CSE 105, Spring 2019 - Homework 7

Due: Wednesday 05/29 midnight

Instructions Upload a single file to Gradescope for each group. All group members’ names and PIDs should be on each page of the submission. Your assignments in this class will be evaluated not only on the correctness of your answers, but on your ability to present your ideas clearly and logically. You should always explain how you arrived at your conclusions, using mathematically sound reasoning. Whether you use formal proof techniques or write a more informal argument for why something is true, your answers should always be well-supported. Your goal should be to convince the reader that your results and methods are sound.

Reading Sipser Sections 4.1, 4.2

Key Concepts TM, Decidability
1. (10 points) **True/False.** Briefly justify your answer for each statement.

1) Any subset of a decidable set is decidable.
   No: Take $L = \Sigma^*$. It is clear that $L$ is decidable and $A_{TM} \subset L$ that is undecidable.

2) Any subset of a regular language is decidable.
   No. Take $L = \Sigma^*$ again.

3) Any regular language is decidable.
   Yes. It is covered by the lecture.

4) Any decidable set is context-free.
   No. The language $L = \{0^n1^n2^n\}$ is not context-free but decidable.

5) There is a recognizable but not decidable language.
   Yes. The language $A_{TM}$ is recognizable but not decidable.

6) Recognizable sets are closed under complement.
   No. We refer to homework 6.

7) Decidable sets are closed under complement.
   Yes. We refer to homework 6.

8) Recognizable sets are closed under union.
   Yes. Given recognizable sets $L_1, L_2$ that are recognized by $M_1$ and $M_2$ respectively. We can construct Turing machine $M$ that simulates $M_1$ and $M_2$ parallelly, and $M$ accepts $x$ if either $M_1$ or $M_2$ accepts $x$.

9) Decidable sets are closed under union.
   Yes. Similar as 8).

10) For any decidable set $A$ and regular set $B$, the set $\{ x : x \in A \text{ or } x \in B \}$ is decidable.
    Yes. The language $\{ x : x \in A \text{ or } x \in B \}$ is equal to $A \cup B^c$. Since decidable languages are closed under complement and union, the claim then follows.
2. (10 points) Determine whether the following languages are decidable, recognizable, or undecidable. Briefly justify your answer for each statement.

1) \( L_1 = \{ <D,w> : D \text{ is a DFA and } w \in L(D) \} \)
   Decidable.

2) \( L_2 = \{ <N,w> : N \text{ is a NFA and } w \in L(N) \} \)
   Decidable

3) \( L_3 = \{ <P,w> : P \text{ is a PDA and } w \in L(P) \} \)
   Decidable.

4) \( L_4 = \{ <M,w> : M \text{ is a TM and } w \in L(M) \} \)
   \( L_4 \) is recognizable but not decidable. We refer to the lecture on May 22nd.

5) \( L_5 = \{ <M,w> : M \text{ is a TM and } w \notin L(M) \} \)
   \( L_5 \) is not recognizable. Since it is a complement of an undecidable language.

3. (10 points) Prove that the language
   \( F_{PDA} = \{ <P> : P \text{ is a CFG and } L(P) \text{ is a finite language} \} \)
   is decidable. (Hint: Use the Chomsky normal form. You can assume problem 2.26 (Page 157) from the textbook is correct.)

   **Solution:** We construct the following Turing Machine
   \( TM \) : On input \( <P> \)
   1). Convert \( P \) to a Chomsky normal form \( P' \), thus \( L(P) = L(P') \).
   2). Let \( m \) be the number of variables of \( P' \)
   3). Check whether there is a string \( w \in L(P') \) with length \( n < |w| < 2n \)
   4). If yes, then rejects, otherwise accept.

   Correctness: We first show that \( TM \) always stops. Since there are at most \( |\Sigma|^{2n} \) strings with length \( n < |w| < 2n \). Also, for every strings \( w \), it is decidable to check whether it is in \( L(P') \) (refer to the textbook). Thus \( TM \) always stops.

   For each \( <P> \), we first prove that
   \( L(P) \) is infinite if and only if there is a string \( w \in L(P') \) with length \( n < |w| < 2n \)
   “\( \Leftarrow \).” Assume there is a string \( w \in L(P) \) with length \( n < |w| < 2n \). Since \( P' \) contains only \( n \) variables, there exists a variable that appear twice in the deviation to generate \( w \).

   Thus there is loop in this deviation. We can run this loop arbitrary times, it implies \( L(P) \) is infinite.
“⇒”. Assume \( L(P) \) is infinite. There is a string \( w \in L(P) \) with length \(|w| > 2n\). Since \( P' \) contains only \( n \) variables, there exists a variable that appears twice in the deviation that generates \( w \). Thus there is a loop in this deviation. By removing this loop in this deviation, we get a string \( w' \in L(P) \) with length \( |w| - n < |w'| < |w| \). Then keep running on this process until we get a string \( w'' \in L(P) \) with length \( n < |w''| < 2n \). The claim then follows.

By the definition of \( TM \), it accepts \( <P> \) if and only if there is no string \( w \in L(P') \) with length \( n < |w| < 2n \). By our claim, we conclude that \( TM \) decides \( F_{PDA} \).

4. (10 points) Prove that the language
\[
L = \{<M, w, t>: M \text{ is a TM and } M \text{ runs more than } t \text{ steps on input } w\}
\]
is decidable.

**Solution**: We construct the following Turing Machine \( TM \):
- On input \( <M, w, t> \)
  1. Simulate \( M \) to run \( t \) steps on input \( w \).
  2. If \( M \) stops on \( t \) steps, then rejects, otherwise accepts.

Correctness: For each input \( <M, w, t> \), we simulate \( M \) for at most \( t \) steps, it is clear this Turing machine always stops. For each input \( <M, w, t> \), it is in \( L \) if and only if \( M \) runs more than \( t \) steps on input \( w \). In other words, the simulation won’t be stopped within \( t \) steps. Thus \( TM \) accepts \( w \).

5. (10 points) Prove that the language
\[
L = \{<D, w, p>: D \text{ is a DFA, } w \in L(D) \text{ and there is a partition } w = xyz \text{ with } 0 < |y| < p \text{ and } xy^2z \in L(D)\}
\]
is decidable.

**Solution**: We construct the following Turing machine:
- On input \( <D, w, p> \)
  1. Check whether \( w \in L(D) \)
  2. Enumerate all partitions of \( w \) to \( w = xyz \) with \( 0 < |y| < p \)
  3. For each partition, check whether \( xy^2z \in L(D) \)
4). If there is a partition that \( xy^2z \in L(D) \), \( TM \) accepts, otherwise rejects

Correctness: For each input \( w \), there are only \( w^2 \) possible partitions, and for partition \( xy^2z \), we know it is decidable to check \( \). (We refer to the lecture). Thus we know the Turing machine always stops. For each input \( <D, w, p> \), it is in \( L \) if and only if there is a partition \( w = xyz \) with \( 0 < |y| < p \) such that \( xy^2z \in L(D) \). By the construction of \( TM \), it will find this partition, and \( TM \) will accept this input. On the other hand, if there is no such partition, \( TM \) won’t accept it.

6. (10 points) Prove that the language
\[
L = \{ <M_1, M_2> : M_1, M_2 \text{ are TMs and } L(M_1) = L(M_2) \}
\]
is undecidable.

Solution: We show that \( E_{TM} = \{ <M> : M \text{ is a TM and } L(M) \text{ is empty} \} \) reduces \( L \).

Assume that \( L \) is decided by a Turing machine \( R \), we construction the following Turing machine,

\( TM \) : On input \( <M> \)
1. Let \( M' \) be a Turing machine that rejects any input, i.e., \( L(M) \) is empty.
2. Run \( R \) on \( <M, M'> \)
3. If \( R \) accepts \( <M, M'> \), then \( TM \) accepts \( <M> \), otherwise rejects.

We prove \( TM \) decides \( E_{TM} \).

For each \( <M> \) in \( E_{TM} \), we have that \( L(M) = \emptyset = L(M') \), thus \( <M, M'> \) in \( L \).

Since \( R \) decides \( L \), it accepts \( <M, M'> \), thus \( TM \) accepts \( <M> \).

For each \( <M> \) not in \( E_{TM} \), we have that \( L(M) \neq \emptyset = L(M') \), thus \( <M, M'> \) in not \( L \). Since \( R \) decides \( L \), it rejects \( <M, M'> \), thus \( TM \) rejects \( <M> \)

Thus \( TM \) decides \( E_{TM} \). On the other hand, we proved in lecture that \( E_{TM} \) is undecidable, which causes a contradiction.