable \(a\) has been completed?

(a) Eventual consistency (i.e., without read-my-write guarantee) (b) Read-my-write
4. Failure Detector

Consider an asynchronous system. Suppose that process p sends a message to process q every 100 ms, and expects a reply from process q at least once every second. When no reply is received within 1 second interval, p assumes that q has crashed.

(a) Is it possible that process q crashes, but process p never concludes that q has crashed?

(b) Is it possible that process q is operating correctly, but process p concludes that q has crashed?