Assignment 2
CMPUT 474

Pranav Wadhwa
1629510

March 17, 2022

1. A **Turning mahine with doubly infinite tape** is similar to an ordinary Turing machine, but its tape is infinite to the left as well as to the right. The tape is initially filled with blanks except for the portion that contains the input. Computations is defined as usual except that the head never encounters an end to the tape as it moves leftward. Show that this type of Turing Machine recognizes the class of Turing-recognizable languages.

Solution: Lets first examine what happen in a regular turing machine when its head is at the left-end of the tape. If the head of the machine is at the left-end and it tries to do an $\delta(q, \sigma) = (q', \sigma', L)$ but the head stays at the same place.

We want to show that Turing Machine with doubly infinite tape recognizes the class of Turing-recognizable languages. We can do this by making the new TM simulate the left bounded machine by adding a new symbol to mark the left end of the regular TM's tape and prevent the head from moving any further left off the that mark.

To simulate the doubly infinite tape by an regular TM we can use 2 tapes. As we already know that we can easily simulate multiple tapes on a single tape, we can use this knowledge to simulate the doubly infinite tape. The first tape would have the input symbol and the right bound and the second tape would have the left bound in reverse order. At the start of the computation the second tape would be black. This way we can get a regular TM to simulate a doubly infinite TM.

2. Show that the collection of decidable languages is closed under the operation of

   1. union
   2. concatenation
   3. star
   4. complementation
   5. intersection

Solution: Let there be 2 decidable languages L_1 and L_2 and their respective turing machines that accept then M_a and M_2. As both M_1 and M_2 are decidable both machine will halt.

1. union

   We construct a turing machine M that recognizes $L_1 \cup L_2$ such that i Read input string w

ii Run M_1 on string w.
If M_w return reject goto next step. Else return accept

iii Run M_2 on string w.
If M_2 return reject then return reject. Else return accept

2. concatenation

We will use the **non-deterministic** TM to prove this. The concatenation of L_1 and L_2 is defined as $L_1 \circ L_2 = \{xy | x \in L_1 \text{ and } y \in L_2\}$. Consider the non-deterministic TM M which would take an input w and partition it into a form $w = xy$ where $x \in L_1$ and $y \in L_2$.

i Read the input w and partition it into 2 string $w = xy$

ii Simulate M_1 on x and simulate M_2 on y

iii return accept if both M_1 and M_2 accept it, else return reject.

3. star

4. complementation

We have a TM M which simulates L_1. If L_1 rejects input w then M return accepts and it L_1 accepts input w then M returns rejects.

5. intersection

Intersection is similar to union. Let M be a TM which simulates $L_1 \cap L_2$ where we run an input w on both machines and M return accept only when both L_1 and L_2 return accept else it return reject.

3. Show that the collection of Turing-recognizable languages is closed under the operation of

1. union
2. concatenation
3. star
4. intersection
5. homomorphism

(Note that you can fine the definition of homomorphism on Page 93, Problem 1.66)

4. Prove the following language is decidable

$$L = \{\langle M \rangle : M \text{ is a DFA that accepts some string of the form } ww^R \text{ for } w \in \{0,1\}^*\}$$

Solution:

The string of the form ww^R for $w \in \{0,1\}^*$ is not a regular language but it is a context-free language. Let $A = \{x | \text{ x is of form } w \in \{0,1\}^*\}$.

It is already known that context-free langugaes are decidable. We need to show that M accept string of form A is decidable. We can contruct a decider TM D_M which decides

on L.
D_M on input $< M >$ where M is a DFA:

1. Construct $B = A \cap L(M)$. B is a CFL as A is a CFL. (This is done through another TM which converts the DFA into a RE and does the intersection with A)
2. If B is empty return reject else return accept.

Therefore, L is decidable.

5. Show that the following langauges are undecidable
   a) Set of descriptions of Turing machines $\langle M \rangle$ such that $\emptyset \in L(M)$
   b) Set of decriptions of pairs of Turing machines $\langle M, M' \rangle$ such that $L(M) \cap L(M') = \emptyset$. Also show that this language is recognizable.

Solution:

a) Let $L = \{\langle M \rangle | M \in TM \text{ and } \emptyset \in L(M)\}$ and we show that L is undecidable.

   Assume the contrary that L is decidable and TM M_L can decide on L. We show that M_L can be used to decide A_{TM}. Then we construct TM D such that:

   $D(\langle M \rangle)$

   1. call $M_L(\langle M \rangle)$
   2. if M_L accepts $\langle M \rangle$
      Simulate M on string x (must halt)
      if M accepts, return accept, else return reject
   3. If M_L rejects $\langle M \rangle$ return reject.

   This contradicts the undecidability of A_{TM} so L is undecidable.

b) We have 2 TM M and M' which simulate the languages $L(M)$ and $L(M')$. We know that decidable languages are closed under intersection then $L(M) \cap L(M')$ is also a decidable language which must have an equivalent TM say I.

   Let $L = \{\langle M, M' \rangle | L(M) \cap L(M') = \emptyset\} = \{\langle I \rangle | I \in TM \text{ and } L(I) = \emptyset\}$

   We know that E_{TM} is undecidable and L is also of the same form. Therefore, this language L is also undecidable.

6. Let A and B be two disjoint languages. Say that language C separates A and B if $A \subseteq C$ and $B \subset \bar{C}$. Show that any two disjoint co-recognizable languages are separable by some decidable language.
7. Say that a variable A in CFL G is usable if it appears ni some derivation of some strings $w \in G$. Given a CFG G and a variable A, consider the problem of testing whether A is usable. Formulae this problem as a language and show that it is decidable.
8. Let L be a CFL. Is $\{1\}^* \subseteq L$ a decidable problem? Is $\{1\}^* = L$ a decidable problem?

9. Consider the problem of determining whether a turing machine M on an input w ever attempts to move its head left at any point during its computation on w. Formulate this problem as a language and show that it is decidable.

10. Consider the problem of determining whether a Turing machine M on an input w ever attempts to move its head left when its head is in the left-most tape cell. Formulate this problem as a language and show that it is undecidable.