COMS 331: Theory of Computing, Spring 2023
Homework Assignment 9

Neha Maddali
Due at 10:00PM, Wednesday, April 19, on Gradescope.

Problem 57.
We want to show that for all $n \in N$ such that $|C(s_n) - C(s_{n+1}| \leq c$. According to Theorem 7 from lecture, $C(f(n)) \leq C(n) + c_f$. Let $f(s_n) = s_{n+1}$ and $g(s_{n+1} = s_n$. Then, we have $C(f(s_n)) \leq C(s_n) + c_f$ and $C(g(s_{n+1})) \leq C(s_{n+1} + c_g$ respectfully. We know $|C(s_n) - C(s_{n+1}|$=max$(C(s_n) - C(s_{n+1}), C(s_{n+1} - C(s_n))$ by definition of absolute value. Using the equations defined earlier, we have max$(C(s_n) - C(s_{n+1}), C(s_{n+1}) - C(s_n))$ =max$(C(g(s_{n+1})) - C(f(s_n)), C(f(s_n)) - C(g(s_{n+1})))$. Then, reducing the left side of the max expression gives $C(g(s_{n+1})) - C(f(s_n)) = C(s_{n+1}) + c_g - C(s_n) - c_f$ using the equations defined earlier. Using those equations, we also know that the right side of the max expression gives $C(f(s_n)) - C(g(s_{n+1})) = C(s_n) + c_f - C(s_{n+1} - c_g$. Both of those expressions will be some constant because every value in the expression equates to a constant. So, let $C(s_{n+1}) + c_g - C(s_n) - c_f = c_a$ and $C(s_n) + c_f - C(s_{n+1}) - c_g = c_b$. Then, we have max$(c_a, c_b)$=some constant c, so it is proven that $|C(s_n) - C(s_{n+1}| \leq c$. ■

Problem 58.
First construct a function f such that f(x)=0 if $|x| \leq m$ where $m \in N$, and f(x)=max$(\{k|T(k) < |x|\})$ otherwise. By the definition of T (which is T(0)=0 and T(n+1)=$2^{T(n)}$), it follows that T(f(x))< $|x|$. Theorem 11 from lecture states that C is not computable, so there is no function that is the lower bound of C, which means that C(x)<f(x) for all $x \in \{0,1\}^*$. Then, we know there exists some x such that C(x)<f(x) and therefore T(C(x))<T(f(x))< $|x|$, which means that T(C(x))< $|x|$. Since m can be any natural number, there are infinitely many functions f, which means there are infinitely many x that will satisfy the inequality. ■

Problem 59.
Let $A \subseteq \{0,1\}^*$ be a decidable language. Corollary 9 from lecture states that C(f(n))≤log(1+n)+c. Because A is decidable, then there is some standard order enumerator E for A such that L(E)=A. Let f(n) be the n^{th} string that is printed by E, so f(n)=x which is technically $x_n \in A$. We also know that $n \leq |A \cap \{0,1\}^{\leq n}|$ because the definition of standard order enumeration. (For clarity, $|A \cap \{0,1\}^{\leq n}|$ is saying the number of strings in A such that the length is less than or equal to n). Then, C(f(n))=C(x) and $n \leq |A \cap \{0,1\}^{\leq n}|$, so using Corollary 9, we have C(x)≤log(1+$|A \cap \{0,1\}^{\leq n}|$)+c, which is what we wanted to prove. ■

Problem 60.
Let $|A \cap \{0,1\}^n| > 2^{tn}$. Also, let $|x| = n$ and for contradiction, let C(x)<tn, where $x \in A$, $n \in N$ and t is a real number between 0 and 1 (exclusive). Then, $|\{x|\ |x| = n \text{ and } C(x) < tn\}| \leq |\{x|\ C(x) < tn\}|$. It follows that $|\{x|\ |x| = n \text{ and } C(x) < tn\}| \leq |\{x|\ |x| < tn\}|$, and $|\{x|\ |x| < tn\}| =$

$2^0 + 2^1 + ... + 2^{tn-1} = 2^{(tn-1)+1} - 1 = 2^{tn} - 1$. So, we have $|\{x| \ |x| = n$ and $C(x) < tn\}| \leq 2^{tn} - 1$, which is strictly less than 2^{tn}, so $|\{x| \ |x| = n$ and $C(x) < tn\}| < 2^{tn}$. But, there is a contradiction because we know $|A \cap \{0,1\}^n| > 2^{tn}$, so there must be some x such that $|x| = n$ and C(x)≥tn. ■

Problem 61.
We know that $C(z_n) \leq C_M(z_n) + c_M$. Then, construct a TM M that outputs z_n. Let the input of M be $\langle s_m, s_n \rangle$, where s_m is $m = |\{0 \leq k < n, M_k(k) \downarrow\}|$ and s_n is $n = |z_n|$. Then, let $z = b_0 b_1 ... b_{n-1}$ and set every bit in z to 0. Then, while the number of 1's in z are less than m, for i=1,2,..., run M_k on k for k=0 to n-1 for i steps. Running for only i steps accounts for if M_k runs forever and allows M to not get stuck in a loop. Then, if $M_k(k) \downarrow$, change b_k to 1. At the end, output z. Back to out inequality we stated at the beginning, we know that $C_M(z_n)$ is the length of the input string (encoding). Since the encoding is defined as $0^{|x|}1xy$, the length of the encoding is $2|s_m| + |s_n| + 1$, so the inequality is $C(z_n) \leq 2|s_m| + |s_n| + 1 + c_m$. Since $|s_m| < |s_n|, C(z_n) \leq 3|s_n| + 1 + c_m$. Then, let constant $c = 1 + c_m$. Also, in lecture it was proven that $|s_n| = \lfloor log(n+1) \rfloor$. It follows that $C(z_n) \leq 3log(n+1) + c$. ■

Problem 62.
From a theorem in lecture, we have $C(v) \leq C_M(v) + c_M$. Then, we construct a TM M where the input is $\langle s_n, s_i \rangle$. Then, let v = the i^{th} element in the standard enumeration of $\{00, 01, 10\}^n$, and then output v. For later reference, note that the standard enumeration of $\{00, 01, 10\}^n$ has 3^n elements, so $0 \leq i \leq 3^n - 1$. Now, we have $C(v) \leq |\langle s_n, s_i \rangle| + c_M$, which by the definition of the encoding gives $C(v) \leq 2|s_n| + |s_i| + 1 + c_M$. By definition of the length of s_n and s_i, we have $C(v) \leq 2\lfloor log(n+1) \rfloor + \lfloor log(3^n - 1 + 1) \rfloor + 1 + c_M$, which can be simplified to $C(v) \leq 2log(n+1) + log(3^n) + 1 + c_M$. Then, let $c_b = 1 + c_M$, and after rearranging the terms we have $C(v) \leq n(log3) + 2log(n+1) + c_b$. ■

Problem 63.
From a theorem in lecture, we have $C(x_G) \leq C_M(x_G) + c_M$. Then construct a TM M where the input is s_n, and then we set $x_G = 1^{\binom{n}{2}}$, and then output x_G. Then $C_M(x_G)$ is the minimum length of the input to M such that M(input)=x_G, so then we have $C(x_G) \leq |s_n| + c_M$, and $|s_n| = \lfloor log(n+1) \rfloor$, so we have $C(x_G) \leq \lfloor log(n+1) \rfloor + c_M$. Then, let $c = c_M$ and we have $C(x_G) \leq log(n+1) + c$. ■