Take Home Final Exam
December 11, 1998
Due: Wednesday, December 16, 1998, 12:00

Q1: **Simple Polygon**
Given n points in the plane, construct a simple polygon having the n points as its vertices.

(a) Give an $O(n \log n)$ algorithm to construct a simple polygon containing the n points. The algorithm should also handle input where three points are on a line.

Q2: **Nesting Boxes**
A d-dimensional box with dimension $(x_1, x_2, \ldots, x_d)$ *nests* within another box with dimensions $(y_1, y_2, \ldots, y_d)$ if there exists a permutation $\pi : [d] \to [d]$ such that $x_{\pi(1)} < y_1, x_{\pi(2)} < y_2, \ldots, x_{\pi(d)} < y_d$.

(a) Prove that the nesting relation is transitive.

(b) Describe an efficient method to determine whether or not one d-dimensional box nests inside another.

(c) Suppose that n d-dimensional boxes $\{B_1, B_2, \ldots, B_n\}$ are given. Describe an efficient algorithm to determine the longest sequence $B_{i_1}, B_{i_2}, \ldots B_{i_k}$ such that B_{i_j} nests within $B_{i_{j+1}}$ for all $1 \leq j < k$. Express the running time of your algorithm in terms of n and d.

Q3: **Dijkstra's Algorithm**
Let $x_1, x_2, \ldots, x_d$ be uniformly and independently chosen real values from the interval $[0..1]$. Let X_i, $1 \leq i \leq d$, be random variables given by:

$$X_i = \begin{cases} 1 & \text{if } x_i < x_j \text{ for all } 1 \leq j < i \text{ ,} \\ 0 & \text{otherwise .} \end{cases}$$

Note that $\Pr[X_1 = 1] = 1$. Let $Y = \sum_{i=1}^{d} X_i$.

(a) Prove $\mathrm{E}[Y] = O(\log d)$.

Let $G = (V, E)$ be a directed graph with $|V| = n$ and $|E| = m$. For $v \in V$, $d(v)$ denotes the in-degree of v. To each edge $(u, v) \in E$ is assigned a weight $\ell(u, v)$ chosen uniformly from $[0..1]$. All edge weights are chosen independently.

Consider Dijkstra's single source shortest path algorithm [Kozen, Lecture 5, Algorithm 5.1] when applied to G.

(b) Let $v \in V$. Prove that the expected number of times $D(v)$ changes (decreases) is $O(\log d(v))$.

(c) Prove that total expected number of changes to $D(\cdot)$ is $O(n(1+\log \frac{m}{n}))$, *i.e.,* the expected number of calls to **decrement** is $O(n(1 + \log \frac{m}{n}))$.

Q4: **Satisfiability**
A boolean formula $F(x_1, \ldots, x_n) = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ is an instance of the 2-SAT problem if each clause C_i is the $\vee$ of two distinct variables, which might or might not be negated.

E.g. $F = (x_1 \vee \overline{x_4}) \wedge (x_2 \vee x_4) \wedge (\overline{x_1} \vee \overline{x_3}) \wedge (\overline{x_2} \vee x_1) \wedge (x_1 \vee \overline{x_3})$ is an instance of 2-SAT.

A formula is *satisfiable* if there exists a truth assignment to the variables such that the formula evaluates to true.

(a) Give a polynomial time algorithm to determine whether a 2-SAT instance is satisfiable.

(b) Give a randomized algorithm whose expected number of satisfied clauses is at least 3/4 of the total number of clauses, whether or not the formula is satisfiable.