Automata Theory - Final

K. Subramani
LCSEE,
West Virginia University,
Morgantown, WV
{ksmani@csee.wvu.edu}

1 Instructions

1. The final is to be turned in by 5 p.m., December 14.
2. Each question is worth 4 points.
3. Attempt as many problems as you can. You will be given partial credit, as per the policy discussed in class.
4. The solutions will be posted on the class URL.

2 Problems

1. **Diagonalization:** The Halting problem is defined as follows: Given a Turing Machine $M = \langle Q, \{0, 1\}, \Gamma, \delta, q_0, \Box, F\rangle$ and a string $w \in \Sigma^*$, determine whether M halts on w. In class, we proved that the Halting problem is undecidable using two different techniques. The first technique was constructive, where we constructed a series of Turing Machines, which led to a contradiction. The second technique was based on the observation that if the Halting problem is decidable then all recursively enumerable languages would become recursive. In this question, I am asking you to prove that the Halting problem is undecidable using diagonalization.
   Hint: Recall that the set of Turing Machines is countable and construct the table of Turing Machines presented with Turing Machines.

2. **Countability:**
   (a) The set $N = \{1, 2, 3, \ldots\}$ is known to be a countable set. Is the set $N \times N$ countable? (2 points.)
   (b) Let $I = (0, 1)$ and let $\Re^+ = (0, \infty)$. Which set has more elements? (2 points.)

3. **Language Theory:** In class we laboriously argued that the concept of undecidability did not apply to problem instances which are characterized by a single instance. Let us say that a problem P has three instances. Can such a problem be undecidable?

4. **Undecidability:** The Total-Halting problem is defined as follows: Given a Turing Machine $M = \langle Q, \Sigma, \Gamma, \delta, q_0, \Box, F\rangle$, determine whether M halt on all inputs $w \in \Sigma^*$. Is the Total-Halting Problem decidable?

5. **Properties of Regular Languages:** Let L_1 and L_2 denote two languages over an alphabet Σ. We define $cor(L_1, L_2)$ as follows:

$$cor(L_1, L_2) = \{w \in \Sigma^* \,:\, w \in L_1^c \text{ or } w \in L_2^c\}$$

   where L^c denotes the complement of language L. Show that $cor(L_1, L_2)$ is regular, if L_1 and L_2 are regular.