T-106.420 Concurrent programming
Homework assignment 2 (Oct. 13, 2003)

The deadline for this exercise is Sunday, Oct. 26, 2003

1. Express each of the following as a formula of first order predicate logic, using suitable quantifications where necessary:

   (a) The average of the array $x[0...N-1]$, $N \geq 1$, is m.

   (b) The segment of the array x that is between indices p and $q-1$ inclusive contains only zeroes.

   (c) The index of the minimum value of the array $x[0...N]$, $N \geq 0$, is i.

   (d) The value of each element in array $x[0...N]$, $N \geq 1$, is unique (within the array).

2. Suppose your machine has the following atomic instruction:

$$\begin{array}{l}\mathsf{flip(lock)} : \langle \mathsf{lock} = (\mathsf{lock} + 1) \ \% \ 2; \quad \# \ \mathsf{flip\ the\ lock} \\ \qquad\quad \mathsf{return(lock);} \rangle\end{array}$$

   Someone suggests the following solution to the critical section problem for *two* processes:

```
int lock = 0;

process CS[i = 1 to 2] {
  while (true) {
    while (flip(lock) != 1)
      while (lock != 0) skip;
    critSection();
    lock = 0;
    nonCritSection();
  }
}
```

   (a) Does this solution work? If, not, give an execution order that results in both processes being in their critical sections at the same time.

   (b) How does the situation change if the number of processes remains the same, but the flip instruction has the form:

$$\begin{array}{l}\mathsf{flip(lock)} : \langle \mathsf{lock} = (\mathsf{lock} + 1) \ \% \ 3; \quad \# \ \mathsf{flip\ the\ lock} \\ \qquad\quad \mathsf{return(lock);} \rangle\end{array}$$