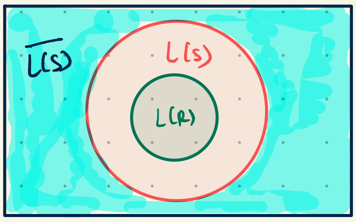
Discussion Notes : Week 8 CSE 105 Spring '23

Agenda: 4.13, 4.17, 4.18, 4.21, 4.24 from Sipser.

4.13 $A = \{ \langle R, S \rangle \mid R$ & S are regular expr. & $L(R) \subseteq L(S) \}$.

Show that A is decidable.



Soln: $L(R) \subseteq L(S) \longleftrightarrow L(R) \cap \overline{L(S)} = \emptyset$

Construct decider X as follows:

$X =$ " On input $\langle R, S \rangle$, where R & S are regular expr.
1. Construct DFA 'P' s.t. $L(P) = \overline{L(S)}$
2. Construct DFA Q s.t. $L(Q) = L(P) \cap L(R)$
3. Run the Turing Machine 'T' on input $\langle Q \rangle$, where T decides E_{DFA}.
4. If T accepts, accept. If T rejects, reject."

X is a decider because T is a decider. Also, X accepts $\langle R, S \rangle$ iff $L(R) \cap \overline{L(S)} = \emptyset \leftrightarrow X$ accepts $\langle R, S \rangle$ iff $L(R) \subseteq L(S)$. & rejects otherwise. $\therefore X$ decides A, so A is decidable.

4.17 Prove that EQ_{DFA} is decidable by testing the 2 DFAs on all strings up to a certain length. Also calculate a length that works.

Sol: Claim: If A & B are DFAs, then $L(A) = L(B)$ iff A & B accept the same strings up to length $\boxed{mn}$, where m is the # states in A & n is the # states in B.

An alternate way to state this claim: $L(A) \neq L(B) \longleftrightarrow A$ & B differ on some string 's' of length AT MOST mn.

Proof: Let 't' be the shortest string on which A & B differ (i.e A rejects & B accepts or vice versa).

Let $l = |t|$.

Suppose towards contradiction, that $l > mn$.

Let $a_0, a_1, a_2, \ldots a_l$ be the sequence of states that A enters on input t.

Let $b_0, b_1, b_2, \ldots b_l$ be the sequence of states that B enters on input t.

Since A has m states, B has n states, there are only mn distinct pairs of the form (a, b) where a is a state of A & b is a state of B.

However, there are $l+1$ pairs of the form (a_i, b_i) & by our assumption, $l > mn$, so $l+1 > mn$.

By the pigeonhole principle $\exists\, i, j$ $(a_i, b_i) = (a_j, b_j)$ (which means that $a_i = a_j$ & $b_i = b_j$).

Notice that if you remove, from t, the substring from position i to position $j-1$, you get a string (say t') such that:

(a) $|t'| < |t|$

(b) A accepts t' iff A accepts t & B accepts t' iff B accepts t } i.e A & B behave the same way on t' as they do on t.

$\therefore$ We have a found a string t', shorter than t, on which A & B differ. But this contradicts the fact that t is the shortest string on which A & B differ. Since each step followed logically from previous steps, our hypothesis must be false.

$\therefore \ell \le mn.$ $\square$

$\therefore$ EQ_{DFA} can be decided by testing the 2 DFAs on all strings up to length mn.

4.18 Show that a language C is Turing-recognizable iff $\exists D$, a decidable language, such that $C = \{x \mid \exists y,\ \langle x, y\rangle \in D\}$.

Sol: We need to prove both directions.

(a) Suppose that D exists & is decided by some T.M P.

Build a T.M Q = " On input x,

1. For each $y \in \Sigma^*$
2. Run P on input $\langle x, y\rangle$
3. If P accepts, accepts."

Clearly Q recognizes C, because if some input $x \in C$, then $\exists y$ such that $\langle x, y\rangle \in D$. Such a y will be found in some finite number of steps. However if $x \notin C$, then Q does not halt.

$\therefore$ If $\exists$ decidable D such that $C = \{x \mid \exists y, \langle x, y\rangle \in D\}$, then C is Turing-recognizable.

(b) Suppose that C is recognizable, & $\exists$ T.M M recognizing C.
Define $D = \{\langle x, y\rangle \mid M \text{ accepts } x \text{ within } |y| \text{ steps}\}$.
For every $x \in C$, $\exists k$ such that M accepts x in k steps. Suppose $y \in \Sigma^*$ & $|y| = k$, then $\langle x, y\rangle \in D$.

However, for every $x \notin C$, no such k exists. (since M will not accept x).

$\therefore C = \{ x \mid \exists y,\ \langle x, y \rangle \in D \}$

(Also, D is decidable because on input $\langle x, y \rangle$, a decider for D would just have to run M on input x for y steps, accepting if M accepts & rejecting otherwise) $\square$

4.21 $S = \{ \langle M \rangle \mid M$ is a DFA that accepts w^R whenever it accepts $w \}$. Show that S is decidable.

Sol: If A is a language, let $A^R = \{ w^R \mid w \in A \}$.

Observation: if $\langle M \rangle \in S$, then $L(M) = L(M)^R$.

Construct T.M T = "On input $\langle M \rangle$, where M is a DFA

1. Construct DFA 'N' recognizing $L(M)^R$.
   (To do this, first construct an NFA that recognizes $L(M)^R$. This can be done with the following steps:

   (a) Keep the same states as in M, & reverse the directions of all transitions in M.
   (b) Set the new accept state to be the start state of M.
   (c) Introduce a new start state (say q_0) & add ϵ-transitions from q_0 to every accept state of M.

   This NFA can then be converted to a DFA)

2. Run T.M F on input $\langle M, N \rangle$, where F decides EQ_{DFA}. If F accepts, accept. If F rejects, reject."

Clearly, T halts on every input (because F is a decider), and T only accepts ⟨M⟩ if $L(M) = L(M)^R$.
$\therefore$ T decides S, so S is decidable.

<u>4.24</u>. Define a 'useless state' in a PDA to be a state that is never entered on any input string.

Let $S = \{ \langle P \rangle \mid P \text{ is a PDA with useless states} \}$.
Prove that S is decidable.

<u>Sol</u>: Construct T.M T =

"on input ⟨P⟩, where P is a PDA

1. For each state q of P
2. Modify P so that q is the <u>only</u> accept state. (let this modified PDA be denoted as P').
3. Run T.M F on input ⟨P'⟩, where F decides E_{PDA}. If F accepts, accept. Else, continue.
4. All states have been identified as NOT useless, so reject."

If a state (q) is NOT useless, then it is reachable from the start state, so by making q the only accept state, there must be some string accepted by the modified PDA. So, if F tells us that ⟨P'⟩ belongs to E_{PDA} (meaning $L(P') = \emptyset$), then q must be useless, so ACCEPT. If all states have been checked & T hasn't yet accepted, then reject.

T is a decider because it halts on every input & T decides S, so S is decidable. $\square$