6.045J/18.400J: Automata, Computability and Complexity Prof. Ron Rivest

Homework 9

Due: 8 May 2002

Problem 1: Consider the *firehouse* problem: Given a graph G, a distance d, and a limit f on the number of firehouses, can you find f or less nodes in G on which to put firehouses such that every node in G is distance d or less from some firehouse? (The distance, in this context, is the number of edges in the shortest path.) More formally:

$$\mathit{FIREHOUSE} = \left\{ \langle G, d, f \rangle \ : \ \begin{array}{l} \text{there exist } f \text{ nodes in } G \text{ so that every node in } G \\ \text{is distance } d \text{ from one} \end{array} \right\}$$

Show that *FIREHOUSE* is $\mathcal{NP}$-complete.

Problem 2: Consider the language

$$\mathit{HALFCLIQUE} = \left\{ \langle G \rangle \ : \ G \text{ has a } \tfrac{|G|}{2}\text{-clique} \right\}$$

Show that *HALFCLIQUE* is $\mathcal{NP}$-complete.

Problem 3: **(Note: this problem is worth 20 points.)** A k-coloring of an undirected graph $G = (V, E)$ is a function $c : V \to \{1, 2, \ldots k\}$ such that $c(u) \neq c(v)$ for every $(u, v) \in E$. That is, it is a way to color the nodes of G one of k different colors, so that no edge touches two nodes of the same color. Consider the language:

$$\mathit{3\text{-}COLOR} = \{\langle G \rangle \ : \ G \text{ has a 3-coloring}\}$$

In this problem, you will show that this language is $\mathcal{NP}$-complete. (This is also Problem 7.34.) This reduction is tricker than those we've asked you to do before, so we are providing some guidance:

To prove that *3-COLOR* is $\mathcal{NP}$-complete, we use a reduction from *3-SAT*. Given a formula ϕ of m clauses on n variables x_1, x_2, $\ldots x_n$, we construct a graphs $G = (V, E)$ as follows. The set of nodes V consists of a vertex for each variable, a vertex for the negation of each variable, 5 vertices for each clause, and 3 special vertices: *TRUE*, *FALSE*, and *RED*. The edges of the graph are of two types: "literal" edges that are independent of the clauses, and "clause" edges that depend on the clauses. The literal edges form a triangle on the special vertices and also form a triangle on x_1, $\overline{x_i}$, and *RED* for $i = 1, 2, \ldots n$.

1. Argue that in any 3-coloring c of a graph containing the literal edges, exactly one of a variable and its negation colored $c(\mathit{TRUE})$ and the other is colored $c(\mathit{FALSE})$. Argue that for any truth assignment for ϕ, there is a 3-coloring of the graph containing just the literal edges.

The widget shown in Figure 1 is used to force the condition corresponding to a clause $(x \vee y \vee z)$. Each clause requires a unique copy of the 5 vertices that are blank in the figure; they connect as shown to the literals of the clause and the special vertex *TRUE*.

2. Argue that if each of x, y and z is colored $c(\mathit{TRUE})$ or $c(\mathit{FALSE})$, then the widget is 3-colorable iff one of x, y, and z is colored $c(\mathit{TRUE})$.

3. Prove that *3-COLOR* is $\mathcal{NP}$-complete.

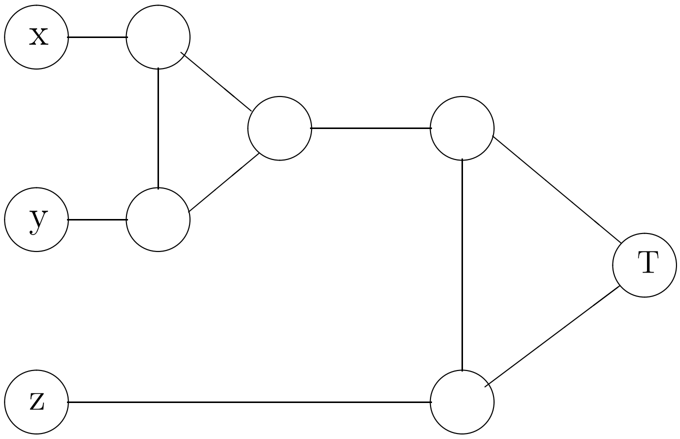

Figure 1: A widget