CS41 Lab 13

The lab this week provides more practice with randomized algorithms. If you missed the lab last week, you should use any extra time to go back and make sure you understand those lab problems.

1. **(K&T 13.7)** We designed an approximation algorithm to within a factor of 8/7 for the MAX-3-SAT problem, where we assumed that each clause has terms associated with three different variables. In this problem, we will consider the analogous MAX-SAT problem: given a set of clauses $c_1, \ldots, c_k$ over a set of variables $x_1, \ldots, x_n$, find a truth assignment satisfying as many of the clauses as possible. Each clause has at least one term in it, and all the variables in a single clause are distinct, but otherwise we do not make any assumptions on the length of the clauses. There may be clauses that have a lot of variables, and others may have just a single variable.

   (a) First consider the randomized approximation algorithm we used for MAX-3-SAT, setting each variable to true or false with probability 1/2 each. Show that the expected number of clauses satisfied by this random assignment is at least $k/2$, that is, at least half the clauses are satisfied *in expectation.* Give an example to show that there are MAX-SAT instances such that no assignment satisfies more than half of the clauses.

   (b) If we have a clause that consists only of a single term (for example just x_1 or just $\overline{x_2}$), then there is only a single way to satisfy it: we need to set the corresponding variable in the appropriate way. If we have two clauses such that one consists of just the term x_i, and the other consists of just the negated term $\overline{x_i}$, then this is a pretty direct contradiction.

   Assume that our instance has no such pair of "conflicting clauses"; that is, for no variable x_i do we have both the clause x_i and the clause $\overline{x_i}$. Modify the randomized procedure from above to improve the approximation factor from 2 to at worst 10/6. That is, change the algorithm so that the expected number of clauses satisfied by the process is at least $6k/10$.

   (c) Give a randomized polynomial-time algorithm for the general MAX-SAT problem, so that the expected number of clauses satisfied by the algorithm is at least a 6/10 fraction of the maximum possible.

   (Note that, by the example in part (a), there are instances where one cannot satisfy more than $k/2$ clauses; the point here is that we'd still like an efficient algorithm that, *in expectation*, can satisfy a 6/10 fraction *of the maximum that can be satisfied by an optimal assignment.*)