Problem Set 4 – due October 1st

Please see the course website for homework policy.

Problem 1 Let D be a directed graph. Let W be a walk (not an Eulerian walk, just any walk) in the graph. Let v_0 be the last vertex of W. Let T be subgraph of D where $u \to v$ is in T if $u \to v$ is the last edge used to depart from u.

(a) Show that T has at most one cycle and, if it has a cycle, that v_0 is on this cycle.

(b) Suppose that G contains an edge $v_0 \to v_1$. Let W' be the walk obtained by concatenating $v_0 \to v_1$ to W and let T' be the graph for W'. Describe T' in terms of W and $v_0 \to v_1$.

(c) Let $\tau(D, v)$ be the number of spanning trees of D rooted at v. Let $\sigma(D, v)$ be the number of connected spanning subgraphs of D with one directed cycle γ, with γ passing through v. Give a simple relation between $\tau(D, v)$ and $\sigma(D, v)$.

Problem 2 Consider the following problem: Given a graph G, with n vertices, determine whether or not it is bipartite. Describe an algorithm to do this and work through a basic estimate of how many steps this will take. The exact answer will depend on your algorithm and on precisely how you model computation. But if your answer is worse than polynomial in n, you are definitely doing something wrong.

Problem 3 Let G be a graph where every vertex has even degree. An *Eulerian orientation* of G is a way to direct the edges of G so that every vertex has in-degree equal to out-degree.

(a) Show that G has an Eulerian orientation.

Let H be a bipartite graph with equally many black and white vertices. A ***prefect matching*** of H is a collection of edges which covers every vertex of H exactly once.

(b) Let G have vertices of degrees $2d_1$, $2d_2$, $\ldots$, $2d_m$. Describe a (polynomial time) algorithm which constructs a bipartite graph H with $2\sum d_i$ vertices so that

$$\#\,(\text{perfect matchings of } H) = \#\,(\text{Eulerian orientations of } G) \cdot \prod_i (d_i)!$$