Dept. of Computer Science | **Assignment # 2** | **CSC 463H1**
University of Toronto | **(Due Feb 12, 3:00 pm)** | **Winter 2021**

Worth: 15%

1. [20 marks]
   Let A be an arbitrary language.

   (a) Show that A is semi-decidable if and only if $A \leq_m A_{TM}$. (This, combined with the fact that A_{TM} is semi-decidable, shows that A_{TM} is **complete** for the class of semi-decidable problems.)

   (b) Show that if A is semi-decidable and $A \leq_m A^c$, then A is decidable.

   Solution.

   (a) $(\longleftarrow)$
   Since $A \leq_m A_{TM}$ and A_{TM} is semi-decidable, by property of reduction it follows that A is semi-decidable.
   $(\longrightarrow)$
   If A is semi-decidable, by definition there is a TM M such that $w \in A \iff M$ accepts w. Define

$$\begin{array}{rcl} f : \Sigma^* & \rightarrow & \Sigma^* \\ w & \mapsto & \langle M, w \rangle \end{array}$$

   f is clearly computable. Moreover

$$w \in A \iff M \text{ accepts } w \iff \langle M, w \rangle \in A_{TM}.$$

   Hence, $A \leq_m A_{TM}$.

   (b) Let f be a computable function witnessing $A \leq_m A^c$, i.e., $x \in A \iff f(x) \in A^c$. By contraposition, $x \notin A \iff f(x) \notin A^c$, i.e., $x \in A^c \iff f(x) \in A$, which proves $A^c \leq_m A$ (witnessed by the same computable function f).
   Since A is semi-decidable and $A^c \leq_m A$, it follows by property of reduction that A^c is semi-decidable, i.e., $A \in coSD$.

   Finally, since $SD \cap coSD = D$, it follows that $A \in D$.

2. [20 marks]
   Let A be an arbitrary language.

   (a) Show that there exists a language B such that $B \not\leq_m A$.

   (b) Show that there exists a language C such that $A \leq_m C$ and $C \not\leq_m A$.

   Solution.

   (a) Consider the set $S = \{B \mid B \leq_m A\}$. For each $B \in S$, there is a computable function $f : \Sigma^* \rightarrow \Sigma^*$ that witnesses the reduction $B \leq_m A$. Moreover, there is a 1-1 correspondence between B and f, i.e., if there are two different sets B and C such that $B \leq_m A$ and $C \leq_m A$, then both the reductions cannot be witnessed by the same function f. Otherwise, for any $w \in B \setminus C$ (assuming without loss of generality that $B \setminus C \neq \emptyset$), it follows that $f(w) \in A$ (since $w \in B$) and also $f(w) \notin A$ (since $w \notin C$), a contradiction.
   We know that the number of Turing machines is countable. In particular, the number of computable functions is countable. Because of the 1-1 correspondence, it follows that S is countable. But the set of languages over Σ is uncountable. Thus, there exists a language B such that B is not in S, i.e., $B \not\leq_m A$.

(b) Pick two distinct characters $a, b \in \Sigma$, and define $C = \{ax \mid x \in A\} \cup \{by \mid y \in B\}$.
(If Σ has only one character a, define $C = \{xx \mid x \in A\} \cup \{ayy \mid y \in B\}$.)
In either case, we have $A \leq_m C$ and $B \leq_m C$.
But since $B \not\leq_m A$ (by part (a)), it follows that $C \not\leq_m A$ (since otherwise $B \leq_m A$ by transitivity of $\leq_m$).

3. [20 marks]
Show that every infinite semi-decidable language has an infinite decidable subset.

Solution. Since A is semi-decidable, there is an enumerator E that prints the strings of A in some order, say $\{x_0, x_1, x_2, \ldots\}$. Inductively define a subset $B \subseteq A$ by declaring that $x_0 \in B$, and for $j > 0$, $x_j \in B$ if and only if its length is greater than x_i for all $i < j$. By construction, $B \subseteq A$ and is infinite, since A must contain strings of arbitrarily large length if A was infinite.

To show that B is decidable, we modify the enumerator E to construct an enumerator E' that prints B in lexicographic order. E' simulates E and prints the first string x_0 that E prints. After that point, every time E prints a string x_j, E' compares its length to the last string x_i that it itself printed, and prints x_j if and only if x_j is longer than x_i. Clearly, E' enumerates B lexicographically, and hence B is decidable.

4. [20 marks]

A Turing machine M has a useless state if there is some state q that is never entered on M's computation beginning on any string w. Let $U = \{\langle M \rangle \mid M \text{ is a Turing machine with a useless state}\}$. Show that $U \in coSD \setminus D$.

Solution. Assume, for a contradiction, that U has a decider S. Construct a decider N for A_{TM} as follows:

On input $\langle M, w \rangle$:

- Construct a TM R on the input alphabet $\{0, 1, 2\}$ as follows: on input 0, R goes through all non-halting states of M and then enters q_{accept}; on input 1, R enters q_{reject}; and on input 2, R simply simulates M on w, and if that simulation enters M's accept state, R enters a new state q_A and accepts.
- Run S on input $\langle R \rangle$. If it accepts, *reject*; if it rejects, *accept*!

Note that the only possible useless state of R is the state q_A because R uses all the other states on inputs 0 and 1. And the only way for R to use the state q_A is if M accepts w. Thus, R does not have a useless state if and only if M accepts w. On the other hand, N accepts $\langle M, w \rangle$ if and only if S rejects $\langle R \rangle$, i.e., R does not have a useless state. Combining the two, we obtain that N accepts $\langle M, w \rangle$ if and only if M accepts w. In other words, N decides A_{TM}.

To show that $U \in coSD$, we show that $U^c \in SD$ by designing a TM N that recognizes U^c as follows:

On input $\langle M \rangle$:

- Make a list of all the states of M
- Repeat the following for $i = 1, 2, \ldots$:
  - Generate string s_i lexicographically
  - Append it to the existing strings with a delimiter
  - Execute M's next move on the inputs $s_1, \ldots, s_i$
  - After each computation, grab the state M moved to, and add it to a *seen* list.
  - The moment all states of M are seen, halt and accept.

If M does not have a useless state, if will take finitely many strings to go through all the states of M, and hence N, on input $\langle M \rangle$, will halt after finitely many steps. But if M has a useless state, N will never halt on input $\langle M \rangle$. In other words, N recognizes U^c.