Problem Set 5, Part a

Due: Thursday, November 19, 2015

Readings:

Chapter 10, but just skim Section 10.9.
Chapter 11 (skim).

Next week: Chapter 12; Sections 13.1-13.3.

Problems:

1. (Based on Exercises 10.2 and 10.3.)
   Consider Dijkstra's mutual exclusion algorithm.
   (a) Show that it is not lockout-free; that is, describe a fair execution of the algorithm in which a particular process is locked out.
   (b) Show that, if we remove the second phase of the algorithm (where the $flag$ is raised to 2 and the other processes' $flag$s are tested), then the resulting algorithm does not satisfy mutual exclusion.
2. (Based on Exercise 10.13.)
   Modify the PetersonNP algorithm to yield a solution to the 2-exclusion problem. This problem allows 2 processes to coexist in the critical region.
   Formally, the mutual exclusion condition is modified to forbid more than two processes to be in the critical region at once. The progress condition for the trying region is also modified, to say that, in a fair execution, if the critical region is occupied by at most one process, and another process is trying, then some trying process must eventually enter the critical region. (Note that we are not assuming that processes eventually return the resource.)
   In writing your algorithm, you may use either sequential-style pseudocode like that on p. 284, or I/O-automata-style pseudocode like that on p. 285-286.
3. (Based on Exercises 10.23 and 10.24)
   Consider a weaker read/write shared-memory model, in which read and write operations on shared variables are no longer instantaneous, but have duration. Specifically, suppose that the shared registers are guaranteed only to be *safe*, that is, a read operation is guaranteed to yield the correct value in the absence of concurrent write operations, whereas a read operation that overlaps a write may yield any value of the variable's type.
   (a) Does the *Bakery* algorithm work correctly in this weaker model? Prove that it does or give a counterexample.
   (b) Does the *Burns* algorithm work correctly in this weaker model? Prove that it does or give a counterexample.
4. Consider the 2-exclusion problem, as defined in Problem 2 above.
   (a) Prove that three processes cannot solve the 2-exclusion problem using just one read/write shared variable.
   (b) Strengthen your solution to part (a) to prove that three processes cannot solve the 2-exclusion problem using two read/write shared variables.