CS301 :: Final Exam

Ryan Magdaleno

January 30, 2024

Problem 1. Multiple Choice.
Note: I will only write what I chose, I got this section fully correct.

Solution ::

a. Which of the following languages is not in NP?
   $L = \emptyset$.

b. T/F. The Turing Recognizable languages are closed under complement.
   False.

c. Which of the following languages is undecidable?
   EQ_{CFG}.

d. Context-Free Languages are closed under which of the following operations:
   Kleene star.

e. T/F. All binary languages are Turing-Recognizable.
   False.

f. Which of the following is not a constrain of the pumping lemma for context-free languages?
   $|y| \geq 1$.

g. If we have DFAs M_1 with $|Q_1|$ states and M_2 with $|Q_2|$ states, how many states are in the DFA which decides their intersection?
   $|Q_1| \cdot |Q_2|$.

h. All languages in P must be at most as complex as which of the following?
   Turing Decidable Languages.

i. Which of the following is the definition of the transition function for NFAs?
$Q \times (\Sigma \cup \epsilon) \to P(Q)$.

j. What is the definition of a Turing Decidable language?
There exists a TM which accepts all strings in the language and rejects all strings not in the language.

Problem 2. Short Answer

a. Give an example of a language that is not finite and not Context-Free.
Solution ::

$$L = \{0^n 1^n 0^n \mid n \geq 0\}$$

---

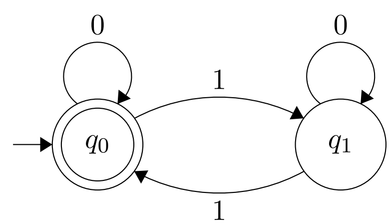
b. What does it mean for a language to be in NP?
Solution ::

It means that, that language has some TM that can solve it in a nondeterministic manner and can be verified in a polynomial time complexity like $O(n)$ for example. The time to solve it may be exponential or higher.

Also NP may be == P but unknown for now.

---

c. Provide the 5-tuple for the following DFA.



Solution ::

$$Q = \{q_0, q_1\}$$
$$\Sigma = \{0, 1\}$$
$$q_0 = q_0$$
$$F = \{q_0\}$$

$\delta =$

	0	1
q_0	q_0	q_1
q_1	q_1	q_0

d. What are the elements of the 7-tuple for a Turing Machine?
Solution ::

$$
\begin{aligned}
Q &= \text{Set of states.} \\
\Sigma &= \text{Input alphabet.} \\
\Gamma &= \text{Tape alphabet.} \\
\delta &= \text{Transition function:} \\
&\quad Q' \times \Gamma \to Q \times \Gamma \times \{L, R\} \\
&\quad Q' = Q - \{q_{accept}, q_{reject}\} \\
q_0 &= \text{Start state.} \\
q_{accept} &= \text{Accept state.} \\
q_{reject} &= \text{Reject state.} \\
&\quad q_0 \in Q \\
&\quad q_{accept} \in Q \\
&\quad q_{reject} \in Q
\end{aligned}
$$

---

e. What is the definition of a regular language?
Solution ::

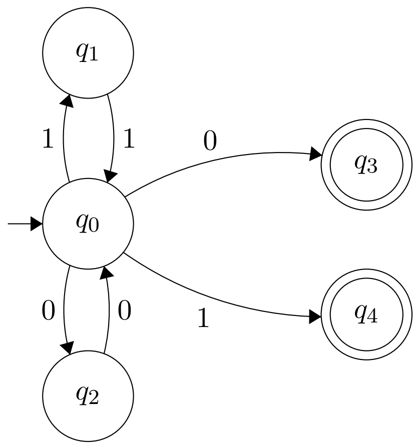
There exists a DFA which can generate that regular language.

Problem 3. NFA Construction

Give an NFA which decides the following language L. $\Sigma = \{0, 1\}$.

$$L = (11 \cup 00)^*(00^* \cup 1)$$

Solution ::



Problem 4. CFG Construction
Give a CFG which generates the following language L. $\Sigma = \{a, b\}$.

$$L = \{(a^i b^{2i} b^j a^j)^* \mid i \geq 1, j \geq 0\}$$

Solution ::

$$S \longrightarrow AZS \mid \epsilon \tag{1}$$
$$A \longrightarrow aBbb \tag{2}$$
$$B \longrightarrow \epsilon \mid aBbb \tag{3}$$
$$Z \longrightarrow \epsilon \mid bZa \tag{4}$$

Problem 5. Non-Regular Language
Prove that the following language L is not regular.

$$L = \{w \in \{0,1\}^* \mid \text{ the number of 0's is at least twice the number of 1's}\}.$$

Solution ::

Let's assume L is regular for the sake of contradiction. There must be some DFA that decides it. Therefore there must be a pumping length p.
Let $s = 1^p0^{2p}$, $s \in L$.
s can be partitioned into xy^iz, $i \geq 0$.
x must be some 1^α number of 1s, $0 \leq \alpha \leq p - \beta$.
y must be some $i\beta$ number of 1s, $1 \leq \beta \leq p - \alpha$, $i \geq 0$.
z must be the remaining 1s and 0s from s.
xy are within p, $\alpha + \beta \leq p$. If we pick $i = 2$ for y we get the following:

$$1^\beta 1^{2\beta} 1^{p-\alpha-\beta} 0^{2p} \longrightarrow 1^{\beta+p} 0^{2p}$$

However, we get the contradiction $2(p + \beta)$ 1s not being $==$ to the $2p$ 0s, because β will always be atleast 1.
$\therefore L$ is not regular.

Problem 6. Turing Decidability
Give an implementation-level description for a Turing Machine M which decides the following language L. You must include an argument that M halts.

$$L = \{0^i 1^j 2^j \mid j \geq 0, i > j\},\ \Sigma = \{0, 1, 2\}$$

Solution ::

$M =$ "On input string w,

1. Check that w is of form $0^*1^*2^*$.
2. Scan right until an unmarked 0 is found if none are found then reject, else mark 0.
3. Scan right until an unmarked 1 is found, if found mark it and go to step 5 else go to step 4.
4. Scan right until an unmarked 2 is found, if found then reject, else go to step 8.
5. Scan right until an unmarked 2 is found, mark it, if not found, reject.
6. Return head to front of tape.
7. Go to step 2.
8. Return head to front of tape and unmark rightmost 0.
9. Scan right and check if there are no unmarked 0s, reject is so. Check then that no 1 or 2 were left unmarked if so reject, else accept."

Halting Justification ::
This will halt because the TM will eventually mark all characters which will then enter an accept or reject state.

Problem 7. Undecidability Proof
Prove that the following language is undecidable via reduction.

$$THREE_{TM} = \{\langle M \rangle \mid M \text{is a TM that accepts exactly 3 strings}\}$$

Solution ::

Suppose for the sake of contradiction that $THREE_{TM}$ is decidable, therefore a TM Z decides it. Define the following TM A:

$A =$ "On input $\langle M, w \rangle$,
  $M' =$ "On input x,
    If $x \notin \{0, 1, 01\}$, reject.
    Return $M(w)$.
  Simulate Z with $\langle M' \rangle$, if M' accepts, then Z accepts, else Z rejects.

A accepts exactly when M accepts w, and rejects otherwise. This is a contradiction since we know A_{TM} is undecidable.
$\therefore THREE_{TM}$ is undecidable.