CS/ECE 374 Spring 2017 — Lanxiao Bai (lbai5)

Homework 2 Problem 1 — Renheng Ruan (rruan2)

1. Draw an NFA that accepts the language $\{w \mid$ there is exactly one block of 0s of even length$\}$. (A "block of 0s" is a maximal substring of 0s.)
2. (a) Draw an NFA for the regular expression $(010)^* + (01)^* + 0^*$.

   (b) Now using the powerset construction (also called the subset construcion), design a DFA for the same language. Label the states of your DFA with names that are sets of states of your NFA.

Solution: 1. The NFA is as presented bellow:

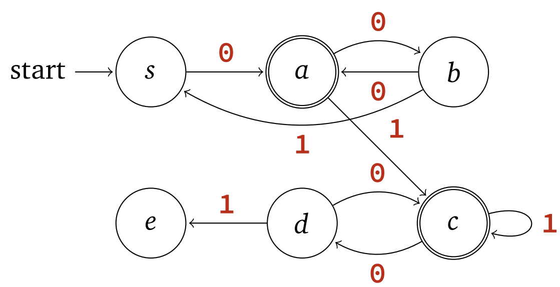

Figure 1. NFA that accepts {w | there is exactly one block of 0s of even length}

2. (a) Regular expression $(010)^* + (01)^* + 0^*$ accepted by NFA is presented below:

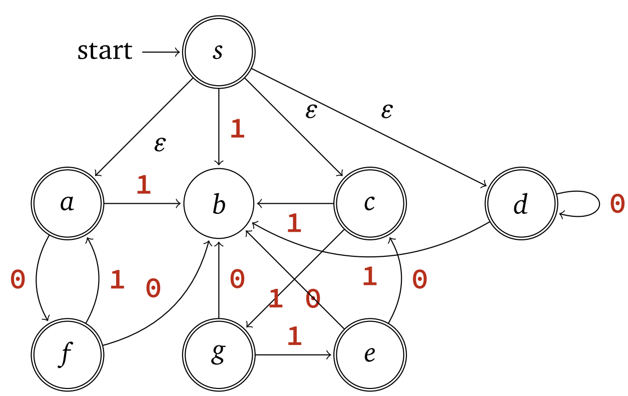

Figure 2. NFA that accepts $(010)^* + (01)^* + 0^*$

(b) Regular expression $(010)^* + (01)^* + 0^*$ accepted by DFA is presented below:

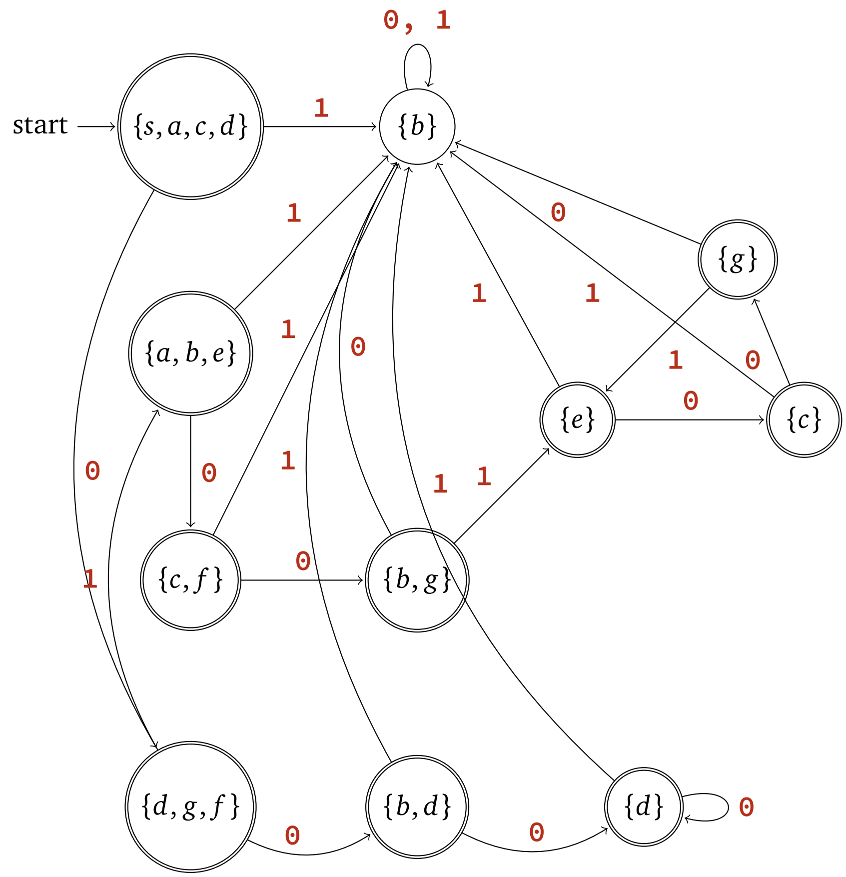

Figure 3. DFA that accepts $(010)^* + (01)^* + 0^*$

■

CS/ECE 374 Spring 2017	Lanxiao Bai (lbai5)
Homework 2 Problem 2	Renheng Ruan (rruan2)

This problem is to illustrate proofs of (the many) closure properties of regular languages.

1. For a language L let FUNKY$(L) = \{w \mid w \in L$ but no proper prefix of w is in $L\}$. Prove that if L is regular then FUNKY(L) is also regular using the following technique. Let $M = (Q, \Sigma, \delta, s, A)$ be a DFA accepting L. Describe a NFA N in terms of M that accepts FUNKY(L). Explain the construction of your NFA.
2. In Lab 3 we saw that $insert\mathbf{1}(L)$ is regular whenever L is regular. Here we consider a different proof technique. Let r be a regular expression. We would like to show that there is another regular expression r' such that $L(r') = insert\mathbf{1}(L(r))$.
   (a) For each of the base cases of regular expressions $\emptyset$, ϵ and $\{a\}, a \in \Sigma$ describe a regular expression for $insert\mathbf{1}(L(r))$.
   (b) Suppose r_1 and r_2 are regular expressions, and r_1' and r_2' are regular expressions for the languages $insert\mathbf{1}(L(r_1))$ and $insert\mathbf{1}(L(r_2))$ respectively. Describe a regular expression for the language $insert\mathbf{1}(L(r_1 + r_2))$ using r_1, r_2, r_1', r_2'.
   (c) Same as the previous part but now consider $L(r_1 r_2)$.
   (d) Same as the previous part but now consider $L((r_1)^*)$.

Solution: 1. Since FUNKY(L) contains w that none of it's proper prefix is in L, so we can formalize it as

$$\text{FUNKY}(L) = L - L\Sigma^+$$

which means that if a is an accepting states in DFA that accepts L, for all non-zero length w, $\delta(a, w)$ should not be accepted.

As a result, the NSA that accepts FUNKY(L) is $N = (Q_N, \Sigma_N, \delta_N, s_N, A_N)$ that

- $Q_N = Q$
- $\Sigma_N = \Sigma$
- $\delta_N : Q \times \Sigma \to \mathbb{P}(Q)$, that $\delta_N(q, w) = \{\delta(q', w) : q' \in \varepsilon\text{reach}(q)\}$
- $s_N = s$
- $A_N = A - \bigcup_{a \in A, w \in \{w \in L : |w| > 0\}} \delta(a, w)$

2. (a)
   - When $r = \emptyset$, $r' = \emptyset$
   - When $r = \varepsilon$, $r = 1$
   - When $r = a, a \in \Sigma$, $r' = 1r + r1$

   (b) For $insert\mathbf{1}(L(r_1 + r_2))$, $r' = r_1' + r_2'$
   (c) For $insert\mathbf{1}(L(r_1 r_2))$, $r' = r_1' r_2 + r_1 r_2'$
   (d) For $insert\mathbf{1}(L(r_1)^*)$, $r' = (r_1)^* r_1' (r_1)^*$

■