Solution sketch 6 - Computational Models

1. Partial solutions. Only ideas are given. Your solutions should include correctness proofs and explanations. For any NPC problem, you should also explain why is it in NP, and show that the reduction is polynomial.

   (a) NPC. Reduction from IS. Given input $G = (V, E), k$ for IS, we construct a set A_v for every $v \in V$ and let $A_v = \{\{u, v\} \in E\}$. We keep the same k.

   (b) NPC. Reduction from $Clique$. Exactly the same reduction as in 1.

   (c) P. We check all possible sets of 10 variables.

   (d) NPC. Reduction from 3SAT. Given a formula ψ input for 3SAT, we construct a formula ψ'. ψ' contains ψ and another copy of ψ in which every variable x is replaced by $\neg y$ and $\neg x$ is replaced by y, where y is a new variable.

   (e) NPC. Reduction from HAMPATH. Given input $G = (V, E)$ for HAMPATH, we construct the input $G, |V| - 1$.

2. (a) Correct. For every nontrivial L_1 and L_2 in p, $L_1 \leq_p L_2$ (prove it).

   (b) Equivalent to P vs. NP. Note that $NP \subseteq R$, so every two nontrivial languages are mapping-reducable to each-other. Thus if $P = NP$ than the claim is true by the observation in (a), and if $P \neq NP$ then the claim is false, say by taking $L_1 = SAT$ and $L2 = \{0\}$.

   (c) Correct. The reduction from any language in RE to A_{TM} is in fact linear in time (see slide 36 of Lecture 10).

   (d) Equivalent to P vs. NP. If $P = NP$ then every language in P is $NP - complete$ (prove it). If $P \neq NP$ then, say, $SAT \notin P$. Assume towards contradiction that $SAT \leq_p L$, so $L \notin P$ and this is obviously not the case.

   (e) Correct. Let $L \in NP$. Then, there exists a polynomial-time reduction f such that $x \in L iff f(x) \in SAT$. Consider the $TM M_L$, that on input x computes $f(x)$, simulates the TM for SAT and answers accordingly.

The correctness follows immediately from the correctness of the reduction. Let $n = |x|$. Assume that f runs in time n^c for some constant c. The length of $f(x)$ is then at most n^c, so the running time of the TM for SAT is at most $(n^c)^{O(logn^c)} = n^{O(logn)}$, so the overall running time of ML is $n^{O(logn)}$ as well.

3. (a) Let M be the nondeterministic TM that decides L in polynomial time. M^* on $x = x_1...x_n$:

   i. Guess a partition of x to $x = w_1...w_k$ (how?).
   ii. For every $iin\{1, ..., k\}$, simulate $M(w_i)$.
   iii. If all simulations accepted, accept. Otherwise, reject.

   The guessing can be done in linear time, and we simulate M at most n times, so M^* runs in polynomial time. Now, if $xinL^*$ there exists a partition $x = w_1...w_k$ exists such that $w_i \in L$ for every i, so there exist computation paths (for every such i) through which M accepts. Thus, there is an accepting computation path for M^*. If $x \notin L^*$ there exists no such partition, and every computation path of M^* rejects. Hence, $L^* \in NP$.

   (b) Let M be the TM that decides L in polynomial time. M^* on $x = x_1...x_n$:

   i. Construct a directed graph $G = (V, E)$ where $V = \{1, ..., n + 1\}$ and for every $i < j, (i, j)E$ if and only if $M(x_{i,...,j-1}) = 1$.
   ii. Check if there exists a path in G from 1 to $n + 1$.
   iii. If such path exists, accept. Otherwise, reject.

   There are $O(n^2)$ possible edges in G, and for every possible edge we simulate M. Then, we run a reachability algorithm (say, BFS). Thus, M^* runs in polynomial time. If a partition $x = w_1...w_k$ exists such that $w_i \in L$ for every i then there is a path from 1 to $n+1$ in G. Otherwise, there is no path. Overall, M^* decides L^* in polynomial time and hence $L^* \in P$.

4. (a) The claim is true. As B is nontrivial there exist $y \in B$ and $z \notin B$. Set $n_0 = max\{|y|, |z|\} + 1$ and $f(x)$ to be y if $x \in A$ and z otherwise. As $A \in P$, f can be computed in polynomial time, and the correctness easily follows. Also, as $|f(x)| < n_0$ for every x satisfying $|x| \geq n_0$, the reduction is also shrinking.

   (b) The claim is false. Assume to the contrary that there is a shrinking reduction f from SAT to SAT with a constant n_0. Denote $SAT_{n_0} = \{< \varphi >\in SAT|| < \varphi > | \leq n_0\}.SAT_{n_0} \in P$ as it is finite. Deciding

SAT in P will be as follows: Given φ, compute the series φ_k such that $\varphi_0 = \varphi$ and $\varphi_k = f(\varphi_{k-1})$. The computation stops when we reach a k' such that $| < \varphi_{k'} > | \leq n_0$ and we answer according to whether $\varphi_{k'} \in SAT_{n_0}$ or not. As the reductions preserve correctness, it is easy to see that $\varphi_{k'} \in SAT_{n_0}$ iff $\varphi \in SAT$. Also, the above procedure can be done in polynomial time, as f is polynomial and we apply it linear number of times. Therefore, $SAT \in P$, in contradiction to $P \neq NP$.

5. Following the guidelines given in the slides, we will first prove that M always halts on w', and then prove that it does so in the same state that M does on w.

   M always halts on w'. Assume that it doesn't. Then, at least, one of the following occurs:

   (a) $M(w')$ crosses infinitely many times the border $(|x+y|, |x+y|+1)$: i.e., it moves infinitely many times back and forth between the cell $|x+y|$ and the cell $|x+y|+1$.

   (b) $M(w')$ spends infinitely many steps in the cells 1 to $|x+y|$, without crossing the border of $(|x+y|, |x+y|+1)$.

   (c) $M(w')$ spends infinitely many steps right to the cell $|x+y|$, without crossing the border of $(|x+y|, |x+y|+1)$.

   Since, $M(w)$ halts in a finite time, is not hard to show (using claim 2) that *none* of the above options can occur.

   $M(w')$ halts in the same state as $M(w)$. Let t be the step in which $M(w')$ halts (by the first step, such t exist, and let i be the head location when it happens. By claim 2, there exists t' such that $M(w)$ is in the same state after t' steps (to prove that, distinguish between the case that $i \leq |x+y|$ and $i > |x+y|$). Namely, $M(w)$ halts in the same state that $M(w')$ does.

6. We will use a two taped, non-deterministic TM that allows the head to stay put as well as moving right and left.

   - $\delta(q_0, (\#, \sqcup)) = \{(q_{i_1}, (\#, \sqcup), (R, S))\}$
     start writing 1^i
   - $\delta(q_{i_1}, (1, \sqcup)) = \{(q_{i_2}, (1, 1), (R, R))\}$
     ensure that $i > 0$

- $\delta(q_{i_2}, (1, \sqcup)) = \{(q_{i_2}, (1, 1), (R, R)), (q_{i_2}, (1, \sqcup), (R, S))\}$
  non-deterministically choose if to write 1 or not on the second tape
- $\delta(q_{i_2}, (\#, \sqcup)) = \{(q_L, (\#, \#), (L, R))\}$
  mark end of 1^i
- $\delta(q_L, (1, \sqcup)) = \{(q_L, (1, \sqcup), (L, S))\}$
  go left
- $\delta(q_L, (\#, \sqcup)) = \{(q_{j_1}, (\#, \sqcup), (R, S))\}$
  start writing q^j
- $\delta(q_{j_1}, (1, \sqcup)) = \{(q_{j_2}, (1, 1), (R, R))\}$
  ensure that $j > 0$
- $\delta(q_{j_2}, (1, \sqcup)) = \{(q_{j_2}, (1, 1), (R, R)), (q_{j_2}, (1, \sqcup), (R, S))\}$
  non-deterministically choose if to write 1 or not on the second tape
- $\delta(q_j, (\sqcup, \sqcup)) = \{(q_a, (\sqcup, \sqcup), (S, S))\}$
  terminate