Problem Set 2

CSC 463

Due by February 14, 2020, 2pm

Each problem set counts for 10% of your mark. You may consult with others concerning the general approach for solving problems on assignments, but you must write up all solutions entirely on your own. Copying assignments is a serious academic offense and will be dealt with accordingly.

You may use any results discussed in lecture, the course textbook, previous problem set, or tutorial. You are encouraged to write precisely and concisely; it should be possible to write your solutions to the problem set within a page per question. Submit your work online by uploading a pdf file or image of your solutions onto Crowdmark.

In all problems, you can assume that you are given valid encodings of Turing machines, grammars etc. as input to the problem. You do not need to check for invalid encodings as part of your description of a Turing machine or reduction.

1. Let w^R be the reverse of a string $w \in \Sigma^*$. Let R be the set of Turing machine encodings $\langle M \rangle$ such that $w \in L(M)$ if and only if $w^R \in L(M)$. Is R semi-decidable? Prove your answer.

2. A Turing machine M has a **useless** state if there is some state q that is never entered on M's computation beginning on any string x. Let $U = \{\langle M \rangle : M \text{ is a Turing machine with a useless state}\}$.

   (a) Show that U is not decidable.

   (b) Show that U is co-semidecidable.

3. Prove the following extension of Rice's theorem. Let P be a non-trivial set of Turing machine encodings satisfying the following two conditions:

   (i) If $L(M_1) = L(M_2)$, then $\langle M_1 \rangle \in P$ if and only if $\langle M_2 \rangle \in P$.

   (ii) There are Turing machines M_1 and M_2 satisfying $L(M_1) \subset L(M_2)$ with $\langle M_1 \rangle \in P$ but $\langle M_2 \rangle \notin P$.

   An example of P satisfying the previous conditions is $P = \{\langle M \rangle : |L(M)| \text{ is finite}\}$.

   Prove that sets P satisfying the previous conditions are **not semidecidable.** (Hint: Construct a reduction $\overline{A_{TM}} \leq_m P$ that uses the Turing machines M_1, M_2 given by the second condition.)

4. Recall that a derivation of a string w from a context free grammar G is **leftmost** if the leftmost variable is always replaced by a rule in the grammar. For instance, for the grammar $S \mapsto 0S1S|1S0S|\epsilon$ that generates binary strings with equal numbers of zeros and ones, the derivations

$$S \mapsto 0S1S \mapsto 01S0S1S \mapsto 010S1S \mapsto 0101S \mapsto 0101,$$

and

$$S \mapsto 0S1S \mapsto 01S \mapsto 010S1S \mapsto 0101,$$

are both leftmost derivations of 0101 in S. Recall that a context-free grammar G is **ambiguous** if there is a string $w \in L(G)$ with at least two different leftmost derivations. Determining if a context-free language is ambiguous or not arises in problems related to natural language processing and programming languages.

Prove that determining if a context-free language is ambiguous or not is **undecidable.** (Hint: Reduce from Post Correspondence's Problem and consult the hint in Sipser, Chapter 5 Exercises.)