CSC 364H (L0101) **Test 3 — solutions** **Summer 99**

1. (a) A graph G is 2-colourable iff all its components consist of one of the following: a single node, or a cycle with and even number of nodes, or a simple path.

   Thus, the following algorithm will do the job: First check that for all $v \in V$, $\deg(V) \leq 2$. If not reject. Otherwise, pick any node v. If $\deg(v) = 0$, v is an isolated node. If $1 \leq \deg(v) \leq 2$, then check that v is part of an even cycle, or that v is part of a simple path, by following one of the edges (perhaps the only edge) attached to v.

   This algorithm is polytime — in fact it's running time is $O(|V|^2)$.

   (b) Suppose that 3-COLOUR $\leq_p$ 2-COLOUR. Then, since by (a) 2-COLOUR is in P, it follows that 3-COLOUR is in P. Since 3-COLOUR is NP-complete, it follows that $P = NP$, which contradicts the assumption.

2. (a) Verify that B is even, and that $|V| \geq B$. Guess-and-check algorithm: *guess* a subset $V' \subseteq V$, and *check* that $|V'| = B$, and that V' is an independent set in G.

   (b) Given an input $I = \langle (V, E), B \rangle$ to IS, we transform it to an instance $I' = \langle (V', E'), B' \rangle$ to EIS, where

$$\begin{cases} V' = V \text{ and } E' = E \text{ and } B' = B & \text{if } B \text{ is even} \\ V' = V \cup \{v_0\} \text{ and } E' = E \text{ and } B' = B + 1 & \text{if } B \text{ is odd} \end{cases}$$

   where v_0 is a new vertex not in V. The correctness condition is: $I \in \text{IS}$ iff $I' \in \text{EIS}$.

3. (a) True: Since $L_1 \leq_m L_2$, there exists a computable function f such that $x \in L_1$ iff $f(x) \in L_2$. Since L_2 is semi-decidable, there is a TM M such that $L_2 = L(M)$. We design M' so that $L_1 = L(M')$ as follows:

   > On input x, M' computes $f(x)$, and then simulates M on $f(x)$, and accepts only if M accepts.

   (b) False: Let $L_2 = \Sigma^*$. Then, trivially, $L_2 \in P$. But $A_{\text{TM}} \subseteq \Sigma^* = L_2$, and A_{TM} is not even decidable, and $P \subseteq$ decidable.

   (c) True: Since L and $\overline{L}$ are semi-decidable, there exist TMs M_1 and M_2 such that $L = L(M_1)$ and $\overline{L} = L(M_2)$. We design M_3 so that $L = L(M_3)$ and M_3 halts on all inputs:

   > On input x, M_3 simulates M_1 and M_2 "in parallel" on input x. If M_1 accepts, then M_3 accepts, if M_2 accepts, then M_3 rejects.

   Clearly $L = L(M_3)$. Since $L \cup \overline{L} = \Sigma^*$, M_1 or M_2 always accepts, so M_3 always halts.