CS41 Lab 11

November 2018

This week, we've been discussing ways to classify problems according to their difficulty, using the notions of polynomial-time reductions and polynomial-time verifiers. In this lab, you'll develop more sophisticated polynomial-time reductions using **gadgets**.

Below is a synopsis of relevant decision problems for this lab.

- Independent-Set. The input for Independent-Set is a graph $G = (V, E)$ and an integer k. Output yes iff G has an independent set of size at least k.
- Sat. The input for Sat is a set of n boolean variables $x_1, \ldots, x_n$ and m clauses $c_1, \ldots, c_m$, where each clause is the or of one or more literals[1] e.g. $c_i = x_1 \vee \bar{x}_2 \vee x_3 \vee \bar{x}_{17}$. Output yes iff there is a truth assignment to $x_1, \ldots, x_n$ that satisfies every clause.
- 3-Sat. The input for 3-Sat is the same as for Sat, except that each clause is the or of exactly three literals.
- Three-Coloring. The input for Three-Coloring is a graph $G = (V, E)$. Output yes iff the vertices can be colored using three colors such that each edge has different-colored endpoints.

1. Show that 3-Sat $\leq_P$ Sat.
2. Show that 3-Sat $\leq_P$ Independent-Set. **Hint:** Given an input $x_1, \ldots, x_n, c_1, \ldots, c_m$ for 3-Sat, create an input graph $G = (V, E)$ on $3m$ vertices as follows:
   (a) For each clause $c_i = \ell_{i,1} \vee \ell_{i,2} \vee \ell_{i,3}$, create three vertices $v_{i,1}, v_{i,2}, v_{i,3}$. Add edges between each pair of vertices in each clause.
   (b) For each variable x_i, add an edge between any pair of vertices that represent x_i and $\bar{x}_i$. For example, if $c_i = x_1 \vee x_2 \vee x_3$ and $c_j = x_6 \vee \bar{x}_1 \vee \bar{x}_3$, then add edges $(v_{i,1}, v_{j,2})$ and $(v_{i,3}, v_{j,3})$.

   If the 3-Sat instance is a yes instance, what is the size of the largest independent set in G? What if the 3-Sat instance is a no input?
3. In the third exercise, you will reduce 3-Sat $\leq_P$ Three-Coloring. Before getting there, it will be helpful to create some interesting three-colorable graphs. In all of the following exercises, you are to create a three-colorable graph (say the colors are red, blue, green) with certain special properties. The graphs you create should include three vertices marked a, b, c but can (and often will) include other vertices. Except for the properties specified, these vertices should be *unconstrained.* For example, unless the problem states that e.g. a cannot be red, it must be possible to color the graph in such a way that a is red. (You may fix colors for other vertices, just not a, b, c, and not in a way that constrains the colors of a, b, c.)
   (a) Create a graph such that a, b, c all have different colors.

[1] A *literal* is either a boolean variable x_i or its negation $\bar{x}_i$.

(b) Create a graph such that a, b, c all have the same color.

(c) Create a graph such that a, b, c do *NOT* all have the same color.

(d) Create a graph such that none of a, b, c can be green.

(e) Create a graph such that none of a, b, c are green, and they cannot *all* be blue.

4. Show that 3-SAT $\leq_{\mathrm{P}}$ THREE-COLORING. (Hint: Associate the color red with TRUE and the color blue with FALSE.)

5. Show that SAT $\leq_{\mathrm{P}}$ 3-SAT.