G22.3520: Honors Analysis of Algorithms

Final Exam, Dec 23, 2009, 12:30-4:00pm

- This is a three and a half hour exam.
- There are six questions, worth 10 points each. Answer all questions and all their subparts.
- Your answer to a subpart of a question may use (without repeating) your answer to a previous subpart.
- Please print your name and SID on the front of the envelope only (not on the exam booklets).
- Please answer each question in a separate booklet, and number each booklet according to the question.
- Read the questions carefully. Keep your answers legible, and brief but precise.
- Assume standard results, unless asked otherwise.

Best of luck!

Problem 1

Let $G = (V, E)$ be a directed acyclic graph where edges are assigned non-negative weights. You can assume that $V = \{v_1, v_2, \ldots, v_n\}$ and if $(v_i, v_j) \in E$ then $i < j$. Let the weight of edge (v_i, v_j) be denoted as w_{ij}. The weight of a (directed) path in the graph is defined as the product of the weights of all the edges on that path. Assume an adjacency list representation of G.

1. Give $O(|V| + |E|)$ time algorithm to compute the sum of the weights of all paths from v_1 to v_n.
2. Let $1 < i_0 < n$ be a fixed index. Give $O(|V| + |E|)$ time algorithm to compute the sum of the weights of all paths from v_1 to v_n passing through v_{i_0}.
3. Let $1 < i_0 < j_0 < n$ be two fixed indices. Give $O(|V| + |E|)$ time algorithm to compute the sum of the weights of all paths from v_1 to v_n passing through v_{i_0} but not passing through v_{j_0}.

Problem 2

A Non-deterministic Finite Automaton N is a 5-tuple $N = (Q, \Sigma, q^s, F, \delta)$ where Q is a finite set of states, Σ is a finite alphabet, $q^s \in Q$ is the start state, $F \subseteq Q$ is the set of accept states, and $\delta : Q \times \Sigma \mapsto \mathcal{P}(Q)$ is the transition function. The language accepted by an NFA N is denoted as $L(N)$. A Deterministic Finite Automaton is an NFA as above except that the transition function is $\delta : Q \times \Sigma \mapsto Q$.

1. Given two NFAs $N_1 = (Q_1, \Sigma, q_1^s, F_1, \delta_1)$ and $N_2 = (Q_2, \Sigma, q_2^s, F_2, \delta_2)$, show how to construct an NFA N such that the set of its states is $Q_1 \times Q_2$ and $L(N) = L(N_1) \cap L(N_2)$.
2. Given an NFA $N = (Q, \Sigma, q^s, F, \delta)$, show how to construct a DFA D such that $L(D) = L(N)$.
3. Given an NFA $N = (Q, \Sigma, q^s, F, \delta)$, show how to construct an NFA N' such that $L(N') = \overline{L(N)}$ where $\overline{L(N)}$ denotes the complement of the language $L(N)$.

Note:

- $\mathcal{P}(X)$ denotes the power set of X, i.e. the set of all subsets of X.
- The cartesian product of two sets X and Y is defined as $X \times Y := \{(x, y) \mid x \in X,\ y \in Y\}$.
- The definition of NFA above does not allow the so-called ε-moves. So ignore ε-moves.
- Proofs of correctness are not necessary. Give a formal description of NFAs/DFAs that you may construct. Partial credit will be given to description in words.

Problem 3

Note that a *tree* is a connected graph with no cycles. Show that for any tree T with $2n+1$ vertices, there exists a vertex $v \in T$ such that after removing v (and all edges incident on it), each connected component of $T \setminus \{v\}$ has at most n vertices.

Hint: A simple greedy-style strategy actually finds such a vertex.

Problem 4

We need to maintain a collection $\mathcal{C}$ of rooted trees such that the following operations can be implemented efficiently:

- BREAK: Given a tree $T \in \mathcal{C}$, remove T from the collection $\mathcal{C}$. Delete its root and add all the subtrees of the root as (distinct) trees in the collection $\mathcal{C}$. The root is also added to $\mathcal{C}$ as a single-node tree. The cost of this operation is k if k was the degree of the root.
- UNION: Given two distinct trees $T_1, T_2 \in \mathcal{C}$, remove them from $\mathcal{C}$, construct a tree T whose set of nodes is the union of the sets of nodes of T_1 and T_2, and then add T to collection $\mathcal{C}$. The cost of this operation is 1.

Assume that initially $\mathcal{C}$ consists of n single-node trees.

1. Give an implementation that ensures that for any tree T in the collection, the height of T is at most $O(\log |T|)$ where $|T|$ is the number of nodes in T. You need to describe how to implement the UNION operation and prove that the upper bound on the height works.

   Assume that the height of a single node tree is zero.
2. Prove that for a sequence of m BREAK and UNION operations, the total cost is $O(m+n)$.

Problem 5

A *directed Hamiltonian cycle* in an n vertex directed graph is a directed cycle containing exactly n edges that includes every vertex exactly once. You can assume that the DIRECTED HAMILTONIAN CYCLE (DHC) problem is NP-complete:

$$\text{DHC} := \Big\{ \langle G' \rangle \mid G'(V', E') \text{ is a directed graph that has a directed Hamiltonian cycle} \Big\}.$$

An edge coloring of a directed graph $G(V,E)$ is a map $\pi : E \mapsto \{\mathsf{Red}, \mathsf{Blue}\}$. Let the AMERICAN HAMILTONIAN CYCLE problem (AHC) be the following:

$$\begin{aligned} \text{AHC} \quad := \quad \Big\{ \langle G, \pi \rangle \mid\ & G(V,E) \text{ is a directed graph with an even number of vertices along with} \\ & \text{an edge coloring } \pi \text{ and } G \text{ has a directed Hamiltonian cycle with no} \\ & \text{two consecutive edges of the same color} \Big\}. \end{aligned}$$

Show that AHC is NP-complete.

Problem 6

Part I:

Suppose $z \in \{0,1\}^k$ and z_i denotes the i^{th} bit of z. For a subset $S \subseteq \{1,2,\ldots,k\}$, $S \neq \emptyset$, let

$$f_z(S) := \bigoplus_{i \in S} z_i.$$

In words, $f_z(S)$ is the XOR of all bits of z in the subset S. Now think of $z \in \{0,1\}^k$ as a string chosen uniformly at random (i.e. every bit of z is independently set to 0 or 1 with probability $\frac{1}{2}$ each). Answer the following questions along with a justification.

1. Fix a subset $S \subseteq \{1,2,\ldots,k\}$, $S \neq \emptyset$. What are the following probabilities?

$$\Pr_z[f_z(S) = 0], \qquad \Pr_z[f_z(S) = 1].$$

2. Fix two distinct subsets $S, T \subseteq \{1,2,\ldots,k\}$, $S \neq \emptyset$, $T \neq \emptyset$, $S \neq T$. What are the following probabilities?

$$\Pr_z[f_z(S) = 0 \text{ and } f_z(T) = 0],$$
$$\Pr_z[f_z(S) = 0 \text{ and } f_z(T) = 1],$$
$$\Pr_z[f_z(S) = 1 \text{ and } f_z(T) = 0],$$
$$\Pr_z[f_z(S) = 1 \text{ and } f_z(T) = 1].$$

Part II:

Let $G(V,E)$ be an undirected graph where $|V| = n = 2^k - 1$ and V is identified with the set of all non-empty subsets of $\{1,2,\ldots,k\}$, i.e.

$$V := \{S \mid S \subseteq \{1,2,\ldots,k\},\ S \neq \emptyset\}.$$

For any choice of $z \in \{0,1\}^k$, we get as above a function $f_z(\cdot) : V \mapsto \{0,1\}$ which can be thought of as a partition of V into two parts. Let $\mathrm{CUT}[f_z]$ denote the number of edges of $G(V,E)$ that are cut by this partition. Prove that

$$\mathbb{E}_z\Big[\ \mathrm{CUT}[f_z]\ \Big] = \frac{|E|}{2}.$$

where $\mathbb{E}_z[\cdot]$ denotes the expectation over the random choice of z.

Part III:

Using Part II or otherwise, design a deterministic polynomial (in n) time algorithm that finds a partition of the graph $G(V,E)$ that cuts at least $\frac{|E|}{2}$ edges (the graph is as in Part II).