**Due: Wed, December 2, beginning of lecture**

NOTE: Each problem set only counts 5% of your mark, but it is important to do your own work (but see below). Similar questions will appear on the first term test. You may consult with others concerning the general approach for solving problems on assignments, but you must write up all solutions entirely on your own. Anything else is *plagiarism*, and is subject to the University's Code of Behavior. You will receive 1/5 points for any (non bonus) question/subquestion for which you say "I do not know how to answer this question". You will receive .5/5 points if you just leave the question blank.
Advice: Do NOT spend an excessive amount of time on any question and especially not on a bonus question. If you wish to spend "free time" thinking about (say) bonus questions that is fine but you should not sacrifice time needed for other courses.

1. (a) Find an example showing that the locality gap of the single flip local search algorithm for max cut can be as bad as 2 (or 1/2 is you like approximation ratios that are $\leq 1$).

   (b) Find an example showing that the locality gap of a Hamming distance $d$ (i.e. flipping at most $d$ nodes) local search algorithm for max cut can be close to 2 assuming $d$ is any fixed constant.

2. Consider the weighted exact Max-2-Sat problem. Supposed we modify the greedy local search algorithm by expanding the local neighbourhood of a truth assignment $\tau$ to also contain the complement of $\tau$ (that is, the truth assignment that compliments all truth values) in addition to the complimenting of a single variable. Show that this modified local search algorithm now achieves at least $\frac{3}{4}$ of the total weight of all clauses.
   Hint: consider and augment the analysis of the greedy local search algorithm.

3. (Bonus and open question): Can you find an efficiently searchable neighbourhood so that exact Max-3-Sat has locality gap "close" to 8/7 for greedy local search? Note that the analysis for the greedy Max-2-Sat Hamming distance 1 local search algorithm immediately gives a locality gap of 4/3 for Max-3-Sat.

4. Suppose we have a flow network $\mathcal{F} = (G, s, t, c)$ with integral capacities $\{c_e | e \in E\}$ and suppose we are given an optimal integral flow $f$ with $val(f) > 0$ in $\mathcal{F}$.
   In the following questions keep in mind min cuts and also residual graphs.

   (a) Show that there is some (not necessarily unique) edge $e \in E$ such that changing the capacity of $e$ to $c'(e) = c(e) - 1$ will decrease the max flow by one unit.

   (b) Show how to efficiently determine (say in time $O(|E|)$) whether or not there is a single edge $e$ such that increasing the capacity of $e$ by one unit will increase the max flow to $val(f) + 1$. That is, we want to increase the flow by one unit while only increasing the capacity of a single edge by one unit.

(c) Show how to efficiently determine (say in time $O(|E|\log n)$)) a minimum number of edges $E' \subset E$ such that increasing capacity of each $e \in E$ by one unit will increase the max flow to $val(f) + 1$.

5. Consider the following *4 to 2 frequency set cover problem*: We are given a collection of sets $\mathcal{S} = \{S_1, \ldots, S_m\}$ for $S_i \subseteq U$ with the property that every $u \in U$ occurs in exactly four different sets $S_i$ in $\mathcal{S}$. There is also a cost function $c : \mathcal{S} \rightarrow \mathbb{R}^{\geq \not{0}}$ and we let $c_i$ denote the cost of set $S_i$. A feasible solution is a sub-collection $\mathcal{S}' \subseteq \mathcal{S}$ such that every $u \in U$ occurs in at least two different sets $S_i$ in $\mathcal{S}'$. The goal is to find a feasible solution $\mathcal{S}'$ so as to minimize the cost $c(\mathcal{S}') = \sum_{S_i \in \mathcal{S}'} c_i$.

   (a) Formulate the *4 to 2 frequency set cover problem* as a $\{0, 1\}$ IP

   (b) Show how to use LP relaxation + rounding to obtain a 3-approximation algorithm. Explain why your rounded solution is a feasible solution to the IP and why it provides a 3-approximation.

6. Consider again weighted exact Max-$K$-Sat problem and the following simple randomized algorithm executing on a propositional CNF formula $F = C_1 \wedge C_2 \ldots \wedge C_r$ with exactly $K$ literals per clause and $n$ propositional variables $x_1, \ldots, x_n$:

   For i = 1 ... n
   
   Randomly with probability $\frac{1}{2}$, set $\tau(x_i) = true$ (resp. false)
   
   End For

   (a) Show that the expected weight of satisfied clauses in $F(\tau)$ is at least $(\frac{2^K-1}{2^K} \cdot W)$ where $W = \sum_{i=1}^{n} w_i$ and $w_i$ is the weight of the $i^{th}$ clause.

   (b) Indicate how you would use the method of conditional expectations to de-randomize this algorithm achieving a deterministic algorithm where the weight of satisfied clauses is at least $(\frac{2^K-1}{2^K} \cdot W)$.

   (c) Suppose we provide the following representation of the input $F$. We represent each propositional variable $x_i$ by fully specifying (i.e. giving the $K$ literals) all the clauses in which the variable $x_i$ occurs (either complemented or not complemented). Indicate how the de-randomized algorithm can be viewed as a greedy (even online) algorithm. More specifically, show how to set each propositional variable in a one pass algorithm where any ordering of the variables suffcies and the decision as to the setting of the truth value for a variable only depends on the initial representation of the variable and the setting of the previous variables. In terms of $K, r$, and $n$ estimate the time for this online algorithm. For unweighted formulas, how does this time bound compare with the oblivious (or non-oblivious) local search algorithm given the same input representation?