Principles of Distributed Computing
Exercise 9

1 Family Dinner

Several families go out to dinner together. To increase their social interaction, they would like to sit at tables so that no two members of the same family are at the same table. Show how to formulate finding a seating arrangement meeting this objective as a maximum flow problem. Assume that there are p families, the ith family has a_i members, and there are q tables, where the jth table has a seating capacity of b_j.

2 Emergency Route Planning

Due to a large-scale flooding, there are n injured people distributed across a region who need to be rushed to hospitals. There are k hospitals in the region, and each of the n people needs to be brought to a hospital that is within a half hour's driving time of their current location. Thus, different people have different options for hospitals, depending on their location. At the same time, the paramedics wish to balance the load on the hospitals. The paramedics all have cell phones, and so they can collectively coordinate who goes to which hospital.

Give a polynomial-time algorithm to decide if, given the locations of all patients and hospitals, the paramedics can choose hospitals so that each patient reaches a hospital within half an hour and no hospital receives more than $\lceil n/k \rceil$ patients.

(You can assume that the maxflow problem in a network can be solved in polynomial time!)