CS 466/666, Fall 2006 due Friday November 10
A. Lubiw

ASSIGNMENT 3

Do not copy from others, and acknowledge your sources.

1. [10 marks] This question is about generating random permutations of $\{1, \ldots, n\}$. Consider the following two methods:

   (a) **Method 1.** Set up an array A, with $A[i] = i$. Then for i ranging from 2 to n, swap $A[i]$ and $A[\text{rand}[1, i]]$, where $\text{rand}[a, b]$ returns a random number in the range $a, \ldots, b$.

   (b) **Method 2.** The heart of this method is to generate n random values from the interval $[0, 1]$, sort these values, and return the permutation that sorts them. The one refinement is that the random values should be generated bit by bit in a lazy fashion—when the sorting algorithm asks to compare two values, we see if the bits generated so far distinguish the numbers, and if not, we generate further random bits until the numbers are distinct.

   Prove that each of these methods correctly finds a random permutation (i.e. that each permutation is equally likely). Analyze the two methods in terms of running time and number of random bits required. Be careful to distinguish if your analyses are worst case or expected case, and if you are counting bit operations or using a unit-cost RAM.

2. [20 marks] Give a randomized algorithm with expected running time $O(n)$ to find the smallest disc enclosing a given set of n points in the plane. You may assume that no 3 points lie on a line and no 4 points lie on a circle. Hints: Use the technique of random reordering to add the points one by one. The major issue is what to do when a new point p is not in the current smallest disc. Prove that in this case the new smallest disc goes through p. Design a subroutine to find the smallest disc enclosing a given set of points *and* having one given point on its boundary. This begins to sound recursive (i.e. you may next want a subroutine to find the smallest disc enclosing a given set of point *and* having *two* given points on its boundary) but note that once there are 3 points required to be on the boundary of a disc then the disc is uniquely determined. To analyze your algorithm (and the subroutines) use backwards analysis. The smallest disc will always go through 3 points, and the probability that removing a random point alters the disc is at most $3/n$. Even with these hints, there is a substantial amount of work to be done for this problem. [Marks: 10 for algorithm and correctness proof; 10 for analysis]

3. [10 marks] Consider the Satisfiability (SAT) problem: given a Boolean formula in conjunctive normal form (CNF), with n variables and m clauses, find a truth value assignment for the variables that makes the formula True. For example, the CNF formula $(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3) \wedge (x_2 \vee \bar{x}_3)$ is satisfiable using the truth value assignment $x_1 = 1$, $x_2 = 0$, $x_3 = 0$. CNF means the formula is a conjunction (AND) of clauses, and each clause is the disjunction (OR) of literals, and each literal is a variable x_i or the negation of a variable $\bar{x}_i$. In the above example, there are 4 clauses and the clause $(\bar{x}_1 \vee \bar{x}_2 \vee \bar{x}_3)$ has 3 literals.

   In 1991 Papadimitriou suggested the following Monte Carlo algorithm for SAT:

```
choose a random truth-value assignment A
repeat t times
  if A satsifes the formula then return YES
  pick an unsatisfied clause C
  randomly choose one of its literals a
  flip a's value in A
end
return NO (probably)
```

For this problem you will analyze this algorithm for the special case of 2-SAT in which every clause has exactly two literals. You will need the following result about random walks on the line:

Theorem Suppose you start at some value $i \in [0..n]$ and at each step you add 1 to i with probability $\frac{1}{2}$ and subtract 1 with probability $\frac{1}{2}$, except that when $i = n$ you always subtract 1. Then the expected number of steps until you reach $i = 0$ is at most n^2, and the probability of taking more steps is less than $\frac{1}{2}$.

(a) Consider a satisfiable CNF 2-SAT formula F. Let A^* be a satisfying truth value assignment. The algorithm starts by choosing a random truth value assignment A. Let i be the number of variables with different values in A and A^*. Prove that each pass through the repeat loop decreases i with probability at least $\frac{1}{2}$.

(b) Using part (a) show that setting $t = n^2$ gives a Monte Carlo algorithm that runs in polynomial time and has probability of error less than $\frac{1}{2}$.

4. BONUS question [5 marks] Prove the theorem about random walks on the line. Hints: Let t_i be the expected number of steps starting from initial value i. Write an equation relating t_i to t_{i-1} and t_{i+1}. Take care of t_0 and t_n. Prove that this this system of $n + 1$ equations in $n + 1$ unknowns has solution $t_i = 2in - i^2$.