ACO Comprehensive Exam **March 2004** **Name** ______________________________

1. Graph Theory

A subgraph H of a connected graph G is called non-separating if $G - V(H)$ is connected. A path with ends u and v is also called a u-v path.

(1) Let G be a connected graph and u, v be two distinct vertices of G. Let P be a u-v path in G and let W be a component of $G - V(P)$. Suppose that, for any $S \subset V(G)$ with $|S| < 3$, every component of $G - S$ contains a vertex from $V(W) \cup \{u, v\}$. Show that there is a u-v path Q in $G - V(W)$ such that Q is non-separating.

(2) Show that the path Q in (1) may further be chosen to be induced.

(3) Show that the conclusion in (2) implies the following theorem of Tutte: Let G be a 3-connected graph, $e = uv \in E(G)$, and $w \in V(G) - \{u, v\}$; then there is an induced non-separating cycle in G through e and avoiding w.

(**Note:** If you are unable to prove (1), you may assume that result in proving (2) and (3).)

2. Probability

Let $Z_1, Z_2, \ldots$ be independent random variables such that

$$Z_n = \begin{cases} a_n & \text{with probability } (1/2)n^{-2}, \\ 0 & \text{with probability } 1 - n^{-2}, \text{ and} \\ -a_n & \text{with probability } (1/2)n^{-2} \end{cases}$$

where $a_1 = 2$ and $a_n = 4 \times \sum_{j=1}^{n-1} a_j$.

(a) Show that $Y_n = \sum_{j=1}^{n} Z_j$ defines a martingale.

(b) Show that $Y = \lim Y_n$ exists almost surely, but that

(c) there exists no M such that $E|Y_n| \leq M$ for all n.

Hint for (c): First argue that $|Y_n| \geq (1/2)a_n$ if and only if $|Z_n| = a_n$.

ACO Comprehensive Exam March 2004 Name ______________________________

3. Analysis of Algorithms

(a) Given an undirected graph $G = (V, E)$, the *maximum cut problem* asks for a partition of V into sets S and $\overline{S}$ so that the number of edges running between these sets is maximized. Consider the following randomized algorithm for this problem. Initialize by putting an arbitrary vertex v_1 in A and v_2 in B. For each of the remaining vertices flip a fair coin to decide if it goes in set A or B. Show that the expected size of cut found is at least OPT/2. Derandomize this algorithm via the method of conditional expectation.

(b) Derive the following two min-max relations from the max-flow min-cut theorem:
Menger's Theorem: Let $G = (V, E)$ be a directed graph with $s, t \in V$. Then, the maximum number of edge-disjoint (vertex-disjoint) $s - t$ paths is equal to the minimum number of edges (vertices) whose removal disconnects s from t.

4. Linear Programming

1. Consider the following linear programming problem in standard form:

$$\min_x \{c^T x : \quad Ax = b, \ x \geq 0\}, \tag{1}$$

where $A \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, and $c \in \mathbb{R}^n$. Let $B \in \mathbb{R}^{m\times m}$ be an invertible submatrix of A, and let $x^* := (x^*_{\mathcal{B}}, x^*_{\mathcal{N}}) = (B^{-1}b, 0)$.

   (a) Suppose that x^* is an optimal solution of (1). Is it true that the associated reduced cost vector $c^T - c^T_{\mathcal{B}} B^{-1} A \geq 0$? Justify your answer.

   (b) Give sufficient conditions for which x^* is the unique optimal solution to (1).

   (c) Give necessary and sufficient conditions for x^* to be an optimal solution to (1).

   (d) Suppose that x^* is a nondegenerate basic optimal solution to (1). Give the dual problem to (1), and prove that the associated dual optimal solution is unique.

   (e) Suppose that x^* is a nondegenerate basic optimal solution with associated dual solution (y^*, s^*). Prove that for all vectors $\delta \in \mathbb{R}^m$ sufficiently small, the optimal solution value to

$$\min\{c^T x : \ Ax = b + \delta, \ x \geq 0\}$$

   is $c^T x^* + \delta^T y^*$.

2. Prove that the system of equations $Ax \leq 0$, $x \neq 0$ is unsolvable if and only if the set $A^T y = c$, $y \geq 0$ is solvable for every vector c.

3. Let $f(x) = \max_{i\in I} a_i^T x + b_i$, where I is a finite set. We define the *ϵ-subdifferential* of f at a point x, written $\partial_\epsilon f(x)$, as

$$\partial_\epsilon f(x) \ := \ \{p : f(y) \geq f(x) + p^T(y - x) - \epsilon, \ \forall \ y \in \mathbb{R}^n\}.$$

Use LP duality to prove that

$$\partial_\epsilon f(x) = \left\{ p = \sum_{i\in I} t_i a_i : \sum_{i\in I} t_i = 1, \sum_{i\in I} t_i(a_i^T x + b_i) \geq f(x) - \epsilon, \ t_i \geq 0 \ \forall \ i \in I \right\}.$$

5. Combinatorial Optimization

Note: You may use well known facts about network flows without proof.

Consider the directed network $G = (N, A)$ with node set N and arc set A. Let u_{ij} be the capacity of arc (i, j) for all $(i, j) \in A$. Given two nodes $s, t \in N$ and an $s - t$ cut $[S, \bar{S}]$, which partitions N into S and $\bar{S}$ with $s \in S$ and $t \in \bar{S}$, let $(S, \bar{S})$ and $(\bar{S}, S)$ denote the set of *forward* and *backward* arcs of $[S, \bar{S}]$ respectively. The capacity of $[S, \bar{S}]$ is defined as $u[S, \bar{S}] = \sum_{(i,j)\in(S,\bar{S})} u_{ij}$. The *minimum cut problem* is to determine an $s - t$ cut of minimum capacity.

Assume that we are given an $s - t$ cut $[S_o, \bar{S}_o]$. Now we can modify the capacity vector u so that $[S_o, \bar{S}_o]$ becomes a minimum cut. Let $G(d)$ denote the network $G = (N, A)$ with capacity vector d. We define

$$\mathbf{P_G} = \min_d \left\{ \sum_{(i,j)\in A} |d_{ij} - u_{ij}| : [S_o, \bar{S}_o] \text{ is a minimum } s - t \text{ cut in } G(d) \right\}.$$

1. Let the network $G'(N, A')$ be obtained from $G = (N, A)$ by deleting the backward arcs of the cut $[S_o, \bar{S}_o]$, that is $A' = A \setminus (\bar{S}_o, S_o)$. Also given a capacity vector d for G, we define the capacity vector d' for G' by $d'_{ij} = d_{ij}$ for all $(i, j) \in A'$. Show that if the capacity vector d' is optimal for $\mathbf{P_{G'}}$ then capacity vector d^* is optimal for $\mathbf{P_G}$, where $d^*_{ij} = d'_{ij}$ for each $(i, j) \in A'$ and $d^*_{ij} = u_{ij}$ for each $(i, j) \in (\bar{S}_o, S_o)$.

2. Let $Excess(S_o, d')$ be the capacity of the cut $[S_o, \bar{S}_o]$ minus the maximum $s - t$ flow in $G'(d')$. Show that the optimal value for $\mathbf{P_{G'}}$ is equal to $Excess(S_o, d')$. How would you determine the optimal capacity vector d^* for $\mathbf{P_G}$?

ACO Comprehensive Exam   March 2004   Name ______________________

6. Algebra

1. (a) Let $(\mathbb{Z}/28\mathbb{Z})^*$ be the group of invertible elements (units) of $\mathbb{Z}/28\mathbb{Z}$. Determine the structure of $(\mathbb{Z}/28\mathbb{Z})^*$ as a direct product of cyclic groups.

   (b) Determine the structure of $(\mathbb{Z}/8\mathbb{Z})^*$ as a direct product of cyclic groups.

2. Let R be a commutative ring with a unit element, and $\varphi : R \to R$ a surjective ring homomorphism. Is φ an isomorphism? Prove or disprove.

3. $\mathbb{Q}$ denotes the rational numbers and $\mathbb{C}$ the complex numbers. If $f(x) \in \mathbb{Q}[x]$ is an irreducible polynomial with roots $\alpha_1, \ldots \alpha_n \in \mathbb{C}$, prove that $\alpha_i - \alpha_j \notin \mathbb{Q}$ for all $i \neq j$.

7. Graph Algorithms

Let v be a vertex of degree two in a multigraph G. By supressing v we mean deleting v and adding an edge joining its neighbors. Please note that this may produce loops or parallel edges. A multigraph is series-parallel if it can be reduced to the graph with no vertices by repeatedly applying the following operations:

(1) suppressing a vertex of degree two,

(2) deleting an edge parallel to another edge,

(3) deleting a loop, and

(4) deleting a vertex of degree at most one.

Design a linear-time algorithm with the following specifications. The input consists of a series-parallel multigraph G, presented as a list of vertices and a list of edges, and a sequence of operations (1)–(4) that reduce G to the null graph. The algorithm should output the size of the maximum independent set of G. Prove correctness of your algorithm, and prove that it runs in linear time. Describe your algorithm from first principles; do not use any general-purpose algorithms as subroutines.

7. Computational Complexity

Two undirected graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ are isomorphic if there is a bijection $\sigma : V_1 \rightarrow V_2$ such that for every pair of vertices $i, j \in V_1$, $(i, j) \in E_1$ iff $(\sigma(i), \sigma(j)) \in E_2$.

The graph isomorphism problem is to decide if two input graphs are isomorphic. The graph isomorphism problem is in $\mathcal{NP}$ and is not known to be decidable in polynomial time.

Define **CAN-EXTEND-GI**: $\{(G, H, \phi) \mid G, H$ are graphs on n vertices, $\phi : \{1, \ldots, i\} \rightarrow \{1, \ldots, n\}$ for some $i \leq n$ and ϕ can be extended to an isomorphism$\}$.

Question: Suppose **CAN-EXTEND-GI** can be decided in polynomial time. Given as input two graphs G and H, show that:

1. It can be decided in polynomial time whether G is isomorphic to H or not.
2. If G is isomorphic to H, an isomorphism can be computed in polynomial time.

ACO Comprehensive Exam **March 2004** **Name** ________________________

7. Randomized Algorithms

Suppose you have a 4-sided (tetrahedral) die, where the probabilities of the four sides are p_1, p_2, p_3, and p_4. Assume that $p_i \geq a$, for each i. How can you obtain estimates $\tilde{p}_i$ such that for each i, $\Pr[|p_i - \tilde{p}_i| > \epsilon p_i] < \delta$, for any choice of constants $0 < a, \epsilon, \delta < 1$? If N is the number of times you need to roll the die, what is the value of N that will give this guarantee? (Of course you should make sure N is not too large.)