MA2C03: TUTORIAL 9 PROBLEMS
FORMAL LANGUAGES AND GRAMMARS

1) Let L be the language over the alphabet $\{0, 1\}$ consisting of all words where the string 00 occurs as a substring.

(a) Devise a regular grammar in normal form that generates the language L. Be sure to specify the start symbol, the non-terminals, and all the production rules.

(b) Write down a regular expression that gives the language L and justify your answer.

Solution: (a) Consult the diagram of the finite state acceptor we built during the tutorial last week. The drawing is on the last page of the solutions. We shall use the algorithm discussed in lecture in order to generate the corresponding regular grammar in normal form. The finite state acceptor had three states $\{i, A, B\}$, where i was the initial state. Correspondingly, we use three non-terminals in our regular grammar: the start symbol $\langle S\rangle$ corresponding to the initial state i, $\langle A\rangle$ corresponding to state A, and $\langle B\rangle$ corresponding to state B. We first write the production rules corresponding to the transitions out of the initial state i :

(1) $\langle S\rangle \to 1\langle S\rangle$.
(2) $\langle S\rangle \to 0\langle A\rangle$.

Next, we write the production rules corresponding to the transitions out of state A :

(3) $\langle A\rangle \to 1\langle S\rangle$.
(4) $\langle A\rangle \to 0\langle B\rangle$.

Finally, we write the production rules corresponding to the transitions out of state B :

(5) $\langle B\rangle \to 1\langle B\rangle$.
(6) $\langle B\rangle \to 0\langle B\rangle$.

Rules (1)-(6) are of type (i). For each accepting state, we will write down a rule of type (iii). Since there is only one accepting state, B, we have only one such rule:

(7) $\langle B\rangle \to \epsilon$.

(b) Recall from last week's tutorial that

$$L = \{w \in A^* \mid w = u \circ 00 \circ v \quad u, v \in A^*\}.$$

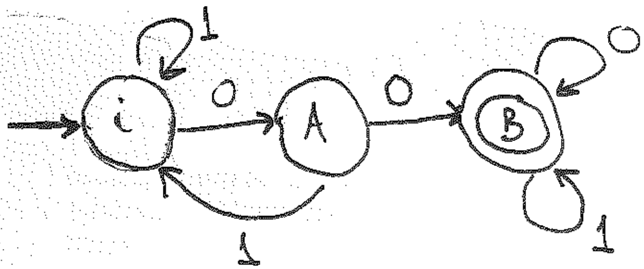
Therefore, $L = A^* \circ 00 \circ A^*$, and we have obtained the regular expression giving us the language L. Compare this solution to last week's tutorial where we proved this language was regular by applying the definition of a regular language.

2) (Annual Exam 2017) Consider the language L over the alphabet $A = \{a, l, p\}$ consisting of all words of the form $a^m l^{2m} p^m$ for $m \in \mathbb{N}^*$. Use the Pumping Lemma to show the language L is not regular.

Solution: Assume L is regular. Then it must have a pumping length P. We will now choose a string in terms of P that is particularly easy to analyse in the setting of the Pumping Lemma. Let this string be $w = a^P l^{2P} p^P$. By the Pumping Lemma, we can break w into three components: x, u, and y, with $u \neq \epsilon$ and $|xu| \leq P$.

Note that if $|xu| \leq P$, then xu must consist of a's as the first P characters in w are a's. Also if $u \neq \epsilon$, we must conclude $u = a^k$ for some $k \geq 1$.

Therefore, by the Pumping Lemma, for all n, $xu^n y \in L$. By choosing $n = 2$, however, we obtain a string $xu^2y = a^q l^{2P} p^P$ with $q > P$. This string cannot be in L. We thus have obtained the needed contradiction showing that the language L cannot be regular.



1
0
0
0
i
A
B
1
1