Problem Set 11 for CS 170

Formatting Please use the following format for the top of the solution you turn in, with one line per item below (in the order shown below):

<your username on cory.eecs>
<your full name>
`CS170, Spring 2003`
`Homework #11`
`Section` <your section number>
`Partners:` <your list of partners>

Note When asked for an algorithm you must give (1) a brief informal description of the algorithm, (2) a precise description using pseudo-code, (3) an informal argument for termination and correctness of the algorithm, and (4) an analysis of the running time of the algorithm. Be clear about what the input to the algorithm is, how you measure the size of the input, and what constitutes a "step" in your running-time analysis.

Problem 0. [Any questions?] (5 points)

What's the one thing you'd most like to see explained better in lecture or discussion sections? A one-line answer would be appreciated.

(Sometimes we botch the description of some concept, leaving people confused. Sometimes we omit things people would like to hear about. Sometimes the book is very confusing on some point. Here's your chance to tell us what those things were.)

Problem 1. [Bipartite Vertex Cover] (30 points)

A *vertex cover* of a graph $G = (V, E)$ is defined as a subset of vertices $W \in V$ such that for each edge $e \in G$ at least one of the two vertices of e is in W. If a cost $C(v)$ is defined on each vertex $v \in G$, then the cost of a vertex cover W is the sum of the costs of all the vertices in W.

Given a bipartite graph $G = (L \cup R, E)$, find a vertex cover with the minimum cost.

HINT: Use the max-flow/min-cut theorem.

Problem 2. [General Vertex Cover] (15 points)

Now consider the vertex cover problem for general graphs G (not necessarily bipartite). In this problem, we will assume that the cost of every vertex is 1. Define the languages

$$\begin{aligned} \mathit{VERTEXCOVER} &= \{(G,k) : \text{the graph } G \text{ has a vertex cover of cost} \le k\} \\ \mathit{BIPARTITE} &= \{(G,k) : \text{the graph } G \text{ is bipartite}\} \\ \mathit{BIVERCOV} &= \mathit{BIPARTITE} \cap \mathit{VERTEXCOVER} \end{aligned}$$

(a) Is *VERTEXCOVER* a good decision-problem variant of the vertex cover problem for general graphs? Justify your answer.

(b) Is *VERTEXCOVER* in **NP**? Justify your answer.

(c) Is *BIVERCOV* in **NP**? Justify your answer.

Problem 3. [**Number Theory**] (50 points)

Define the languages

$$\begin{aligned} \mathit{PRIMES} &= \{p \in \mathbb{N} : p \text{ is prime}\} \\ \mathit{COMPOSITES} &= \{n \in \mathbb{N} : n \text{ is composite}\} \\ \mathit{FACTORING} &= \{(n,k) : n \text{ has some factor } d \text{ with } 1 < d < k\} \end{aligned}$$

If L is a language and $\overline{L} = \{0,1\}^* \setminus L$ denotes its complement, we say that $L \in \mathbf{CoNP}$ if and only if $\overline{L} \in \mathbf{NP}$.

(a) Is *COMPOSITES* in **NP**? Justify your answer.

(b) Is *PRIMES* in **CoNP**? Justify your answer.

(c) What do you think of the following proof that $\mathit{PRIMES} \in \mathbf{NP}$?

> Let V be a verifier algorithm demonstrating that $\mathit{COMPOSITES} \in \mathbf{NP}$; e.g., $V(n,w)$ is true if w is a factor of n and $1 < w < n$, otherwise $V(n,w)$ is false.
>
> Let $W(n,w) = \neg V(n,w)$. Then W is a verifier algorithm demonstrating that $\mathit{PRIMES} \in \mathbf{NP}$.

(d) Show that $\mathit{FACTORING} \in \mathbf{NP}$.

(e) Show that $\mathit{FACTORING} \in \mathbf{CoNP}$. (For this part, assume that $\mathit{PRIMES} \in \mathbf{NP}$.)

(f) It is currently unknown whether $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$, or whether $\mathbf{NP} \stackrel{?}{=} \mathbf{CoNP}$. However, most researchers believe it would be surprising if $\mathbf{P} = \mathbf{NP}$, or if $\mathbf{NP} = \mathbf{CoNP}$. Using this, argue that it would be surprising if *FACTORING* is **NP**-complete.

Problem 4. [**Bonus Problem**] (0 points)

Prove that $\mathit{PRIMES} \in \mathbf{NP}$.

HINT: Use the following number-theoretic fact (which you may freely assume, without proof), along with some number theory and group theory.

FACT 1 *Suppose T satisfies the recurrence relation $T(p) \leq \lg p + \sum_q T(q)$, where the sum is over the set of primes q that divide $p-1$, counting multiplicities (e.g., if $q^2 \mid p-1$, then $T(q)$ appears twice in the sum). Then $T(p) = O((\lg p)^2)$.*