Qual, Spring 2011
Algorithms

February 17, 2011

Duration: 3 hours
Number of problems: 5

1 Weighted edge cover

Let $G = (V, E)$ be a weighted undirected graph with n vertices. A *minimum weight edge cover* is a subset of the edges of G, of minimum weight, such that every vertex is covered by some edge in the set.

(A) Let P be a set of n points in the plane, and consider the complete graph on P, where the weight of an edge is the Euclidean distance between its endpoints. Prove that the minimum weight edge cover of this graph form a planar graph.

(B) Let R and B be sets of real numbers on the real line, and consider the complete bipartite graph defined on these two sets, where the weight of the edge rb, for $r \in R$ and $b \in B$, is $|r - b|$. Provide a dynamic programming algorithm that computes the minimum weight edge cover of this graph in polynomial time. What is the running time of your algorithm as a function of $n = |R| + |B|$?

2 Rooted tree

Suppose your are given a rooted tree T, where every edge e has a non-negative *length* $\ell(e)$. Your goal is to assign a stretched length $s\ell(e) \geq \ell(e)$ to every edge e in T, so that every root-to-leaf path in T has the same total stretched length, and the stretched lengths are in some sense optimal.

(A) Describe and analyze an algorithm to minimize the total stretched length $\sum_e s\ell(e)$.

(B) Now suppose every edge also has a *value* $\$(e)$, which may be positive, negative, or zero; the value of an edge is equal to the cost of increasing its length by 1. Describe and analyze an algorithm to minimize the total stretch *cost* $\sum_e \$(e) \cdot s\ell(e)$.

Faster algorithms are worth more points.

3 No such thing as a free lunch

Suppose that a group of N people (where $N > 50$) are waiting in line to eat at a restaurant. Suppose that the first person gets to eat for free. The second person gets to eat for free if they have the same birthday as the first person. Then everyone else must pay to eat. The third person gets to eat for free if the second person did not get to eat for free AND they have the same birthday as either the first or the second person. Then everyone else must pay to eat. The "eat free" rule proceeds in the obvious manner. Determine the probability that the nth person gets to eat for free. Determine the value for n such that their probability of eating for free is larger than anyone elses such probability

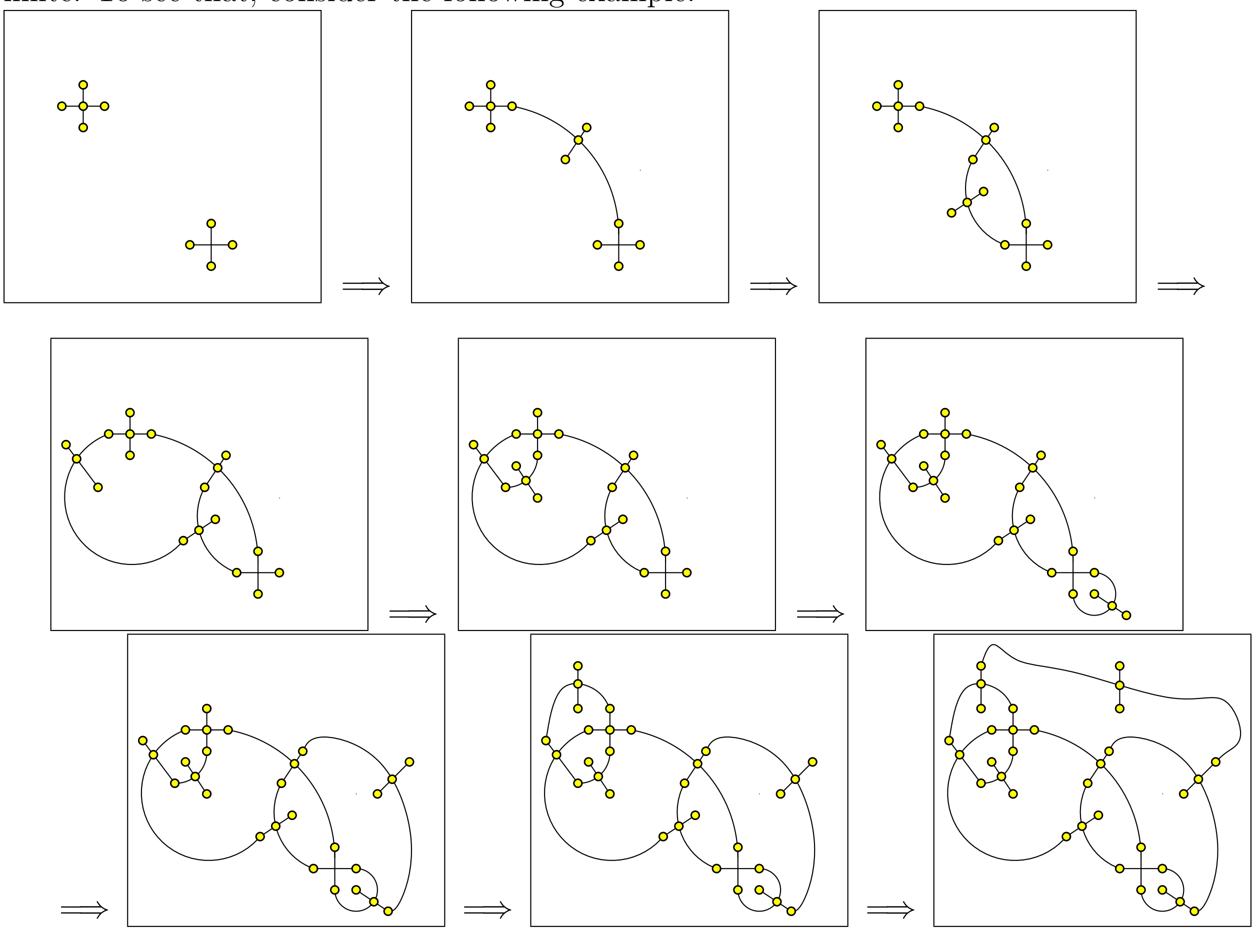
4 Matchings from a black-box

Let $G = (V, E)$ be an undirected graph and let $w : E \to \mathbb{R}_+$ be a non-negative weight function on the edges. Assume you have a black-box algorithm to compute the minimum-weight perfect matching in a given graph (the algorithm works even when weights are negative). Show how to use that algorithm to find a maximum weight matching of a given size k if one exists (your algorithm should report that there is no matching of size k if that is the case).

5 Attack of the Brussels Sprouts

The game *Brussels Sprouts* (a variant of *Sprouts*), starts with n crosses drawn in the plane. Each cross is a spot with four free ends. Each move involves joining two distinct free ends with a curve (not crossing any existing curve) and then putting a short stroke across the line to create two new free ends. The game ends when no two free ends can be joined without crossing an existing curve.[1]

So each move removes two free ends and introduces two more. Despite this, the game is finite. To see that, consider the following example:



Provide a precise bound on the number of steps in the game, as a function of n (the number of crosses). Prove the bound you provide.

[1] In the original game, two players alternate in who is drawing the curves, and the last one to draw the curve is the winner.