Illinois Institute of Technology
Department of Computer Science

# Solutions to Homework Assignment 8

CS 430 Introduction to Algorithms
Spring Semester, 2018

1. Notice that the boolean formula have $\phi$ have more than $n$ variables. As each variable can be either true or false, there are more than $2^n$ possible assignments. Thus the truth table will have more than $2^n$ rows. The reduction takes more than $2^n$ spaces and operations, which is not a polynomial reduction.

2. For each variable and its corresponding negation, *e.g.* $x$ and $\neg x$, create corresponding vertices in graph G. For a clause $x \vee y$, they are equivalent to both $\neg x \to y$ and $\neg y \to x$. For each clause $x \vee y$, we add edge $\neg x \to y$ and $\neg x \to y$ in G. If assignment is true, we will be able to find a path in G with the starting vertex and ending points be negations of each other. For example, if we find $x \to ... \to \neg x$, we have $x \to \neg x$, which is always true. Then we just need to find paths between all pairs of the vertex and its negation. We can just use BFS for each pair, which takes $O(|V| + |E|)$ time. Together, it takes $O|V|(|V| + |E|)$ time.

3. (a) It is easy to show that the gadget itself is 3-colorable as Fig. 1. Furthermore, this gadget has a
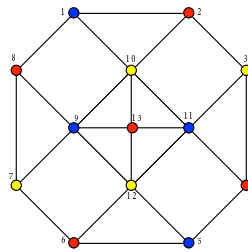


Figure 1: 3-colorable gadget

symmetric property, which is the color of the node 1 could be always the same as the node 5, and the color of node 2 could always be the same as node 6, *i.e.* the opposite corners always have the same color. To prove that the opposite corners are forced to have the same color, we just need to show why node 1 always have the same color as node 5. The rest of corner pairs can be proved similarly.

Given the gadget structure , node 9 and node 11 must have the same color, which is different from the colors of node 10, 12 and 13. Node 10 and 12 must have the same color, which is different from the colors of node 9, 11 and 13. Without loss of generality, let node 13 be colored red, node 9 and 11 be colored blue, node 10 and 12 be colored yellow as shown in the figure. Then the end points of edge $(1, 2)$ must be one blue and one red. Similarly the end points of edge $(3, 4)$ must be one red and one yellow. Then the end points of edge $(5, 6)$ must be one blue and one red. If node 1 and node 5 are not with the same color, then 1 and 6 must be with the same color. Without loss of generality, let them be colored blue. Then node 2 and node 5 must be colored as red. Since the end points of edge $(3, 4)$ are one red and one yellow. A red node will be adjacent to a red node. Thus the assumption node 5 have the same color does not hold.

(b) Given the assignment of colors to the corners such that opposite corners have the same color, we can always find a three coloring similar to Fig. 1. If the corners such that opposite corners have

the same color, we just need to set the colors of the end points of twos neighboring edges. For example, if we set the color on edge (1,2) and edge (2,3) , the colors of node 5, 6 and 7 will be determined. The color of node 4 and node 8 are also determined. Notice that the end points of edge (1,2) and edge (2,3) must not use the same set of colors. Otherwise node 10 and 11 will be forced to have the same color and we have neighboring same color.

(c) We can always replace a crossed edge $(u, v)$ with $u$ embedded at a corner of the gadget and $v$ connected to the opposite corner with an edge, as Fig. 2, which takes time $O(E^2)$. The resulting graph is planar and the symmetric property of the gadget implies that $u$ and $v$ cannot be assigned the same color. We have shown a reduction of the 3-colorability for an arbitrary graph to the same
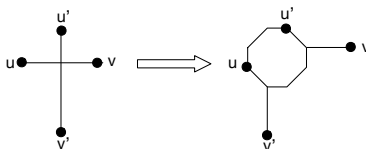
Figure 2: Replacing a cross with the gadget

problem for a planar by using this gadget. Since we can verify a color assignment by checking if the end points of every edge have different colors in $O(E)$ time, 3-coloring a planar graph belongs to the class NP. Because 3-colorable is NP-complete for general graphs, we can prove that 3-colorable is NP-complete for planar graphs.