Theory of Computation Assignment no. 5

Goktug Saatcioglu

(1) (i) $A = \{a^i b^j \mid i \neq j\}$. Let $x = a^i$ where $i \geq 0$ and $y = a^j$ where $j \geq 0$. If $i \neq j$, we claim that x and y are not equivalent with respect to A. Let $z = b^j$, then $xz \in A$ but $yz \notin A$. Thus, x and y are not equivalent and A has an infinite number of non-equivalent words with respect to A. Therefore, A is not regular.

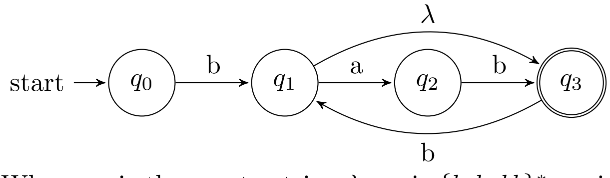
(ii) $B = \{w \in \{a, b\}^* |$either w contains 'aa' or 'bb', or $w = (ab)^{2^n}$ for some number $n\}$. Let $x = (ab)^i$ where $i \geq 1, i = 2n, n \geq 0$ and $y = (ab)^j$ where $j \geq 1, j = 2n, n \geq 0$. If $i \neq j$, we claim that x and y are not equivalent with respect to B. Let $z = (ab)^i$, then $xz \in B$ but $yz \notin B$. Thus, x and y are not equivalent and B has an infinite number of non-equivalent words with respect to B. Therefore, B is no regular.

(2) Assume for the sake of contradiction that $E = \{(ab)^n c(ab)^n \mid n \geq 0\}$ is regular. Then E has a pumping constant p. Now consider the word $w = (ab)^p c(ab)^p$ such that we can write $w = xyz$ and the properties of the pumping lemma hold. By the properties that states that $|xy| \leq p$ and $|y| > 0$, we conclude that y consists of at least single b and even $|y|$ means y consists of $|y|$ amount of ab's while an odd $|y|$ means that y consists of $|y| - 1$ amount of ab's and a single extra b. Consequently, x is then defined as the remaining letters after y is removed from $(ab)^p$. By the property $xy^n z \in E\ \forall n \geq 0$, there exists a $w' = xy^0 z$ such that $w' \in E$. If $|y|$ is odd then we remove at least a single b from $(ab)^p$ which means that w' is no longer in the form $(ab)^p c(ab)^p$. Similarly, if $|y|$ is even then we remove at least a single ab from $(ab)^p$ such that w' is now at most in the form $(ab)^{p-1} c(ab)^p$. For both cases $w' \notin E$ which is a contradiction to our initial claim that E is regular. Thus, $E \notin REG$ by the pumping lemma.

(3) By the definition of $x \sim_A y$ we know that $\forall z \in \Sigma^*\ xz \in A \iff yz \in A$. This then defines $[x]_A = \{y \in \Sigma^* \mid x \sim_A y\}$. By the symmetry of equivalence relations, we can say that $x \sim_A y \implies y \sim_A x$ meaning that $\forall z \in \Sigma^*\ yz \in A \iff xz \in A$. This then defines $[y]_A = \{x \in \Sigma^* \mid y \sim_A x\}$. Thus, we see that $[x]_A = \{y \in \Sigma^* \mid x \sim_A y\} = \{x \in \Sigma^* \mid y \sim_A x\} = [y]_A \therefore [x]_A = [y]_A$.

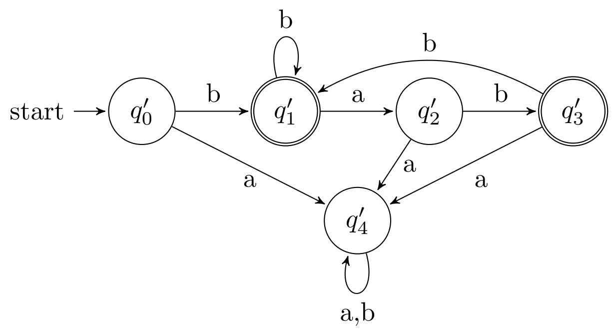
(4) Assume for the sake of contradiction that $E = \{ww \mid w \in \{a, b\}^*\}$ is regular. Then E has a pumping cosntant p. Now consider the word $w = a^p b a^p b$. Then we can write w as $w = xyz$ such that the properties of the pumping lemma hold. By the properties that states that $|xy| \leq p$ and $|y| > 0$, we can conclude that y consists of only a's (so does xy). By the property $xy^n z \in E\ \forall n \geq 0$, there exists a $w' = xy^0 z$ such that $w' \in E$. Let $k = |y|$ and because $0 < k \leq p$ we know that $w' = a^{p-k} b a^p b$. If $w' \in E$, then $a^{p-k} b a^p b$ is in the form $w'w'$ which means that $p - k = p$. However, this is not possible since $k > 0$ and we get the contradiction that $w' \notin E$. This in turn is a contradiction to our initial claim that E is regular and we conclude that E is not regular.

(5) (i) Below is the NFA that recognizes the language $L = \{b, bab\}^*$.



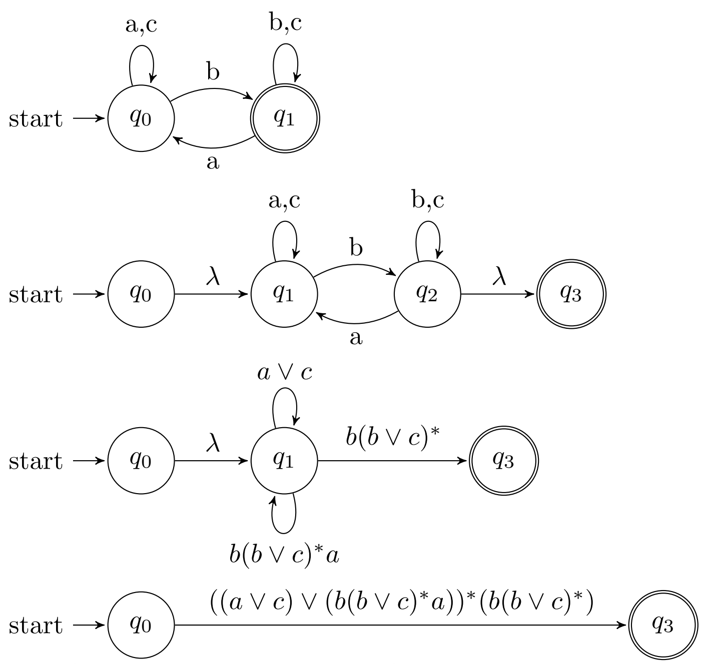
Where q_0 is the empty string λ, q_1 is $\{b, babb\}^*$, q_2 is $\{w \in L$ that end with an $a\}$, q_3 is b, bab^*.

(ii) The conversion of the NFA to a DFA is given below.



Where $q_0' = \{q_0\}$, $q_1' = \{q_1, q_3\}$, $q_2' = \{q_2\}$, $q_3' = \{q_3\}$ and $q_4' = \emptyset$. States that are not denoted by prime are defined above in (a).

(6) The steps to finding a regular expression for the NFA in Figure 2.38 are given below.



Therefore, the regular expression for the language described by the NFA is $((a \vee c) \vee (b(b \vee c)^*a))^*(b(b \vee c)^*)$.

(7) Let Prefix$(L) = \{u \,|\, \text{there is a string } v \text{ such that } uv \in L\}$. If L is regular then there is a DFA, say M, that recognizes L. Now consider the DFA M' such that all states from which an accepting state is reachable in M as accepting states for M'. This DFA now recognizes Prefix(L) since if u is a prefix of w, meaning there is a string v such that $uv = w$, and $w \in L$, then we must go over the letters of u followed by the letters of v to reach an accepting state in M. Thus, if there is a legal path that ends with an accepting state for w, we can create a DFA that identifies Prefix(L) by making all states in the path leading to the accepting state for w become accepting states. This then means that all prefixes of w (i.e. u) is now recognized by the DFA M'. M' is identical to M for its alphabet Σ, set of states Q, starting state q_0 and transition function δ, and the set of accepting states F is defined as above. If we allow for u to be λ, then we also keep that accepting states from M. Since there is a DFA M' that identifies Prefix(L), Prefix(L) is regular. Therefore, we conclude that if L is regular then so is Prefix(L).