Homework 2 (7 problems, due: 9 October 2006)

Graph reconstruction Read and understand the solution to the first problem, and submit solutions for the remaining.

C1.6 Graph embeddings: Our graphs are undirected and simple with vertex set $[n]$. Let G_1 and G_2 be graphs with m edges. For graphs H and G, we say that $f : [n] \to [n]$ is an embedding of H in G if (a) f is one-to-one and onto, and (b) for all $\{i, j\} \in E(H)$, we have $\{f(i), f(j)\} \in E(G)$. Suppose for each graph H with $m - 1$ edges, the number of subgraphs of G_1 that are isomorphic to H is equal to the number of subgraphs of G_2 that are isomorphic to H. Then, show that for every graph H with *at most* $m - 1$ edges, the number of embeddings of H in G_1 is equal to the number of embeddings of H in G_2.

Solution: Order the edges of G_1 as $e_1, e_2, \ldots, e_m$ and G_2 as $f_1, f_2, \ldots, f_m$ so that the graph $G_1 - e_i$ is isomorphic to $G_2 - f_i$; fix an embedding σ_i of $G_1 - e_i$ in $G_2 - f_i$ for each $i \in [m]$. Let H be some graph on $[n]$ with $k \leq m - 1$ edges. We want to show that the number of embeddings of H in G_1 is equal to the number of embeddings of H in G_2. Let

$$\begin{aligned} S_1 &\triangleq \{(f, i) : f \text{ is an embedding of } H \text{ in } G_1 - e_i\}; \\ \text{and } S_2 &\triangleq \{(f, i) : f \text{ is an embedding of } H \text{ in } G_2 - f_i\}. \end{aligned}$$

Note that $(f, i) \in S_1$ if and only if $(\sigma_i \cdot f, i) \in S_2$; so, $|S_1| = |S_2|$. Now, if f is an embedding of H in G, there are exactly $m - k$ indices i such that f is an embedding of H in $G - e_i$. It follows that

$$\text{the number of embeddings of } H \text{ in } G_1 = \frac{1}{m-k}|S_1|.$$

Similarly,

$$\text{the number of embeddings of } H \text{ in } G_2 = \frac{1}{m-k}|S_2|.$$

But, we just argued that $|S_1| = |S_2|$. So, the number of embeddings of H in G_1 is equal to the number of embeddings of H in G_2.

C2.1 Let N be finite set, and let $\mathcal{P}(N)$ be the power set of N. Let $f : \mathcal{P}(N) \to \mathbb{R}$. Define $e : \mathcal{P}(N) \to \mathbb{R}$ by

$$e(T) \triangleq \sum_{S : S \supseteq T} f(S). \tag{1}$$

Suppose for some subset T of size m we have $e(T) \neq 0$, but $e(T') = 0$ for all proper subsets T' of T. Show that there are at least 2^m sets $S \subseteq N$ such that $f(S) \neq 0$.

C2.2 Let $N = \binom{[n]}{2}$. Let G be a graph on $[n]$ and m edges, that is, $G \in \binom{N}{m}$. Let the function $f : \mathcal{P}(N) \to \mathbb{R}$ be defined as follows: if G' has exactly m edges, then $f(G')$ is the number of embeddings of G in G'; if G' des not have exactly m edges, then $f(G) \triangleq 0$. Using this f, define $e : \mathcal{P}(N) \to \mathbb{R}$ as in (1). Show that $e(H)$ is exactly the number of embeddings of H in G.

C2.3 Observe that the number of G' for which $f(G') \neq 0$ is at most $n!$ (Why?). Use C1.6 and C2.2 to conclude that if two graphs G_1 and G_2 with m edges have the same list (or deck as in Lovász's paper) of subgraphs with $m-1$ edges, then the resulting e's obtained from them (as in C2.2) take the same value for all graphs H with at most $m-1$ edges. Conclude that if G_1 and G_2 are not isomorphic, then $2 \cdot n! \geq 2^m$. That is, if $m > 1 + \log_2(n!)$, then the graph can be reconstructed from its deck. Note that Lovász's proof, presented in class, showed that we can reconstruct the graph provided it has more than $\frac{1}{2}\binom{n}{2}$ edges.

Permutations

C2.4 A permutation in $\mathcal{S}_n$ is a *transposition* if it has one cycle with two elements and $n-2$ cycles with one element each. Show how you will write the cycle $(a_1, , a_2 \ldots, a_m)$ as a product of $m-1$ transpositions. If $\sigma \in \mathcal{S}_n$ is a permutation with k cycles, show that σ can be written as a product of $n-k$ transpositions. Can it be written as a product of fewer than $n-k$ transpositions?

C2.5 Suppose you are given an array $(A[i] : i = 1, 2, \ldots, n)$, which contains the numbers $1, 2, \ldots, n$ stored in some order. To move the elements of the array, the only operation we are allowed is swap(i, j), where i and j are distinct indices in the range $1, 2 \ldots, n$. This operation exchanges the values in $A[i]$ and $A[j]$. Give a linear-time algorithm that uses the swap operation repeatedly so that in the end the element i is in the location $A[i]$. Your program can use an auxiliary bit-array $(B[i] : i = 1, 2, \ldots, n)$, and a constant number of other variables, each holding an integer in the range $0, 1, \ldots, n+1$. How many swaps will your algorithm need if the initial content of the array corresponds to a permutation with k cycles (that is, if we define the permuation $\sigma : [n] \rightarrow [n]$ by $\sigma[i] \triangleq A[i]$, then σ has k cycles)? Can an algorithm (not necessarily linear-time) use even fewer swaps?

C2.6 Let $\sigma \in \mathcal{S}_n$ be a permutation with k cycles. Consider the transposition $(1, 2)$. How many cycles can $(1, 2) \cdot \sigma$ have?

C2.7 Consider a permutation $\rho \in \mathcal{S}_n$ with exactly one non-trivial cycle $(a_1,\ a_2, \ldots, a_m)$. Suppose $\sigma \in \mathcal{S}_n$. Describe the cycles of the permutation $\sigma \cdot \rho \cdot \sigma^{-1}$.

Please send me email (jaikumar@tifr.res.in) when you spot errors. – Jaikumar