CS 722 Fall 2016
Homework Assignment #3
Solutions

1. Consider the polynomial-time reduction used to prove 3SAT $\leq_p$ VERTEX-COVER. Let:

    $\phi = (x_1 \vee x_2 \vee \neg x_2) \wedge (x_1 \vee \neg x_1 \vee x_2) \wedge (x_1 \vee \neg x_1 \vee \neg x_2)$

    Here, $\neg x_i$ is the negation of x_i.

    a. Give (G_ϕ, k) constructed from ϕ by the reduction.
    b. Give one satisfying assignment A for ϕ, and a corresponding k-vertex cover for G_ϕ produced by the proof.
    c. Give one k-vertex cover for G_ϕ that is distinct from the one you gave in (b), and give a corresponding assignment for ϕ produced by the proof.

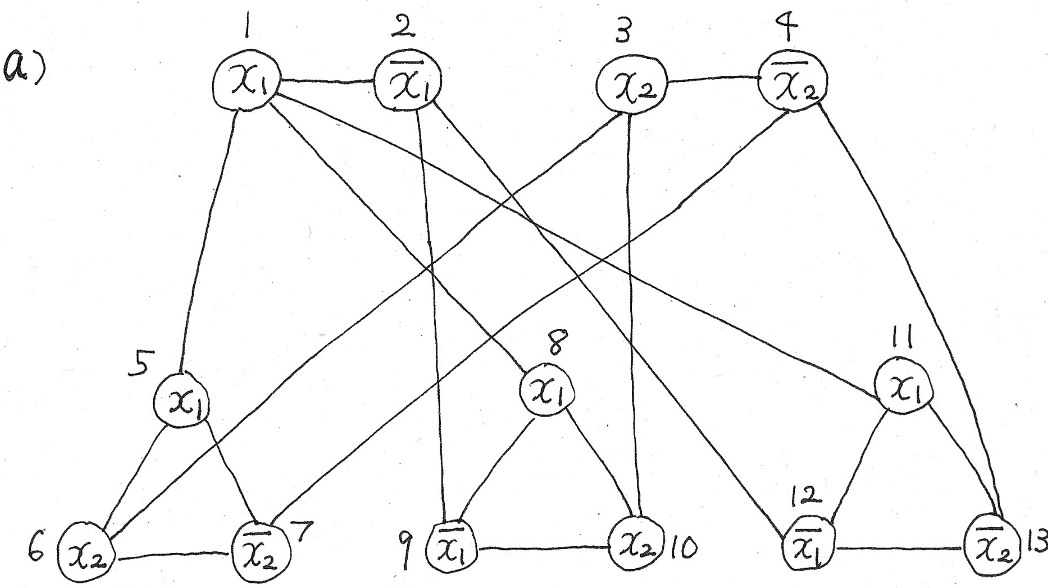
2. Consider the polynomial-time reduction used to prove 3SAT $\leq_p$ SUBSET-SUM, and consider the formula ϕ in Question (1).

    a. Give the "sum table" constructed from ϕ by the reduction.
    b. Give one satisfying assignment A for ϕ, and a corresponding subset S' produced by the proof.
    c. Give one subset that adds to t and is distinct from the one you gave in (b), and give a corresponding assignment for ϕ produced by the proof.

    There are different correct choices of assignments, vertex covers, and subsets – the following are examples.

1

a)



$$k = m + 2n = 2 + 2\cdot 3 = 8$$

b) $A = \{x_1 = 0,\ x_2 = 1\}$ $VC = \{2, 3, 5, 7, 8, 10, 11, 13\}$

c) $VC = \{1, 4, 6, 7, 9, 10, 11, 12\}$ $A = \{x_1 = 1,\ x_2 = 0\}$

2

a)

	1	2	C_1	C_2	C_3
X_1	1	0	1	1	1
$\overline{X_1}$	1	0	0	1	1
X_2		1	1	1	0
$\overline{X_2}$		1	1	0	1
g_1			1	0	0
h_1			1	0	0
g_2				1	0
h_2				1	0
g_3					1
h_3					1
t	1	1	3	3	3

b) $A = \{x_1 = 0,\ x_2 = 1\}$ $S' = \{\overline{X_1}, X_2, g_1, h_1, g_2, g_3, h_3\}$

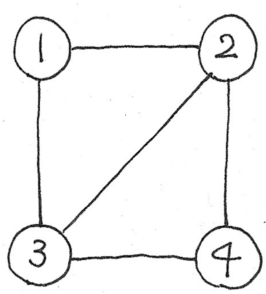
c) $S' = \{X_1, \overline{X_2}, g_1, g_2, h_2, g_3\}$ $A = \{x_1 = 1,\ x_2 = 0\}$

3. Consider the UHAMPATH problem treated on page 319 of the textbook (page 291 of the 2nd edition) and the LPATH problem in Problem 7.21 on page 324 (Problem 7.20 on page 296 of the 2nd edition). Find and describe a polynomial-time reduction from UHAMPATH to LPATH. Justify your reduction by showing that its worst-case runtime is bounded by a polynomial of the input size of UHAMPATH and the equivalence condition holds.

Given an instance <G, s, t> of UHAMPATH, the reduction computes n = the number of nodes in G and produces the instance <G, s, t, n> of LPATH. This reduction runs in polynomial time since computing the number of nodes can be done in polynomial time.

Proof of the equivalence condition: Suppose G has a Hamiltonian path p from s to t. Since p visits each node of G exactly once, it is a simple path of length n. Suppose p is a simple path of length at least n from s to t. If the length of p is greater than n, it cannot be simple as it must repeat at least one node. So the length of p is n. Since p is simple, the n nodes in p must be distinct, implying that p visits each of the n nodes exactly once. Hence p is a Hamiltonian path from s to t.

4. Study the reduction from the 4-colorability problem to SAT described in the "Boolean Satisfiability" article handed out on 11/08/16 (toward the end of page 77 and in a beginning part of page 78). Specifically, it describes a reduction function f: G → ϕ from an undirected graph G to a Boolean formula ϕ s.t. G is 4-colorable iff ϕ is satisfiable. Give the Boolean formula generated from the following graph by the reduction:



Also give a coloring of G and the corresponding satisfying assignment.

$\neg[((c_{10} \wedge c_{20}) \vee (\neg c_{10} \wedge \neg c_{20})) \wedge ((c_{11} \wedge c_{21}) \vee (\neg c_{11} \wedge \neg c_{21}))] \wedge$
$\neg[((c_{10} \wedge c_{30}) \vee (\neg c_{10} \wedge \neg c_{30})) \wedge ((c_{11} \wedge c_{31}) \vee (\neg c_{11} \wedge \neg c_{31}))] \wedge$
$\neg[((c_{20} \wedge c_{30}) \vee (\neg c_{20} \wedge \neg c_{30})) \wedge ((c_{21} \wedge c_{31}) \vee (\neg c_{21} \wedge \neg c_{31}))] \wedge$
$\neg[((c_{20} \wedge c_{40}) \vee (\neg c_{20} \wedge \neg c_{40})) \wedge ((c_{21} \wedge c_{41}) \vee (\neg c_{21} \wedge \neg c_{41}))] \wedge$
$\neg[((c_{30} \wedge c_{40}) \vee (\neg c_{30} \wedge \neg c_{40})) \wedge ((c_{31} \wedge c_{41}) \vee (\neg c_{31} \wedge \neg c_{41}))]$

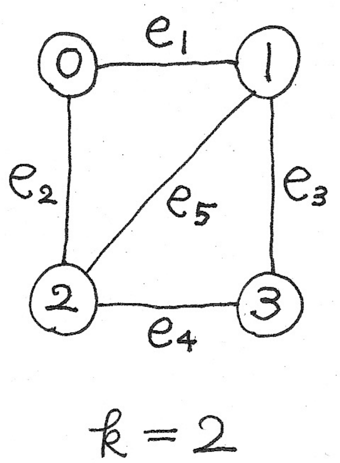
An example coloring is:

node 1 and 4 colored by 0 = 00
node 2 colored by 1 = 01
node 3 colored by 2 = 10

The corresponding satisfying assignment is:

$c_{10} = 0 \wedge c_{11} = 0 \wedge c_{20} = 0 \wedge c_{21} = 1 \wedge c_{30} = 1 \wedge c_{31} = 0 \wedge c_{40} = 0 \wedge c_{41} = 0$

5. Find and describe a direct polynomial-time reduction from VERTEX-COVER to SAT (not the reduction implied by Cook-Levin Theorem 7). Justify your reduction by showing that its worst-case runtime is bounded by a polynomial of the input size of VERTEX-COVER and the equivalence condition holds. Also, show the Boolean formula generated from the following graph and k = 2 by your reduction:



There would be several good reductions. The following is an example reduction.

We begin by a description of the reduction of the above example graph, G, and then generalize. First, label the four nodes by 0, 1, 2, 3 and the five edges by e_i, $1 \le i \le 5$, as shown. The existence of a 2-vertex cover is encoded by two node variables n_0, n_1. Generate the conjunction of five formulas C_i, $1 \le i \le 5$, each encoding "$e_i = (u_i, v_i)$ is incident to n_0 or n_1", namely, C_i is $(n_0{=}u_i \vee n_0{=}v_i \vee n_1{=}u_i \vee n_1{=}v_i)$. Thus the conjunction is:

$$(n_0{=}0 \vee n_0{=}1 \vee n_1{=}0 \vee n_1{=}1) \wedge$$
$$(n_0{=}0 \vee n_0{=}2 \vee n_1{=}0 \vee n_1{=}2) \wedge$$
$$(n_0{=}1 \vee n_0{=}3 \vee n_1{=}1 \vee n_1{=}3) \wedge$$
$$(n_0{=}2 \vee n_0{=}3 \vee n_1{=}2 \vee n_1{=}3) \wedge$$
$$(n_0{=}1 \vee n_0{=}2 \vee n_1{=}1 \vee n_1{=}2)$$

This has solutions $n_0{=}1, n_1{=}2$ and $n_0{=}2, n_1{=}1$. The variables n_0 and n_1 are encoded by 2-bit variables $n_{00}n_{01}$ and $n_{10}n_{11}$, respectively. The node numbers 0, 1, 2, 3 are encoded by 00, 01, 10, 11. Each equality comparison is translated into two 1-bit equality comparisons, then to a conjunction of two literals. For example, $n_1{=}1$ is translated into $n_{10}{=}0 \wedge n_{11}{=}1$, which is then translated into $\neg n_{10} \wedge n_{11}$.

Now the generalization. Let n be the # of nodes of the input graph G. If $k > n$, G cannot have a k-vertex cover, so the reduction immediately generates a trivial unsatisfiable Boolean formula like $x \wedge \neg x$. If $k \le n$, label n nodes by $0, \ldots, n-1$ and the m edges by e_i, $1 \le i \le m$. The existence of a k-vertex cover is encoded by k node variables $n_0, \ldots, n_{k-1}$. Generate the conjunction $\phi = \bigwedge_{1 \le i \le m} C_i$, where each C_i encodes "$e_i = (u_i, v_i)$ is incident to one of $n_0, \ldots, n_{k-1}$". Namely, C_i is

$$(n_0{=}u_i \vee n_0{=}v_i \vee \cdots \vee n_{k-1}{=}u_i \vee n_{k-1}{=}v_i)$$

Each node variable n_i is encoded by p-bit variables $n_{i0} \cdots n_{i(p-1)}$ where $p = \lceil \log_2 n \rceil$. The node numbers $0, \ldots, n-1$ are encoded by p-bit binary numbers. Each equality comparison is translated into a conjunction of p literals as described in the above example. Thus the generated formula has $m \cdot 2k \cdot p$ literals. Since $k \le n$, $m \cdot 2k \cdot p \le m \cdot 2n \cdot \lceil \log_2 n \rceil$, which is polynomially bounded by m, n.

Proof of the equivalence condition. Suppose ϕ is satisfied by an assignment A. Then A makes each C_i true and determines a solution $n_0{=}a_0, \ldots, n_{k-1}{=}a_{k-1}$, meaning that each edge $e_i = (u_i, v_i)$ is incident to one of $a_0, \ldots, a_{k-1}$. Since some of the a_i may be identical, $\{a_0, \ldots, a_{k-1}\}$ is a vertex cover containing at most k nodes. If it has less than k nodes, add redundant nodes so that it is a vertex cover of exactly k nodes. Thus G has a k-vertex cover. Suppose that G has a k-vertex cover $\{a_0, \ldots, a_{k-1}\}$. Then each edge $e_i = (u_i, v_i)$ is incident to one of $a_0, \ldots, a_{k-1}$. So $n_0{=}a_0, \ldots, n_{k-1}{=}a_{k-1}$ is a solution to each C_i. Thus this solution determines an assignment that makes each C_i true, and satisfies ϕ.