Solution of a Problem in Concurrent Programming Control

E. W. Dijkstra
Technological University, Eindhoven, The Netherlands

A number of mainly independent sequential-cyclic processes with restricted means of communication with each other can be made in such a way that at any moment one and only one of them is engaged in the "critical section" of its cycle.

Introduction

Given in this paper is a solution to a problem for which, to the knowledge of the author, has been an open question since at least 1962, irrespective of the solvability. The paper consists of three parts: the problem, the solution, and the proof. Although the setting of the problem might seem somewhat academic at first, the author trusts that 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.

The Solution

The common store consists of:

"Boolean array b, c[1:N]; integer k"

The integer k will satisfy \(1 \leq k \leq N\), \(b[i]\) and \(c[i]\) will only be set by the ith computer; they will be inspected by the others. It is assumed that all computers are started well outside their critical sections with all Boolean arrays mentioned set to true; the starting value of k is immaterial.

The program for the ith computer (\(1 \leq i \leq N\)) is:

\[
\begin{align*}
L0: & \quad b[i] := false; \\
L1: & \quad \text{if } k \neq i \text{ then } c[i] := true; \\
L2: & \quad \text{begin } c[i] := true; \text{go to } L1 \\
L3: & \quad \text{if } b[k] \text{ then } k := i; \text{go to } L1
\end{align*}
\]

and the proof. Although the setting of the problem might seem somewhat academic at first, the author trusts that