Concurrent Systems

An $n$-process mutual exclusion algorithm has entry and exit protocols given below. In order to express the algorithm concisely we use a lexicographic “less than” relation on ordered pairs of integers, so that:

$$(a, b) < (c, d)$$

if $a < c$ or if $a = c$ and $b < d$.

Entry protocol for the critical region for process $i$:

- $taking[i] := true$;
- $ticket[i] := \max(ticket[0], ticket[1], \ldots, ticket[n-1]) + 1$;
- $taking[i] := false$;
- for $j := 0$ to $n-1$ do
  - begin
    - while $taking[j]$ do no-op;
    - while $ticket[j] \neq 0$ and $(ticket[j], j) < (ticket[i], i)$ do no-op;
  - end

Exit protocol for the critical region for process $i$:

- $ticket[i] := 0$;

(a) Illustrate fully the operation of the algorithm by showing, for a small value of $n$, successive values of the arrays $taking$ and $ticket$ under a variety of concurrent executions. Explain by means of short comments on the values. [14 marks]

(b) Is it possible or likely that a value in the array $ticket$ might overflow? Why? [2 marks]

(c) A RISC processor has an atomic read-and-clear-memory instruction. Give pseudo-code for the entry and exit protocols using the instruction. [4 marks]