This is the dining philosophers problem.

The main Method of ConcurrentPhilosopher.java now runs in parallel.

My solution to the problem was to alternate the order in which the philosophers pick up the forks. If the philosopher's number is even then he picks up his left fork first, odd philosophers pick up their right fork first. This solves the deadlock problem.

We also could have made the action of picking up two forks at once atomic. We also could have used a waiter that only allows n-1 philosophers to eat at once. There are many solutions to this problem.