module Table
  op getforks(int), relforks(int);

body
process Waiter {
  bool eating[5] = ([5] false);
  while (true)
    in getforks(i) and not (eating[left(i)] and
      not eating[right(i)]) -> eating[i] = true;
    [] relforks(i) ->
      eating[i] = false;
  ni
}
end Table

process Philosopher[i = 0 to 4] {
  while (true) {
    call getforks(i);
    eat;
    call relforks(i);
    think;
  }
}

Figure 8.6 Centralized dining philosophers using rendezvous.

Copyright © 2000 by Addison Wesley Longman, Inc.