COMS 331: Theory of Computing, Spring 2023
Homework Assignment 7

Neha Maddali

Due at 10:00PM, Wednesday, April 5, on Gradescope.

Problem 44.
Let A $\sqcup$ B = $\{0x \mid x \in A\} \cup \{1y \mid y \in B\}$, and we will show that A $\leq_m$ A $\sqcup$ B. By the definition of $\leq_m$, $x \in A \implies f(x) \in A \sqcup B$. So, f(x)=0x makes that statement true, and if $x \notin A$, then $f(x) \notin A \sqcup B$. Therefore, $A \leq_m$ A $\sqcup$ B. ■

Problem 45.
Let A,B,C $\subseteq \{0,1\}^*$. It is true that A $\leq_m$ C and B $\leq_m$ C. By the definition of $\leq_m$, $x \in A \implies f(x) \in C$, and $y \in B \implies g(y) \in C$. So, if A $\sqcup$ B $\leq_m$ C is true, then $z \in$ A $\sqcup$ B implies that $h(z) \in C$. By the definition of join, $z \in 0x$ or $z \in 1y$ or ($z \in 0x$ and $z \in 1y$). So, if $z \in 0x$, then $h(z) = f(x)$. If $z \in 1y$, then h(z) = g(y). Since f(x), g(y) $\in$ C, it follows that $h(z) \in C$. Therefore, if $A \leq_m$ C and B$\leq_m$ C, then A $\sqcup$ B $\leq_m$ C. ■

Problem 46.
Suppose that A is $\leq_m$-complete for CE. Then there exists a computable function f such that for every x, $x \in A$ if and only if $f(x) \in CE$. Use f to define the reduction from H to $A \sqcup A^c$. Let M be a TM and w be a string. We want to decide whether M halts on input w. Define a TM M' as follows: on any input x, M' first computes f(x), then simulates M on input w for $|f(x)|$ steps. If M has halted on w within $|f(x)|$ steps, then M' accepts, otherwise M' loops forever. Now show that M halts on w if and only if there exists x such that $f(x) \in CE$ and M' halts on x. If M halts on w, then there exists a natural number n such that M halts on w within n steps. Let x be a string such that $f(x) = 1^n$. Then M' simulates M on w for n steps, which is enough time for M to halt on w. Therefore, M' halts on x. Conversely, suppose that there exists x such that $f(x) \in CE$ and M' halts on x. Then M' computes f(x) and simulates M on w for $|f(x)|$ steps. Since $f(x) \in CE$, there exists a computation of M on w that halts within $|f(x)|$ steps. Therefore, M halts on w. There exists a computable reduction from H to $A \sqcup A^c$. Therefore, if $A \sqcup A^c$ is c.e. or co-c.e., then H is c.e. or co-c.e., respectively. But H is known to be c.e.-complete, which means that it is neither c.e. nor co-c.e. Therefore, $A \sqcup A^c$ is neither c.e. not co-c.e. ■

Problem 47.
Let B be a language that is computably enumerable. Then, there is a TM M_B that accepts all strings in B. To show that $\exists B$ is c.e., we must build a TM $M_{\exists B}$ for $\exists B$. $M_{\exists B}$ has an input of $x \in \{0,1\}^*$. Then, we have a nested for loop where the outer loop is for i=0,1,2... and the inner loop starts with for j=0, which $j \leq i$, j++. The value of i will represent the number of steps for which we simulate M_B, and j will represent the jth w that we are enumerating. Within the nested loops, we will enumerate w_j. Then, we will construct a string-pairing $\langle x, w_j \rangle$ according to the given

definition and run M_B on $\langle x, w_j \rangle$ for i steps. Then, if M_B accepts, $M_{\exists B}$ will also accept. The given definition of $M_{\exists B}$ works because it avoids infinite loops where M_B doesn't accept or reject. By running M_B on string-pairings concurrently, it allows some strings to never accept/reject without affecting other strings. Therefore, $\exists B$ is c.e. ■

Problem 48.
Let A be a c.e. language, and A = $\exists B$. Then, there is a TM $M_{\exists B}$ such that $L(M_{\exists B}) = \exists B$. Then, we must create a TM M_B for language B. The input for M_B is $\langle x, w \rangle$, where x,w $\in \{0,1\}^*$. We can then run $M_{\exists B}$ on x for $|w|$ (length of w) steps, and if $M_{\exists B}$ accepts, then M_B accepts, and otherwise it rejects. Running $M_{\exists B}$ on x for $|w|$ steps takes finite steps, so the machine won't run forever. It is guaranteed to accept or reject, which is the definition of a TM for a decidable language. Also, $x \in \exists B$, which is decided by $M_{\exists B}$, implies that there is some $\langle x, w \rangle \in$ B because $x \notin \exists B$ means that there doesn't exist a w such that $\langle x, w \rangle \in B$. By the construction of M_B, it is proven that for every c.e. language A$\subseteq \{0,1\}^*$, there is a decidable language B $\subseteq \{0,1\}^*$ such that $A = \exists B$. ■

Problem 49.
Assume that A is decidable which means there is a TM M_A such that M_A accepts all strings in A and rejects all strings not in A. Then, we can use $A_{TM} = \{\langle M, w \rangle|$ M accepts w$\}$ to show the reduction $A \leq_m A_{TM}$. M_A has an input of $\langle M \rangle$, and if M' satisfies the condition, accept, otherwise reject. The TM for A_{TM} has an input of $\langle M, w \rangle$, and we construct M' as follows: on input w, if w="01", accept, otherwise run M on x and if M accepts, then accept. (The language that M' decides will be Σ^* is M accepts x, or $\{01\}$ if M does not accept x.) Continuing in the steps for the TM for A_{TM}, we will then run M_A on the $\langle M' \rangle$ that we constructed, and if M_A accepts, accept, otherwise reject. We can see that there is a contradiction because the TM for A_{TM} is decidable, but A_{TM} is actually undecidable. By the reduction we showed, it follows that A is undecidable. ■

Problem 50.
Let $A = \{k \in N | M_k(3) \uparrow\}$. To show that A is $\leq_m$-complete for coCE, we must show that $A \in$ coCE and A is $\leq_m$-hard for coCE. Let the compliment of A be A^C. If $A \in$ coCE, then $A^C \in$ CE. Let M_{A^C} be a TM for A^C. The input for M_{A^C} is a string k. If $M_k(3)$ halts, then accept. $A^C \in CE$ because there is a TM that decides A^C. Therefore, $A \in$ coCE. To show A is $\leq_m$-hard for coCE, we can show that $H \leq_m A^C$, where H is the halting problem. This will use the same reduction process as seen in problem 49. Since $A \in$ coCE and A is $\leq_m$-hard for coCE, $\{k \in N | M_k(3) \rightarrow\}$ is $\leq_m$-complete for coCE. ■