Assignment 2

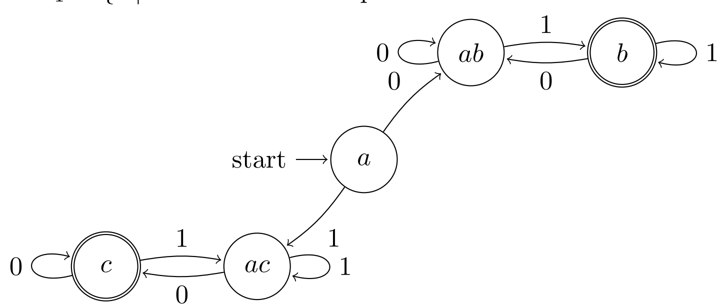
Aleksandr Salo

Due September 16, 2014

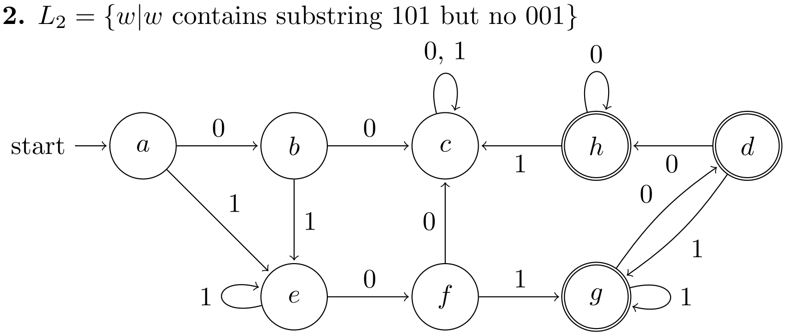
1 Designing FSA

Assume $\Sigma = \{0, 1, \}$

1. $L_1 = \{w|w \text{ contains an unequal number of 01 and 10 substrings}\}$



2. $L_2 = \{w|w \text{ contains substring 101 but no 001}\}$



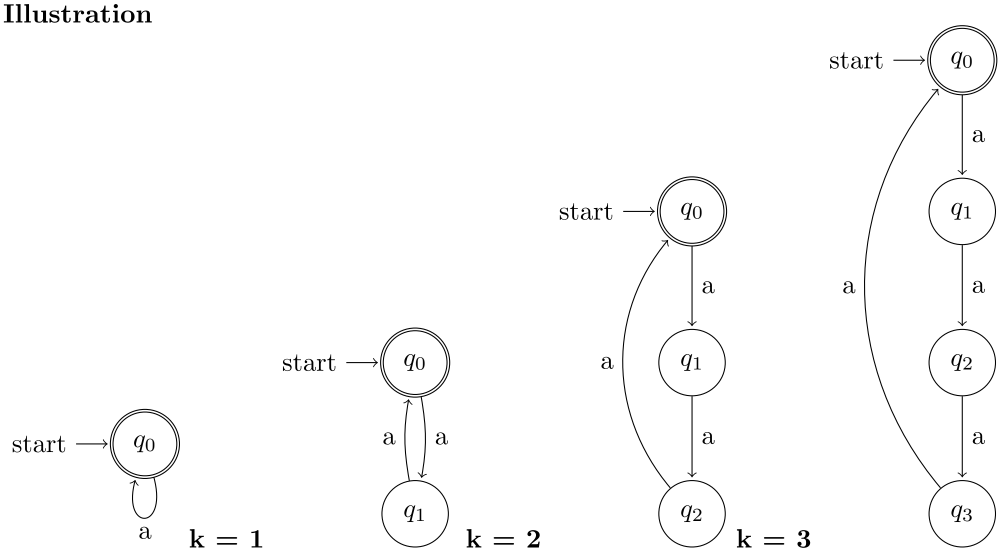
2 Proof of regularity

1.36 Let $B_n = \{a^k | k \text{ is a multiple of n}\}$. Show that for each $n \geq 1$, the language B_n is regular.
Proof (by construction)
1. The language is regular if there exists a DFA that recognizes it.
2. The alphabet Σ for any of the languages B_n is {a}.
3. Let us for any n construct the DFA in the following fashion:

$$Q = \{q_i | i \text{ is an integer in } [0, n-1]\}$$
$$\Sigma = \{a\}$$
$$\delta:$$
$$\delta(q_i, s) = \begin{cases} q_{i+1} | \text{ for any integer } i \text{ in } [0, n-2] \\ q_0 | \text{for } i = n-1 \end{cases}$$
$$q_0 = q_0$$
$$F = \{q_0\}.$$

4. Note, that this DFA would accept empty string as it is a multiple of anything. Besides it will accept k input symbols (making k transitions) if the k equals to n. Then, DFA, consuming next symbols, will go to its following states q_i, and it will take exactly k - 1 symbols (and transitions) to go to the state labeled as q_{n-1}. Then, on k-th symbol DFA will return to its initial and accepting state. That will repeat. Hence, only providing k is multiple of n that constructed above DFA will ever accept thus making the language B_n regular language.

Illustration



k = 1 **k = 2** **k = 3** **k = 4**

3 Proving languages to be non-regular

1. Prove that L_{abb} is not R.L.
Proof (by contradiction):
1. Assume L_{abb} is a regular language (for the sake of the following contradiction).
2. Thus pumping lemma applies, let p be the pumping length for L_{abb}.
3. Let $s = a^{2p}b^{4p}$ ($|s| \geq p, s \in L$ - conditions of P.L. are satisfied)
4. Show there is **no way** to divide s into three parts in a way that fulfills the P.L. conditions.
4.1. By applying 3^{rd} condition of P.L. we conclude that xy must not contain any b
4.2. By applying 2^{nd} condition of P.L. we conclude that y must contain at least one a
4.3. Consider string $xy^2z = xyyz$, which has more number of a that half the number of b.
5. That leads us to contradiction because $xyyz$ is not in L_{abb}.
6. Hence there is no way to divide $s = a^{2p}b^{4p}$ into three parts in a way that fulfills the P.L. conditions.
7. Therefore L_{abb} is **not** a regular language.

2.

1.46 Prove language is not a R.L. using P.L. and the closure property.

a. $L_a = \{0^n 1^m 0^n | m, n \geq 0\}$
Proof (by contradiction):
1. Assume L_a is a regular language (for the sake of the following contradiction).
2. Thus pumping lemma applies, let p be the pumping length for L_a.
3. Let $s = 0^p 1^p 0^p$ ($|s| \geq p, s \in L$ - conditions of P.L. are satisfied)
4. Show there is **no way** to divide s into three parts in a way that fulfills the P.L. conditions.
4.1. By applying 3^{rd} condition of P.L. we conclude that xy must not contain any 1
4.2. By applying 2^{nd} condition of P.L. we conclude that y must contain at least one 0
4.3. Consider string $xy^2z = xyyz$, which has more number of 0 before the 1 than after the 1.
5. That leads us to contradiction because $xyyz$ is not in L_a.
6. Hence there is no way to divide $s = 0^p 1^p 0^p$ into three parts in a way that fulfills the P.L. conditions.
7. Therefore L_a is **not** a regular language.

c. $L_c = \{w | w \in \{0,1\}^*$ is not a palindrome$\}$
Proof outline
1. Show that regular languages are closed under complement operation.
2. Prove with P.L. that $\overline{L_c}$ (complement) is not a regular language.
3. Derive that L_c itself not regular language.

Proof (by contradiction):
1. By definition, the complement of a language L (with respect to an alphabet Σ such that Σ^* contains L) is Σ^* - L.
2. Since Σ^* is regular (closure under star operator), the complement of a regular language is always regular as it consist of Σ^* without L.
3. Let us prove (using P.L.) that complement language $\overline{L_c}$ is not regular (that will prove that L_c itself is not regular).
4. Deriving the complement we found, that $\overline{L_c} = \{w | w$ is binary palindrome$\}$.
5. Assume $\overline{L_c}$ is a regular language (for the sake of the following contradiction).
6. Thus pumping lemma applies, let p be the pumping length for $\overline{L_c}$.
7. Let $s = 0^p 1^p 0^p$ ($|s| \geq p, s \in \overline{L_c}$ - conditions of P.L. are satisfied)
8. Show there is **no way** to divide s into three parts in a way that fulfills the P.L. conditions.
8.1. By applying 3^{rd} condition of P.L. we conclude that xy must not contain any 1
8.2. By applying 2^{nd} condition of P.L. we conclude that y must contain at least one 0
8.3. Consider string $xy^2z = xyyz$, which has more number of 0 before the 1 than after the 1.

9. That leads us to contradiction because $xyyz$ is not in $\overline{L_c}$.
10. Hence there is no way to divide $s = 0^p1^p0^p$ into three parts in a way that fulfills the P.L. conditions.
11. Therefore $\overline{L_c}$ is **not** a regular language.
12. We already proved that regular languages are closed under the complement operations. Thus, given $\overline{L_c}$ is not regular, L_c could not be regular.

1.53: Let $\Sigma = \{0, 1, +, =\}$ and $ADD = \{x = y + z | x, y, z$ are binary integers, and x is the sum of y and z$\}$. Show that ADD is not R.L.

Proof: (by contradiction)

1. Assume ADD is a regular language (for the sake of the following contradiction).
2. Let p be the pumping length for ADD.
3. Let s = abc.
4. Note that + and = may appear once once each in s. Hence b cannot contain + nor = because it would lead to the trivial contradiction (we could not pump more equal signs into equation).
5. Let choose s as $1^p = 1^p + 0^p$. In words, let x and y not contain zeros at the beginning and z equal to zero.
6. Let us show that there is no way to divide s into 3 parts in a way that fulfills the pumping lemma conditions.
   (a) Applying pumping lemma condition $|ab| < p, ab$ must not contain only 1's.
   (b) Applying pumping lemma condition $|b| > 0$, b must contain at least one 1.
   (c) Applying pumping lemma condition $\forall i \geq 0, ab^ic \in ADD$, consider ab^2c which has changed the x-value without changing any of either y or z values. Hence equation (x = y + z) doesn't hold true anymore (we deliberately chose x not to contain leading zeros, so it's values must change with any new symbol replicated).
7. Thus, there is no way to divide s as $1^p = 1^p + 0^p$ into 3 parts that satisfy the pumping lemma conditions. This is contradiction. ADD must not be regular.