15295 Spring 2017 #5 -- Problem Discussion
February 15, 2017

This is where we collectively describe algorithms for these problems. This week the focus was on shortest path algorithms.

To see the problem statements follow [this link](). To see the scoreboard, go to [this page]() and select this contest.

A. Two Buttons

B. Dorm Water Supply

C. The Labyrinth

D. Volleyball1

E. The Two Routes

F. Complete The Graph

Raymond, Tom, and I all had different solutions to this problem. I'll just describe mine.

Let A be an assignment of weights to the unknown edges. Notice that if A is modified by increasing the length of one of the edges by 1 the distance from s to t may stay the same or it may increase by 1. So my idea was this. Given a total T of the weight of all of the unknown edges, we write a function that takes T and outputs the weights of all the unknown edges such that their weights are all at least 1, and an increase of T by 1 just increases the weight of one edge by 1. The specific function I used is illustrated below on three edges.

T=3	1 1 1
T=4	1 1 2
T=5	1 2 2
T=6	2 2 2
T=7	2 2 3

The highest value of T will be k (the number of unknown edges) times L. The lowest is k.

Define SP(T) to be the shortest path from s to t with the above assigned weights as a function of T. SP(T) can be computed by assigning weights as described above, then running Dijkstra's algorithm on the resulting graph. If $SP(k) > L$ or $SP(k*L) < L$ then there is no solution. If $SP(k)=L$ then that's the solution. If $SP(k*L) = L$ then that is the solution. Otherwise we know there a solution with T such that $k < T < k*L$. We find it by doing binary search on T. --- Danny

G. Generating Sets