HW6

Due: Oct 22

Attach a fully filled-in cover sheet to the front of your printed homework. Your name should not appear anywhere except on the cover sheet; each individual page of the homework should include your banner ID instead.

While collaboration is encouraged in this class, please remember not to take away notes from any collaboration sessions. Also please list the names and logins of any of your collaborators at the beginning of each homework.

Please monitor Piazza, as we will post clarifications of questions there. You should hand in your solutions by 12:55 to the CSCI 51 bin on the second floor of the CIT. Late homeworks are not accepted.

Problem 1

Consider the following two languages. For each language, determine whether you can use Rice’s Theorem to prove it is undecidable. If so, use Rice’s Theorem to prove it is undecidable. If not, explain why you cannot use Rice’s Theorem, and prove it is undecidable without using Rice’s Theorem.

a. \( L_{DR} = \{ \langle M \rangle \mid \text{‘mccoy’} \in L(M) \} \)

   Note: the input alphabet of \( M \) consists of \{c, m, o, y\}.

b. \( L_{MUDD} = \{ \langle M \rangle \mid \langle M \rangle \in L(M) \} \)

Problem 2

Show that the following languages are decidable.

a. \( L_a = \{ \langle M \rangle \mid M \text{ is a DFA such that } L(M) \text{ contains a string with substring 010} \} \)

b. \( L_b = \{ \langle M, N \rangle \mid M \text{ and } N \text{ are DFAs with } L(M) = L(N) \} \)

c. \( L_c = \{ \langle M \rangle \mid M \text{ is a DFA that accepts } w \text{ whenever it accepts } w^R \} \)

Hint: for some problems, you might want to consider operations on DFAs like union and complement.
Problem 3

A Turing Machine is a Daystrom if, on each input $x$, it either halts or, if it does not halt, it eventually reaches a configuration it had previously visited. A language is Daystrommable if there exists a Daystrom that recognizes it. Show that $L$ is Daystrommable iff it is decidable.
The following questions are lab problems.

**Lab Problem 1**

In the following problem, $M$ denotes a Turing Machine. Determine whether or not each of the following languages is decidable. Justify your answer.

*Hint: Problem 3 may help.*

a. $L = \{\langle M, w \rangle \mid$ on input $w$, $M$ visits every state $q \in Q \setminus \{q_{accept}, q_{reject}\}\}$

b. $L = \{(M, w) \mid$ on input $w$, $M$'s head reaches the end of $w$: that is, $M$ reads every symbol in $w\}$

c. $L = \{(M, w) \mid$ on input $w$, at each step, $M$ only writes the symbol already on the tape (leaving the tape unchanged) or writes the blank symbol onto the tape\}$

**Lab Problem 2**

Consider the following language for TMs $M_1, M_2$:

$$L = \{\langle M_1, M_2, k \rangle \mid |L(M_1) \cap L(M_2)| \geq k\}$$

Is this language Turing-recognizable? If so, is it decidable? Prove your answers.

*Hint: You may want to use non-determinism.*

**Lab Problem 3**

So far we have learned about decidability and recognizability. In this problem, we will learn about function incomputability.

We define a function $f$ from $\{0,1\}^* \to \{0,1\}^*$ to be *computable* if there exists a TM that on every input $x$ halts and accepts with the string $f(x)$ on its tape.

We say a function is *uncomputable* if it is not computable.
We now define \textsc{BorgComplexity} of a string \( x \) to be the minimum number of states of a Turing Machine that outputs \( x \) on input the empty string \( \varepsilon \). Note: the smallest possible Turing Machine does not have to be unique.

Prove by contradiction that \textsc{BorgComplexity} is uncomputable. You may assume that there exist strings with arbitrarily large \textsc{BorgComplexity}.

(Hint: Think about the expression “the smallest positive integer that cannot be described in fewer than 100 words”. Why can’t such an integer exist?)