COL352: Assignment 1

Sachin 2019CS10722

January, 2022

1 Question 1

Given an alphabet $\Gamma = \{l_1, ..., l_k\}$, construct an NFA that accepts strings that don't have all the characters from Γ. Can you give an NFA with k states?

Consider the NFA N = $(Q, \Sigma, \delta, q_0, F)$ where,
$Q = 2^\Gamma = \{\phi, \{l_1\}, \{l_2\}, ...\Gamma\}$
$\Sigma = \Gamma$
$q_0 = \phi$
$F = Q \setminus \Gamma$
$\delta : Q \times \Sigma \times Q$ defined as follows:

$$\delta(q, a) = \begin{cases} q & \text{if } a \in q \\ q \bigcup \{a\} & \text{otherwise} \end{cases} \tag{1}$$

Claim: N accepts only those strings that don't have all the characters from Γ.

Proof:

Let S = $x_1 x_2 .....x_k$ be any arbitrary string, let's consider the run of s on N.
$\phi \xrightarrow{x_1} \{x_1\} \xrightarrow{x_1} \{x_1, x_2\}....... \xrightarrow{x_k} \bigcup\{x_i\}$
NOTE: {} is a set in above expression, and $\bigcup\{x_i\}$ will contain single copy of each alphabet.

1. Case 1: $\bigcup\{x_i\} = \Gamma$
   This means that string s contains all the alphabets from Γ.
   Also, final state of N = $\bigcup\{x_i\} = \Gamma \notin$ F.
   Hence, N does not accepts the string s.

2. Case 1: $\bigcup\{x_i\} \subset \Gamma$
   This means that string s does not contains all the alphabets from Γ.
   Also, final state of N (let p_k) $= \bigcup\{x_i\} \neq \Gamma => p_k \in$ F.
   Hence, N does accepts the string s.

Hence proved that N accepts only those strings that don't have all the characters from Γ.

2 Question 2

An all-NFA M is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ that accepts $x \in \Sigma^*$ if every possible state that M could be in after reading input x is a state from F . Note, in contrast, that an ordinary NFA accepts a string if some state among these possible states is an accept state. Prove that all-NFAs recognize the class of regular languages.

To prove that all-NFA's recognise the class of regular languages we need to show two things, firstly that the language accepted by all-NFA's is regular, and secondly given any regular language there exists an all-NFA which accepts it. Following are the proofs of these parts,

To Prove: Language accepted by all-NFA is regular.
Proof: Now by the definition, all-NFA M is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ that accepts $x \in \Sigma^*$ if every possible state that M could be in after reading input x is a state from F . This would mean the all-NFA's are NFA because NFA a ccepts the string even if some of the states reached after reading an input x is in accept state F. NOw we know that the language accepted by NFA is regular. Therefore the language accepted by all-NFA is also regular. Hence proved.

To Prove: For every regular language there exists an all-NFA that accepts it.
Proof: We know that for every regular language there exists a DFA which accepts it. Now the definition of a DFA M is that it is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ that accepts $x \in \Sigma^*$ if the state that M could be in after reading input x is a state from F. Now we also know that the set of states DFA M would be in after reading the input x is a singleton set (Deterministic nature) and the state belongs to F if x is accepted by DFA. So every DFA is an all-NFA. Therefore for every regular language, there exists an all-NFA that accepts it. Hence proved.

Now above two facts would imply that the all-NFA's recognize the class of regular languages.

3 Question 3

4 Question 4

5 Question 5

For any string $w = w_1w_2...w_n$ the reverse of w written w^R is the string $w_n...w_2w_1$. For any language A, let $A^R = w^R|w \in A$. Show that if A is regular, then so is A^R. In other words, regular languages are closed under the reverse operation.

As every regular language has a DFA which accepts it, let D be $(Q, \Sigma, \delta, q_0, F)$ that accepts the language A. Now we will construct an NFA N from this DFA. The steps of the construction are given below. We will then show that the language accepted by NFA is indeed A^R.

Construction: To construct this NFA N we will have to reverse all the edges of the DFA D. Also make the start state of the D as the accepting state of the N. Add ϵ transitions from the accepting states of D to a new state s in NFA. Make s the start state of the NFA.
So N is $(Q_1, \Sigma, \delta', q_0', F')$ such that

$$Q_1 = Q \cup \{s\}$$
$$F' = q_0$$
$$q_0' = s$$
$$\delta'(q_1, a) = (q_2 \mid \delta(q_2, a) = q_1 \; and \; q_1, q_2 \in Q) \cup (s \mid q_1 \in F \; and \; a = \epsilon)$$

To Prove: The language accepted by N is A^R.
Proof: Now take any string $w_1w_2....w_n$ from the language A. The path(path is state then alphabet taken then next state reached) taken by this string to accept state in D would be $q_0, w_1, q_1, w_2, q_2.....q_n, w_n, f$ where $q_i \in Q$ and $f \in F$. Now take the reverse of the string $w_nw_{n-1}....w_1$. There exist a path from start to accept state in N as which is as follows $s, \epsilon, f, w_n, q_n, w_{n-1}.....q_1, w_1, q_0$. We know that q_0 is an accept state of N. Thus we have got an NFA that has an accepting path for any string w in A^R.
Now we have proved that for every string in A we have a accepting path in N. Now we can also similarly prove the reverse direction too i.e. for every string w in A^R we have accepting path in D for reverse of w. The proof of this goes as follows:
Take any string $w_1w_2....w_n$ from the language A^R. The pathtaken by this string to accept state in N would be $s, \epsilon, f, w_1, q_n, w_2.....q_1, w_n, q_)$ where $q_i \in Q$ and $f \in F$. Now take the reverse of the string $w_nw_{n-1}....w_1$. There exist a path from start to accept state in D as which is as follows $q_0, w_n, q_1, w_{n-1}, q_2.....q_n, w_1, f$. We know that f is an accept state of D.Thus we have got an DFA that has an accepting path for any string w in A.
Hence we have proved that regular languages are closed under the reverse operation.

6 Question 6