

491✓ (dining philosophers) Five philosophers are sitting around a round table. At the center of the table is an infinite bowl of noodles. Between each pair of neighboring philosophers is a chopstick. Whenever a philosopher gets hungry, the hungry philosopher reaches for the two chopsticks on the left and right, because it takes two chopsticks to eat. If either chopstick is unavailable because the neighboring philosopher is using it, then this hungry philosopher will have to wait until it is available again. When both chopsticks are available, the philosopher eats for a while, then puts down the chopsticks, and goes back to thinking, until the philosopher gets hungry again. The problem is to write a program whose execution simulates the life of these philosophers with the maximum concurrency that does not lead to deadlock.

§ see book Subsection 8.1.2