Solution sketch 5 - Computational Models - Spring 2012

As always, these solutions might contain mistakes. Please let us know if you found any!
(Questions graded: Q1 Q2 b d f h j k Q3 Q4 Q5)

1. (a) Define $\mathcal{C} = \{L(M) \mid \langle M\rangle \in L\}$. Since L is not trivial, $\emptyset \neq \mathcal{C} \subset \mathcal{RE}$. By Theorem 4 in lecture 9, $L_{\mathcal{C}} = \{\langle M\rangle \mid L(M) \in \mathcal{C}\}$ is undecidable. Since L is semantic, $L_{\mathcal{C}} = L$ (you have to formally prove this statement, and recognize exactly where the fact that L is semantic is needed).

   (b) No. $\{\langle M\rangle \mid$the number of states of M is even$\}$ satisfies (i) and (ii) and of course decidable by looking at the TM's description.

2. (a) $\mathcal{RE} \setminus \mathcal{R}$. $\mathcal{RE}$: Let $x_1, x_2, ...$ be a lexical ordering of all strings. Use a universal TM U. for $i = 1...\infty$ run M on the inputs $x_1, ..., x_i$ for i steps. accept if M halts on any input (prove correctness). $\notin \mathcal{R}$: $H_{TM} \leq_m L$. Given $\langle M, w\rangle$, we can compute a TM $\langle M'\rangle$ in the following way. M' will ignore its input x and simply run M on w and return $true$. This reduction is computable. If M halts on w then M' halts on all inputs as required. If M does not halt on w then M' does not halt on any input, as required.

   (b) $\mathcal{R}$. Either the TM that always answers $true$, or the TM that always answers $false$ decides this language.

   (c) None of the above. $\notin \mathcal{RE}$: $\overline{H_\epsilon} \leq_m L$. Given the TM $\langle M\rangle$, we can compute $\langle M', 0, 1\rangle$. M' does the following for an input x. if $x = 0$ it runs M on ϵ. otherwise, it returns $false$. The reduction is computable. If M halts on ϵ then M' will halt on both 0 and 1, as required. If M does not halt on ϵ then M' will only halt on 1. $\notin co-\mathcal{RE}$: $H_\epsilon \leq_m L$. Given the TM $\langle M\rangle$, we can compute $\langle M', 0, 1\rangle$. M' does the following for an input x. if $x = 0$ it runs M on ϵ. otherwise, it starts an infinite loop. The reduction is computable. If M halts on ϵ then M' halts only on 0, as required. If M does not halt on ϵ then M' will not halt on 0 or 1.

   (d) $\mathcal{RE} \setminus \mathcal{R}$. $\in \mathcal{RE}$: We will use a universal TM U to run M on all inputs as in section (a). If we encounter 3 different inputs that M accepts, we will return $true$ (prove correctness).$\notin \mathcal{R}$: Rice's theroem.

   (e) $\mathcal{R}$. This is trivial. It's the language of all TMs since for any M, $L(M) \in RE$.

   (f) $\mathcal{R}$. This is a finite language. All finite languages are regular and of course decidable.

   (g) $co-\mathcal{RE} \setminus \mathcal{R}$. $\in co-\mathcal{RE}$: Run in parallel on all inputs, and answer yes if we reach the $|x| + 7$ position on the tape, otherwise we are in a loop. $\notin \mathcal{R}$: We can show a mapping reduction $\overline{A_{TM}} \leq_m L$. Given $\langle M, w\rangle$ we construct $\langle M'\rangle$. M' checks if its input x is an accepting computational history of the run of M on w. If it is, we keep going right on the tape forever. Otherwise, halt. We saw that we don't need any space to the right of the description of x to check if it's an accepting computational history and therefore, $M' \in L$ iff M accepts w.

   (h) $\mathcal{R}$. Given M and x, we run M on x for $|x| + |Q| + 1$ steps (where Q is the set of states of M), and accept iff M halts. To prove correctness, notice that after $|x|$ steps, the head will only see blanks. If the run does not stop in $|Q| + 1$ more steps, it means that it entered one of its states at least twice (while the head sees blanks, and there for the run will never stop.

   (i) $co-\mathcal{RE} \setminus \mathcal{R}$. $\in co-\mathcal{RE}$: Use a TM that goes over all strings alphabetically. for each string check if it's in $L(A_1)$ but not in $L(A_2)$ or vice versa. If such a string exists, answer yes, otherwise, we are in a loop. $\notin \mathcal{R}$: We know that E_{LBA} is undecidable. We give a mapping reduction from E_{LBA}. Given A we return $A_1 = A$ and $A_2 = A'$, where $L(A') = \emptyset$. $L(A) = \emptyset$ iff $L(A_1) = L(A_2)$.

   (j) $\mathcal{R}$. The algorithm was given in Lecture 3.

   (k) $co-\mathcal{RE} \setminus \mathcal{R}$. $\in co-\mathcal{RE}$: Use a TM that goes over all strings alphabetically. for each string check if it's in $L(G_1)$ but not in $L(G_2)$ or vice versa. If such a string exists, answer yes, otherwise, we are in a loop. $\notin \mathcal{R}$: We know that $A = \{G \mid L(G) = \Sigma^*\}$ is undecidable. We give a mapping reduction from A. Given G we return $G_1 = G$ and $G_2 = G'$, where $L(G') = \Sigma^*$. $L(G) = \Sigma^*$ iff $L(G_1) = L(G_2)$.

3. (a) Yes. Let $L_1 = \{\langle M, x\rangle | M$ is a TM that halts on x or has an even number of states$\}$. $L_2 = \{\langle M, x\rangle | M$ is a TM that halts on x or has an odd number of states$\}$ It's easy to show that L_1,$L_2 \in \mathcal{RE} \setminus \mathcal{R}$ but $L_1 \cup L_2 = \Sigma^*$.

   (b) No. Since then we could construct a TM that decides L_1. Let M_1 and M_2 be the TMs that accept L_1 and L_2 respectively. Let M_u and M_i be the TMs that decide $L_1 \cup L_2$ and $L_1 \cap L_2$ respectively. We shall construct the TM M that decides L_1. M works in the following manner on an input x: If $M_i(x) = true$, return $true$. If $M_u(x) = false$, return $false$. Otherwise, we run M_1 and M_2 in parallel on x. if M_1 accepts, return $true$. If M_2 accepts, return $false$. Correctness sketch: M_i and M_u are guaranteed to halt. if x is in the intersection we can return $true$. If x is not in the union, we can return $false$. Otherwise, it's either in $L_1 \setminus L_2$ or in $L_2 \setminus L_1$. Therefore, it's guaranteed that M_1 or M_2 will return $true$ and we will halt with the correct answer.

4. No. Since $\langle M_{loopy}\rangle \in L$, as in the proof of Rice's theorem we have that $\overline{A_{TM}} \leq_m L$, so $L \notin \mathcal{RE}$ (you had to fully show this).

5. Non of the above. Let M_A be a TM that accepts A.

   To show that $B \notin \mathcal{RE}$, we show that $\overline{H_{TM}} \leq_m B$. Given $\langle M, x\rangle$ we construct $\langle M'\rangle$. M' gets an input y, and simulates a run of M on x for $|y|$ steps. If M does not halt in this time, M' simulates a run of M_A on y and return the same answer. If M does halt in this time, M' rejects.

   To show that $B \notin co-\mathcal{RE}$, we show that $H_{TM} \leq_m B$. Given $\langle M, x\rangle$ we construct $\langle M'\rangle$. M' gets an input y, and simulates a run of M on x. Then M halts, M' simulates a run of M_A on y and return the same answer.

   Prove correctness of both reductions.