COMP 5711: Advanced Algorithm
Assignment 2

Fixed Parameter Algorithms (KT Ch 10)

Problem 1 (20pts) We claimed that the Hitting Set Problem was NP-complete. To recap the definitions, consider a set $A = \{a_l.....a_n\}$ and a collection $B_1, B_2, \ldots, B_m$ of subsets of A. We say that a set $H \subseteq A$ is a *hitting set* for the collection $B_1, B_2, \ldots, B_m$ if H contains at least one element from each B_i, that is, if $H \cap B_i$ is not empty for each i. (So H "hits" all the sets B_i.)

Now suppose we are given an instance of this problem, and we'd like to determine whether there is a hitting set for the collection of size at most k. Furthermore suppose that each set B_i has at most c elements, for a constant c. Give an algorithm that solves this problem with a running time of the form $O(f(c,k) \cdot p(n,m))$, where $p(\cdot)$ is a polynomial function, and $f(\cdot)$ is an arbitrary function that depends only on c and k, not on n or m.

Problem 7 (25pts) The *chromatic number* of a graph is the minimum k such that it has a k-coloring, i.e., each vertex is assigned one of k colors such that no two adjacent vertices share the same color.

(a) Show that the chromatic number of a graph G is at most $w + 1$, where w is the tree-width of G.

(b) Give an algorithm with running time $f(w) \cdot \text{poly}(n)$ to find the chromatic number of a graph G, where n is the number of vertices in G and w is its tree-width. You may assume that a tree decomposition of G is already given.

Randomized Algorithms (KT Ch 13)

Problem 1 (20 pts) Suppose we are given a graph $G = (V, E)$, and we want to cover each node with one of three colors. We say that an edge (u, v) is satisfied if the colors assigned to u and v are different. Consider a 3-coloring that maximizes the number of satisfied edges, and let c^* denote this number. Give a polynomial-time algorithm that produces a 3-coloring that satisfies at least $\frac{2}{3}c^*$ edges with constant probability. [Hint: $c^* \leq m$ where m is the number of edges in G.]

Problem 7 (30 pts) We have designed in class an (7/8)-approximation algorithm for the MAX-3SAT problem. now, consider the general MAX SAT problem, in which each clause can consists of any number of terms. We require that all variables in a single clause are distinct.

(a) Suppose as in MAX-3SAT, we assign each variable independently to true or false with probability 1/2 each. Show that the expected number of clauses satisfied is at least $k/2$, where k is the total number of clauses. Give an example to show that there are MAX SAT instances such that no assignment satisfies more than $k/2$ clauses.

(b) For a clause with a single variable (e.g., x_1 or $\overline{x_2}$), there is only one way to satisfy it. If there are two such clauses that are negation of each other, then this is a direct contradiction and cannot be both satisfied. Assume that in the input instance there are no such conflicting clauses. Modify the algorithm above to so that the expected number of satisfied clauses is at least $0.6k$.

(c) Give a randomized algorithm for the general MAX SAT problem that satisfies at least $0.6 \cdot \mathsf{OPT}$ clauses in expectation, where OPT is the maximum number of clauses satisfied by an optimal assignment.

Problem 11 (25 pts) Suppose you assign k jobs to k machines, each job to any of the k machines uniformly and independently at random.

(a) Let $N(k)$ be the expected number of machines that do not receive any jobs, so that $N(k)/k$ is the expected fraction of machines with nothing to do. What is $\lim_{k\to\infty} N(k)/k$? Give the exact value, without using asymptotic notation.

(b) Suppose that each machine will only do one job if more than one has been assigned to it, and reject the rest. Let $R(k)$ be the expected number of rejected jobs. What is $\lim_{k\to\infty} R(k)/k$?

(c) Now assume that each machine is able do two jobs assigned to it, and reject the rest. Let $R_2(k)$ be the expected number of rejected jobs. What is $\lim_{k\to\infty} R_2(k)/k$?