Computational Complexity - Scrimmage I (Solutions)

D. Desovski
LCSEE,
West Virginia University,
Morgantown, WV
{desovski@csee.wvu.edu}

1. Let $\Sigma = \{0,1\}$, and let $L \subseteq \Sigma^*$. Show that $(L^*)^* = L^*$.

   Proof: Observe that by the definition of Kleene closure $L^* \subseteq (L^*)^*$. Thus, we only need to show that $(L^*)^* \subseteq L^*$, by using mathematical induction on the size of the word.
   Base case : $w = \lambda$, $|w| = 0$, $w \in (L^*)^*$ and $w \in L^*$.
   Inductive hypothesis: For any w with $|w| = n$, $w \in (L^*)^* \Rightarrow w \in L^*$.
   Inductive proof: Let $|w| = n+1$. Then w has one of the following forms:

   (a) $w = k0$, $|k| = n$. According to the inductive hypothesis $k \in (L^*)^* \Rightarrow k \in L^*$. If $w = k0 \in (L^*)^* \Rightarrow k \in (L^*)^* \Rightarrow k \in L^* \Rightarrow$k0=w$\in L^*$.

   (b) $w = k1$, $|k| = n$. According to the inductive hypothesis $k \in (L^*)^* \Rightarrow k \in L^*$. If $w = k1 \in (L^*)^* \Rightarrow k \in (L^*)^* \Rightarrow k \in L^* \Rightarrow$k1=w$\in L^*$.

   □

2. Prove that the set of all functions $N \to N$ is not countable.

   Proof: This is true by Theorem 1.4. □

3. Is N^* countable?

   Proof: No, N^* is not countable. We will provide a proof by diagonalization. Assume N^* is countable, then there is an ordering of the words $w_0, w_1, w_2, \ldots$ We will refer to the digits of a particular word w by using square brackets (i.e., $w = w[0]w[1]w[2]\ldots$). Construct a new word k with every digit different from the corresponding word in the ordering. $k[i] = (w_i[i]+1)\, mod\, 10$, if $w_i[i]$ is a digit. Otherwise, if $w_i[i]$ is λ, set $k[i] = 1$. We can see that $k \in N^*$, but it is different from every word in the ordering. This implies that our assumption is false and consequently N^* is not countable. □

4. Design a Turing Machine, that given a number i, in binary, outputs $i\ div\ 3$.

   Solution: We will construct a 2 tape Turing Machine.

   (a) Write 0 on the second tape.

   (b) On the first tape keep subtracting 3 from the input in binary and count the number of subtractions by adding 1 in binary on the second tape.

   (c) When the number on the first tape becomes less than 3, the answer is found on the second tape.

   □

5. Show that the Program Termination Problem is undecidable.

   Proof: This is true by Theorem 3.1. □

6. Prove that every infinite computably enumerable set contains an infinite decidable set.

   Proof: We will use the following result which was proven in class. Homework 3.4: An infinite set is decidable if and only if it can be enumerated in increasing order by a one-to-one computable function (See Scrimmage II (Solutions)). Let g be the characteristic function of an infinite computably enumerable set A. We define $f(0) = \mu_y[g(y) = 1]$, $f(x+1) = \mu_y[g(y) = 1 \wedge f(x) < y]$. This function is one-to-one computable and induces an increasing order enumeration of an infinite subset of $A \Rightarrow$ this subset is decidable by using Homework 3.4. $\square$