CSC236 Assignment 3

Jiahong Zhai(1005877561)

1a.

Let Σ be a finite alphabet. The set of extension of regular expression $\mathcal{REQ}$(over Σ) is the smallest set such that:

Basis: $\emptyset$, ϵ and a (for each $a \in \Sigma$) belong to $\mathcal{REQ}$.

Induction: If R and S belong to $\mathcal{REQ}$, then (R+S), (RS), R* and R? also belong to $\mathcal{REQ}$.

1b.

P(R): $R \in \mathcal{REQ} \Longrightarrow \exists R' \in \mathcal{RE}, \mathcal{L}(R) = \mathcal{L}(R')$

WTS: $\forall R \in \mathcal{REQ}$, P(R)

Let $R \in \mathcal{REQ}$

Basis: R is $\emptyset$ or ϵ or a(for each $a \in \Sigma$)

$R \in \mathcal{RE}$, then let R' = R. We have $\mathcal{L}(R) = \mathcal{L}(R')$.

Induction: Assume T and S $\in \mathcal{REQ}$ (defined as in 1a) and P(S), P(T). Which is to say, $\exists S' \in \mathcal{RE}$, $\mathcal{L}(S) = \mathcal{L}(S')$, $\exists T' \in \mathcal{RE}$, $\mathcal{L}(T) = \mathcal{L}(T')$. We want to show P(R) for R = S+T or ST or S* or S?

Case1: R = S+T

Let R' = S' + T'.

By definition 7.7 we know, $\mathcal{L}(R) = \mathcal{L}(S+T) = \mathcal{L}(S) \cup \mathcal{L}(T) = \mathcal{L}(S') \cup \mathcal{L}(T') = \mathcal{L}(S'+T') = \mathcal{L}(R')$

So, $\mathcal{L}(R) = \mathcal{L}(R')$

Case2: R = ST

Let R' = S'T'

By definition 7.7 we know, $\mathcal{L}(R) = \mathcal{L}(ST) = \mathcal{L}(S) \circ \mathcal{L}(T) = \mathcal{L}(S') \circ \mathcal{L}(T') = \mathcal{L}(S'T') = \mathcal{L}(R')$

So, $\mathcal{L}(R) = \mathcal{L}(R')$

Case3: R = S*

Let R' = S'*

By definition 7.7 we know $\mathcal{L}(R) = \mathcal{L}(S^*) = \mathcal{L}(S)^* = \mathcal{L}(S')^* = \mathcal{L}(S'^*) = \mathcal{L}(R')$.

So, $\mathcal{L}(R) = \mathcal{L}(R')$

Case4: R = S?

Let R' = (S' + ε)

$\mathcal{L}(R) = \mathcal{L}(S?) = \mathcal{L}(S) \cup \mathcal{L}(\varepsilon) = \mathcal{L}(S') \cup \mathcal{L}(\varepsilon) = \mathcal{L}(S'+\varepsilon) = \mathcal{L}(R')$

So, $\mathcal{L}(R) = \mathcal{L}(R')$

■

2a.

For all element a in the alphabet, there is a state q_1, such that in any state if the next reading element is a, then the automata will transit to q_1.

2b.

There is only one q_1 for any element a in the alphabet. Because if there are more than one q_1 for a, then if the current state is one of the q_1s and the next reading element is a, then automata must transit to all q_1s. But this contradict to the definition of Deterministic Finite State Automata. So, there is only on q_1 for a, let's call it q_{1a}.

If one string is ending by a, then obviously the last state is q_{1a}. If $q_{1a} \in F$, then the string is accepted. So, $\mathcal{L}(M)$ is the set of all strings whose last element's q_1 is in F.

2c.

By 2b we know that for 1 there is a q_1. For 0 there is a q_1, we call it q_0 here.

Case1: $q_1 \in F$, $q_0 \in F$

$\mathcal{L}(M)$ is the set of all strings over {0,1}

Case2: $q_1 \in F$, $q_0 \notin F$

$\mathcal{L}(M)$ is the set of strings over {0,1}, whose last element is 1.

Case3: $q_1 \notin F$, $q_0 \in F$

$\mathcal{L}(M)$ is the set of strings over {0,1}, whose last element is 0.

Case4: $q_1 \notin F$, $q_0 \notin F$

$\mathcal{L}(M)$ is $\emptyset$

So, in total, 4 languages

■

3.

No

Let R be 01* over alphabet {0,1}

Let L be $\mathcal{L}(R)$.

L = {0, 01, 011, 0111, 01111, 011111……} all strings that are started by 0 and followed by any number of 1.

WTS S, that L = SS* does not exist

Case1: $\forall x \in S, 0 \notin x$

So, $\forall y \in SS^*, 0 \notin y$.

So, 0111 $\notin$ SS*, but 0111$\in$L.

So, L$\neq$SS*

So, such an S does not exist

Case2: $\exists x \in S, 0 \in x$

Let x be a0b (a and b can be 0 or 1 or ε)

Then xx which is a0ba0b$\in$SS*, but a0ba0b$\notin$L, because there are 2 0s in a0ba0b and strings in L could have only one 0.

So, L$\neq$SS*

So, such an S does not exist

So, the answerer is no

■