Massachusetts Institute of Technology
6.042J/18.062J: Mathematics for Computer Science
Professor David Karger

Tutorial 5
March 10, 1997

Solutions for Tutorial 5

Problem 1. *Let R be a relation on a possibly infinite set A. Recall that the powers of a relation are recursively defined by*

$$R^1 := R$$

$$R^{n+1} := R^n \circ R.$$

We define the Transitive Closure *of R to be the relation given by $\bar{R} = \bigcup_{i=1}^{\infty} R^n$*

Prove that $\bar{R}$ is transitive.

Show that for any $(a,b) \in \bar{R} \wedge (b,c) \in \bar{R}$, $(a,c) \in \bar{R}$.

If $(a,b) \in \bar{R}$, then $(a,b) \in R^i$ for some $1 \leq i \leq \infty$, by the definition of $\bar{R}$ (union of all subsets of R^n). Similarly, if $(b,c) \in \bar{R}$, then $(b,c) \in R^j$ for some $1 \leq j \leq \infty$. (a,c) is formed by $R^j \circ R^i = R^{i+j}$ (which can be shown by applying the $R^{n+1} := R^n \circ R$ rule i times on $R^j \circ R^i$). So $R^j \circ R^i \subset \bar{R}$ and thus $(a,c) \in \bar{R}$, so $\bar{R}$ is transitive.

Problem 2. The Dining Philosophers

There are 4 philosophers who sit down to eat dinner at a round table. In front of each of them is a plate of spaghetti and to the right of each of them is a fork. In order to eat, each philosopher must do the following:

1. *pick up the fork directly to his left*
2. *pick up the fork directly to his right*
3. *eat*
4. *put down the fork directly to his left*
5. *pick down the fork directly to his right*

(a) *Show the state machine for this scenario. Show that it is possible for a series of state transitions to occur such that no philosopher can eat (this is called deadlock).*

PickUpLeftFork(p)
Precondition

p has no forks
there is a fork to p's left
Effect
p has a left fork
no longer a fork to p's left

PickUpRightFork(p)
Precondition
p has a left fork
there is a fork to p's right
Effect
p has a right fork
no longer a fork to p's right

Eat(p)
Precondition
has two forks
hasn't eaten
Effect
Has eaten

PutDownLeftFork(p)
Precondition
has a left fork
has eaten
Effect
no longer has a left fork
fork is now on his left

PutDownRightFork(p)
Precondition
has a right fork, but no left fork
has eaten
Effect
no longer has a right fork
fork is now on his right

Deadlock occurs when all philosophers execute PickUpLeftFork but not PickUpRightFork at which point each philosopher has one fork, but there are no more forks on the table.

(b) *Detecting a problem in their eating strategy, the philosophers start over. Each philosopher is assigned a unique number 1 through 4. A fork must be used by*

philosophers in numerical order. For example, if a fork is sitting between philosophers 2 and 4, 4 must wait until 2 has picked up the fork, eaten, and put it down before 4 can use the fork. The philosophers may be seated in any order.

Show the state machine for this scenario. Prove that with any seating arrangement and any order of philosophers, all philosophers will get to finish their dinner.

Modify PickUpLeftFork and PickUpRightFork as follows:

PickUpLeftFork(p)
Precondition
p has no forks
there is a fork to p's left
the philoshopher to p's left has a higher number than p *Effect*
p has a left fork
no longer a fork to p's left

PickUpRightFork(p)
Precondition
p has a left fork
there is a fork to p's right
the philoshopher to p's right has a higher number than p *Effect*
p has a right fork
no longer a fork to p's right

Because the philosophers each have a unique value, let there be some set, S of philosophers who haven't eaten yet. By well-ordering, there must be some philosopher who's number is the lowest, and who can therefore start eating because the people to his left and right will wait for him to finish eating before picking up the fork, or else they're already done eating.