CS 573: Algorithms, Fall 2014

Homework 2, due Monday, October 13, 23:59:59, 2014

Version 1.02

Name:	
Net ID:	Alias:

Neatly print your name(s), NetID(s). Staple this sheet to the top of your homework. If you are on campus, submit the homework by submitting it in the homework boxes in the basement of SC. Also, please submit the homework electronically on moodle.

> To acknowledge the corn – This purely American expression means to admit the losing of an argument, especially in regard to a detail; to retract; to admit defeat. It is over a hundred years old. Andrew Stewart, a member of Congress, is said to have mentioned it in a speech in 1828. He said that haystacks and cornfields were sent by Indiana, Ohio and Kentucky to Philadelphia and New York. Charles A. Wickliffe, a member from Kentucky questioned the statement by commenting that haystacks and cornfields could not walk. Stewart then pointed out that he did not mean literal haystacks and cornfields, but the horses, mules, and hogs for which the hay and corn were raised. Wickliffe then rose to his feet, and said, "Mr. Speaker, I acknowledge the corn".
> – Funk, Earle. "A Hog on Ice and Other Curious Expressions

1. Breakable graphs. (40 PTS.)

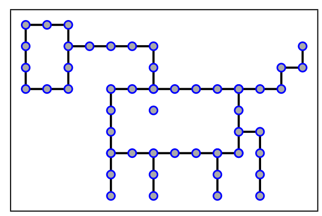
In the following, let c_1, c_2 be some absolute, sufficiently large constants. Consider a graph $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ with n vertices. A subset $Y \subseteq \mathsf{V}$ is en ***isolator*** if:

(I) In the graph $\mathsf{G} \setminus Y$ (i.e., the induced graph on $\mathsf{V} \setminus Y$) is disconnected, and every connected component of this graph has at most $(3/4)n$ vertices.

(II) $|Y| \leq c_2 |n|^{2/3}$ (i.e., Y is significantly smaller than n).

(A) (10 PTS.)

A ***grid*** graph is a graph where the vertices are points (x, y) in the plane, where x, y are integer numbers, and two vertices (x, y) and (x', y') can be connected only if $|x - x'| + |y - y'| = 1$. See picture on the right. Prove that there is always an isolator in such a graph. Your proof must be self contained, elementary and *short*.



(Hint: Start thinking about the case where the vertices of G are contained in $[\![N]\!] \times [\![N]\!]$, where $N = 4\left\lceil n^{2/3} \right\rceil$ and $[\![N]\!] = \{1, \ldots, N\}$. Then solve the case the vertices of G are contained in $[\![N]\!] \times [\![n]\!]$, and finally solve the general case where the vertices are contained in $[\![n]\!] \times [\![n]\!]$.)

(B) (10 PTS.) A graph is ***breakable***, if for any subset $X \subseteq \mathsf{V}$, of size at least c_1, we have that there is an isolator in the induced graph G_X, and furthermore the isolator can be computed in polynomial time in X.

Prove that in a breakable graph, there is always a vertex of constant degree.

(This part is pretty hard, so do not be surprised if you can not do this part. If can not do this part, just assume it is correct, and continue to the next part of the question.)

(C) (10 PTS.) Given a breakable graph G, provide a polynomial time constant approximation algorithm for the largest independent set in G.

(D) (10 PTS.) For an arbitrary fixed $\varepsilon > 0$, provide a polynomial time algorithm that computes $(1 - \varepsilon)$-approximation to the largest independent set in a breakable graph. What is the running time of your algorithm? Prove both the bound on the running time of your algorithm, and the quality of approximation it provides.

(This part is also hard, do not be surprised if you can not do this part.)

2. Greedy algorithm does not work for coloring. Really. (30 PTS.)

Let G be a graph defined over n vertices, and let the vertices be ordered: $v_1, \ldots, v_n$. Let G_i be the induced subgraph of G on $v_1, \ldots, v_i$. Formally, $\mathsf{G}_i = (\mathsf{V}_i, E_i)$, where $\mathsf{V}_i = \{v_1, \ldots, v_i\}$ and

$$\mathsf{E}_i = \left\{ uv \in \mathsf{E} \;\middle|\; u, v \in \mathsf{V}_i \text{ and } uv \in \mathsf{E}(G) \right\}.$$

The greedy coloring algorithm, colors the vertices, one by one, according to their ordering. Let k_i denote the number of colors the algorithm uses to color the first i vertices.

In the ith iteration, the algorithm considers v_i in the graph G_i. If all the neighbors of v_i in G_i are using all the k_{i-1} colors used to color G_{i-1}, the algorithm introduces a new color (i.e., $k_i = k_{i-1} + 1$) and assigns it to v_i. Otherwise, it assign v_i one of the colors $1, \ldots, k_{i-1}$ (i.e., $k_i = k_{i-1}$).

Give an example of a graph G with n vertices, and an ordering of its vertices, such that even if G can be colored using $O(1)$ (in fact, it is possible to do this with two) colors, the greedy algorithm would color it with $\Omega(n)$ colors. (Hint: consider an ordering where the first two vertices are not connected.)

3. Maximum Clique (30 PTS.)

Let $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ be an undirected graph. For any $k \geq 1$, define $\mathsf{G}^{(k)}$ to be the undirected graph $\left(\mathsf{V}^{(k)}, \mathsf{E}^{(k)}\right)$, where $\mathsf{V}^{(k)} = \mathsf{V} \times \mathsf{V} \times \cdots \times \mathsf{V}$ is the set of all ordered k-tuples of vertices from V and $\mathsf{E}^{(k)}$ is defined so that $(v_1, v_2, ..., v_k)$ is adjacent to $(w_1, w_2, ..., w_k)$ if and only if for each i (for $i = 1, \ldots, k$) either vertex v_i is adjacent to w_i in G, or else $v_i = w_i$.

(A) (10 PTS.) Prove that the size of the maximum clique in $\mathsf{G}^{(k)}$ is equal to the kth power of the size of the maximum clique in G. That is, if the largest clique in G has size α, then the largest clique in $\mathsf{G}^{(k)}$ is α^k, and vice versa.

(B) (10 PTS.) Show an algorithm that is given a clique of size β in $\mathsf{G}^{(k)}$ and outputs a clique of size $\left\lceil \beta^{1/k} \right\rceil$ in G.

(C) (5 PTS.) Argue that if there is an c-approximation algorithm for maximum clique (i.e., it returns in polynomial time a clique of size $\geq \mathrm{opt}/c$) then there is a polynomial time $c^{1/k}$-approximation algorithm for maximum clique, for any constant k. What is the running time of your algorithm, if the running time of the original algorithm is $T(n)$. (Hint: use (A) and (B).)

(D) (5 PTS.) Prove that if there is a constant approximation algorithm for finding a maximum-size clique, then there is a polynomial time approximation scheme for the problem.[1]

[1] Can one prove that there is FPTAS in this case? I do not know.