HW 9

Due on **Wednesday, November 3, 2021 at 10am**

CS 473: Algorithms, Fall 2021 Version: **1.0**

Submission guidelines and policies as in homework 1.

25 (100 PTS.) How many colors do we really need?

25.A. (30 PTS.) Prove that a graph G with a chromatic number $k > 2$ (i.e., k is the minimal number of colors needed to color G), must have $\Omega(k^2)$ edges.

25.B. (30 PTS.) Prove that a graph G with m edges can be colored using $4\sqrt{m}$ colors (Hint: If a vertex v has "low" degree, then first color $G \setminus v$ recursively, and then color v.)

25.C. (40 PTS.) Describe a polynomial time algorithm that given a graph G, which is 3-colorable, it computes a coloring of G using, say, at most $O(\sqrt{n})$ colors. (Hint: Let v be a vertex, and let $N = N(v)$ be its neighbors. The induced subgraph $G_{N(v)}$ is two-colorable.)

26 (100 PTS.) Scheduling.

You are given a graph $\mathsf{G} = (\mathsf{V}, \mathsf{E})$ with n vertices and m edges. An edge between two vertices represents a conflict. Given k, describe an algorithm, as efficient as possible, that outputs a partition of V into k disjoint sets $\mathsf{V}_1, \ldots, \mathsf{V}_k$ with a "few" active conflicts. A conflict $uv \in \mathsf{E}$ is ***active*** if u and v belong to the same set V_j, for some j.

26.A. (50 PTS.) Describe a deterministic algorithm (as fast as possible) for this problem, that outputs a partition with at most m/k active conflicts. Prove its correctness and bound its running time.

26.B. (50 PTS.) Describe a faster randomized algorithm, that in expectation generates at most m/k active conflicts. Prove the correctness of your algorithm, and bound its running time.

27 (100 PTS.) Stab these triangles.

You are given a set $\mathcal{T}$ of m triangles in the plane, and a set P of n points in the plane. The task at hand is to pick a minimum size set $X \subseteq P$, such that for all $\triangle \in \mathcal{T}$, we have that $\triangle \cap X \neq \emptyset$. Let ζ be the size of the optimal (i.e., minimum) set that has this property.

Assume that for all $p \in P$, at most α triangles of $\mathcal{T}$ contains p, where α is some number (potentially much smaller than m).

Consider the greedy algorithm that at the ith iteration adds to the solution the point that stabs the largest number of triangles not stabbed yet (let s_i be this number of triangles).

27.A. (10 PTS.) Prove that $s_1 \geq s_2 \geq \cdots$.

27.B. (30 PTS.) The ***epoch*** starting at iteration u, is a maximum range of iterations $u, u+1, \ldots, v$, such that $s_v \geq s_u/2$. Give a bound, as tight as tight as possible, on $v - u + 1$ (i.e., the length of the epoch) in terms of ζ.

27.C. (30 PTS.) Provide an upper bound, as small as possible, on the minimum number of epochs needed to cover all the iterations performed by the algorithm.

27.D. (30 PTS.) Using **27.B.** and **27.C.** provide an upper bound, as tight as possible, on the size of the solution output by the algorithm, as a function of ζ and α.