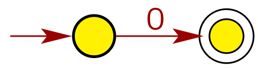
CS 341: Foundations of Computer Science II
Prof. Marvin Nakayama

Homework 4

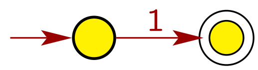
1. Use the procedure described in Lemma 1.55 to convert the regular expression $(((\texttt{00})^*(\texttt{11}))\cup \texttt{01})^*$ into an NFA.

   Answer:

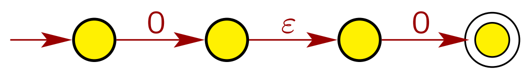
0



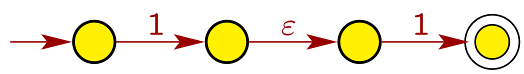
1



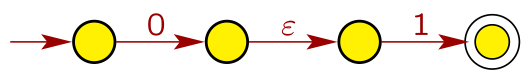
00



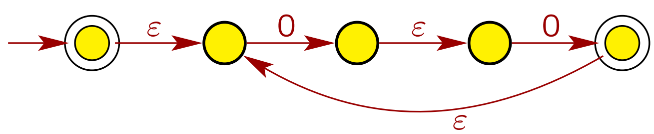
11



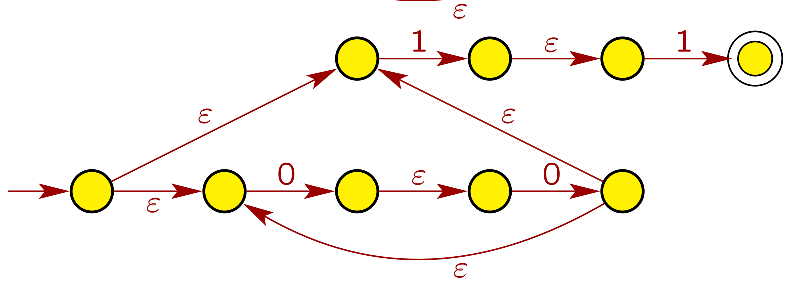
01



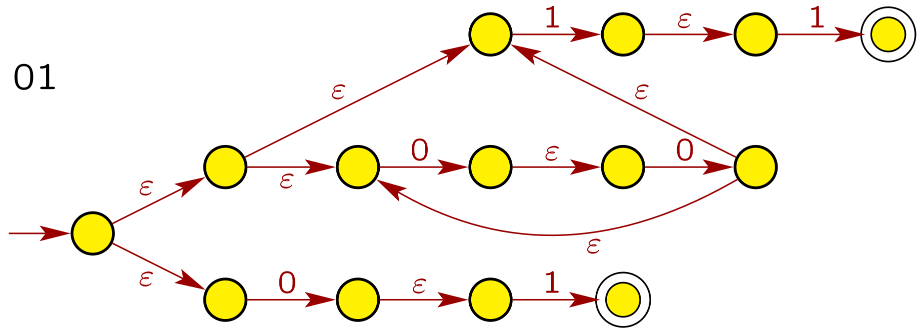
(00)*



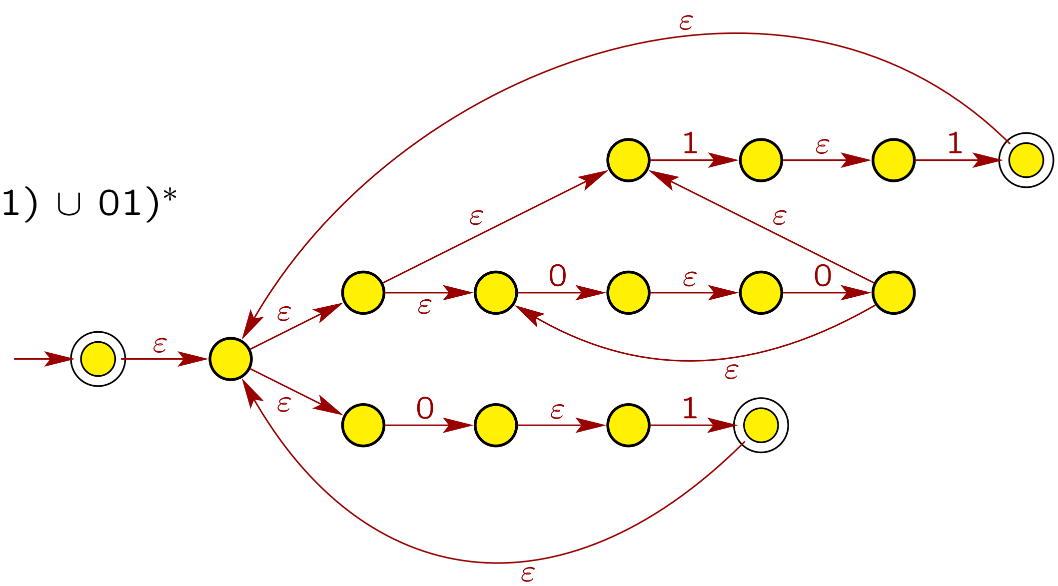
(00)*(11)



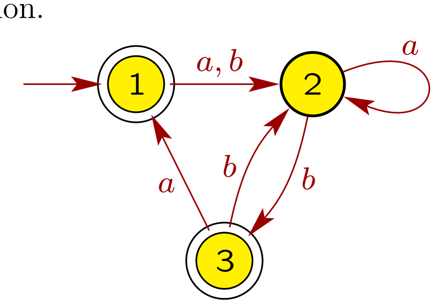
$(00)^*(11) \cup 01$



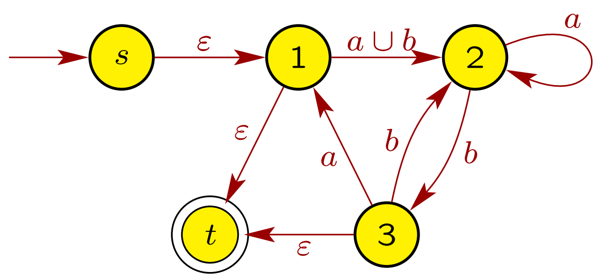
$((00)^*(11) \cup 01)^*$



2. Use the procedure described in Lemma 1.60 to convert the following DFA M to a regular expression.



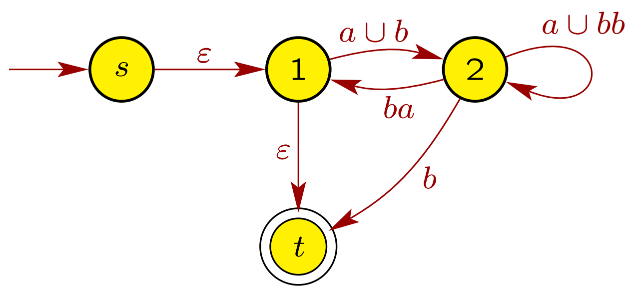
Answer: First convert DFA M into an equivalent GNFA G.



Next, we eliminate the states of G (except for s and t) one at a time. The order in which the states are eliminated does not matter. However, eliminating states in a different order from what is done below may result in a different (but also correct) regular expression. We first eliminate state 3. To do this, we need to account for the paths

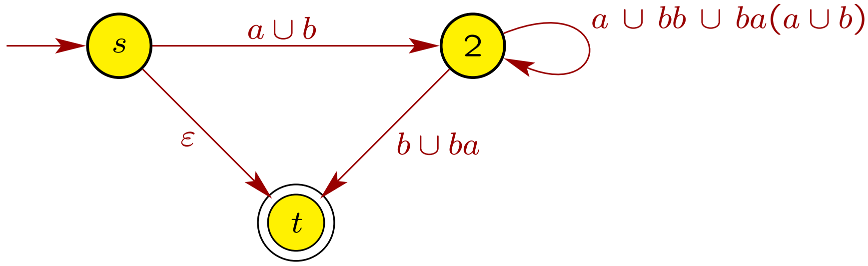
- $2 \to 3 \to 1$, which will create an arc from 2 to 1 labelled with ba;
- $2 \to 3 \to 2$, which will create an arc from 2 to 2 labelled with bb; and
- $2 \to 3 \to t$, which will create an arc from 2 to t labelled with $b\varepsilon = b$.

We combine the previous arc from 2 to 2 labelled a with the new one labelled bb to get the new label $a \cup bb$.

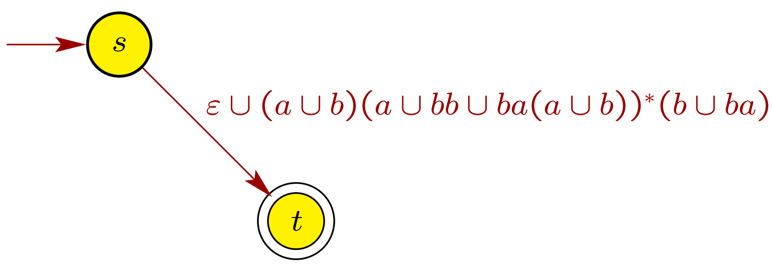


We next eliminate state 1. To do this, we need to account for the following paths:

- $s \to 1 \to 2$, which will create an arc from s to 2 labelled with $\varepsilon(a \cup b) = a \cup b$.
- $s \to 1 \to t$, which will create an arc from s to t labelled with $\varepsilon\varepsilon = \varepsilon$.
- $2 \to 1 \to 2$, which will create an arc from 2 to 2 labelled with $ba(a \cup b)$. We combine this with the existing 2 to 2 arc to get the new label $a \cup bb \cup ba(a \cup b)$.
- $2 \to 1 \to t$, which will create an arc from 2 to t labelled with $ba\varepsilon = ba$. We combine this arc with the existing arc from 2 to t to get the new label $b \cup ba$.



Finally, we eliminate state 2 by adding an arc from s to t labelled $(a \cup b)(a \cup bb \cup ba(a \cup b))^*(b \cup ba)$. We then combine this with the existing s to t arc to get the new label $\varepsilon \cup (a \cup b)(a \cup bb \cup ba(a \cup b))^*(b \cup ba)$.



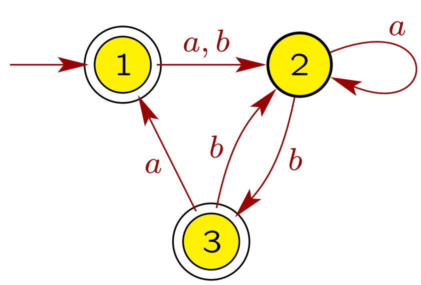
So a regular expression for the language $L(M)$ recognized by the DFA M is

$$\varepsilon \cup (a \cup b)(a \cup bb \cup ba(a \cup b))^*(b \cup ba).$$

Writing this as

$$\underbrace{\varepsilon}_{\text{stay in } 1} \cup \underbrace{(a \cup b)}_{1 \text{ to } 2}\underbrace{(a \cup bb \cup ba(a \cup b))^*}_{(2 \text{ to } 2)^*}\underbrace{(b \cup ba)}_{\text{end in } 3 \text{ or } 1}$$

should make it clear how the regular expression accounts for every path that starts in 1 and ends in either 3 or 1, which are the accepting states of the given DFA.



3. Prove that the following languages are not regular.

   (a) $A_1 = \{\, www \mid w \in \{a, b\}^* \,\}$

   Answer: Suppose that A_1 is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string $s = a^p b a^p b a^p b$. Note that $s \in A_1$

since $s = (a^p b)^3$, and $|s| = 3(p+1) \geq p$, so the Pumping Lemma will hold. Thus, we can split the string s into 3 parts $s = xyz$ satisfying the conditions

i. $xy^i z \in A_1$ for each $i \geq 0$,
ii. $|y| > 0$,
iii. $|xy| \leq p$.

Since the first p symbols of s are all a's, the third condition implies that x and y consist only of a's. So z will be the rest of the first set of a's, followed by $ba^p ba^p b$. The second condition states that $|y| > 0$, so y has at least one a. More precisely, we can then say that

$$\begin{aligned} x &= a^j \text{ for some } j \geq 0, \\ y &= a^k \text{ for some } k \geq 1, \\ z &= a^m ba^p ba^p b \text{ for some } m \geq 0. \end{aligned}$$

Since $a^p ba^p ba^p b = s = xyz = a^j a^k a^m ba^p ba^p b = a^{j+k+m} ba^p ba^p b$, we must have that $j + k + m = p$. The first condition implies that $xy^2 z \in A_1$, but

$$\begin{aligned} xy^2 z &= a^j a^k a^k a^m ba^p ba^p b \\ &= a^{p+k} ba^p ba^p b \end{aligned}$$

since $j + k + m = p$. Hence, $xy^2 z \notin A_1$ because $k \geq 1$, and we get a contradiction. Therefore, A_1 is a nonregular language.

(b) $A_2 = \{ w \in \{a, b\}^* \mid w = w^{\mathcal{R}} \}$.

Answer: Suppose that A_2 is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string $s = a^p ba^p$. Note that $s \in A_2$ since $s = s^{\mathcal{R}}$, and $|s| = 2p + 1 \geq p$, so the Pumping Lemma will hold. Thus, we can split the string s into 3 parts $s = xyz$ satisfying the conditions

i. $xy^i z \in A_2$ for each $i \geq 0$,
ii. $|y| > 0$,
iii. $|xy| \leq p$.

Since the first p symbols of s are all a's, the third condition implies that x and y consist only of a's. So z will be the rest of the first set of a's, followed by ba^p. The second condition states that $|y| > 0$, so y has at least one a. More precisely, we can then say that

$$\begin{aligned} x &= a^j \text{ for some } j \geq 0, \\ y &= a^k \text{ for some } k \geq 1, \\ z &= a^m ba^p \text{ for some } m \geq 0. \end{aligned}$$

Since $a^p ba^p = s = xyz = a^j a^k a^m ba^p = a^{j+k+m} ba^p$, we must have that $j + k + m = p$. The first condition implies that $xy^2 z \in A_2$, but

$$\begin{aligned} xy^2 z &= a^j a^k a^k a^m ba^p \\ &= a^{p+k} ba^p \end{aligned}$$

since $j + k + m = p$. Hence, $xy^2z \notin A_2$ because $(a^{p+k}ba^p)^{\mathcal{R}} = a^pba^{p+k} \neq a^{p+k}ba^p$ since $k \geq 1$, and we get a contradiction. Therefore, A_2 is a nonregular language.

(c) $A_3 = \{ a^{2n}b^{3n}a^n \mid n \geq 0 \}$.

Answer: Suppose that A_3 is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string $s = a^{2p}b^{3p}a^p$. Note that $s \in A_3$, and $|s| = 6p \geq p$, so the Pumping Lemma will hold. Thus, we can split the string s into 3 parts $s = xyz$ satisfying the conditions

i. $xy^iz \in A_3$ for each $i \geq 0$,
ii. $|y| > 0$,
iii. $|xy| \leq p$.

Since the first p symbols of s are all a's, the third condition implies that x and y consist only of a's. So z will be the rest of the first set of a's, followed by $b^{3p}a^p$. The second condition states that $|y| > 0$, so y has at least one a. More precisely, we can then say that

$$\begin{aligned} x &= a^j \text{ for some } j \geq 0, \\ y &= a^k \text{ for some } k \geq 1, \\ z &= a^m b^{3p} a^p \text{ for some } m \geq 0. \end{aligned}$$

Since $a^{2p}b^{3p}a^p = s = xyz = a^ja^ka^mb^{3p}a^p = a^{j+k+m}b^{3p}a^p$, we must have that $j + k + m = 2p$. The first condition implies that $xy^2z \in A_3$, but

$$\begin{aligned} xy^2z &= a^ja^ka^ka^mb^{3p}a^p \\ &= a^{2p+k}b^{3p}a^p \end{aligned}$$

since $j + k + m = 2p$. Hence, $xy^2z \notin A_3$ because $k \geq 1$, and we get a contradiction. Therefore, A_3 is a nonregular language.

(d) $A_4 = \{ w \in \{a, b\}^* \mid w \text{ has more } a\text{'s than } b\text{'s} \}$.

Answer: Suppose that A_4 is a regular language. Let p be the "pumping length" of the Pumping Lemma. Consider the string $s = b^pa^{p+1}$. Note that $s \in A_4$, and $|s| = 2p + 1 \geq p$, so the Pumping Lemma will hold. Thus, we can split the string s into 3 parts $s = xyz$ satisfying the conditions

i. $xy^iz \in A_4$ for each $i \geq 0$,
ii. $|y| > 0$,
iii. $|xy| \leq p$.

Since the first p symbols of s are all b's, the third condition implies that x and y consist only of b's. So z will be the rest of the b's, followed by a^{p+1}. The second condition states that $|y| > 0$, so y has at least one b. More precisely, we can then say that

$$\begin{aligned} x &= b^j \text{ for some } j \geq 0, \\ y &= b^k \text{ for some } k \geq 1, \\ z &= b^m a^{p+1} \text{ for some } m \geq 0. \end{aligned}$$

Since $b^p a^{p+1} = s = xyz = b^j b^k b^m a^{p+1} = b^{j+k+m} a^{p+1}$, we must have that $j + k + m = p$. The first condition implies that $xy^2z \in A_4$, but

$$\begin{aligned} xy^2z &= b^j b^k b^k b^m a^{p+1} \\ &= b^{p+k} a^{p+1} \end{aligned}$$

since $j + k + m = p$. Hence, $xy^2z \notin A_4$ because it doesn't have more a's than b's since $k \geq 1$, and we get a contradiction. Therefore, A_4 is a nonregular language.

4. Suppose that language A is recognized by an NFA N, and language B is the collection of strings *not* accepted by some DFA M. Prove that $A \circ B$ is a regular language.

   Answer: Since A is recognized by an NFA, we know that A is regular since a language is regular if and only if it is recognized by an NFA (Corollary 1.20). Note that the DFA M recognizes the language $\overline{B}$, the complement of B. Since $\overline{B}$ is recognized by a DFA, by definition, $\overline{B}$ is regular. We know from a problem on the previous homework that $\overline{B}$ being regular implies that its complement $\overline{\overline{B}}$ is regular. ($\overline{\overline{B}}$ is the complement of the complement of B.) But $\overline{\overline{B}} = B$, so B is regular. Since A and B are regular, their concatenation $A \circ B$ is regular by Theorem 1.23.

5. (a) Prove that if we add a finite set of strings to a regular language, the result is a regular language.

   Answer: Let A be a regular language, and let B be a finite set of strings. We know from class (see page 1-95 of Lecture Notes for Chapter 1) that finite languages are regular, so B is regular. Thus, $A \cup B$ is regular since the class of regular languages is closed under union (Theorem 1.22).

   (b) Prove that if we remove a finite set of strings from a regular language, the result is a regular language.

   Answer: Let A be a regular language, and let B be a finite set of strings with $B \subseteq A$. Let C be the language resulting from removing B from A, i.e., $C = A - B$. As we argued in the previous part, B is regular. Note that $C = A - B = A \cap \overline{B}$. Since B is regular, $\overline{B}$ is regular since the class of regular languages is closed under complement. We proved in an earlier homework that the class of regular languages is closed under intersection, so $A \cap \overline{B}$ is regular since A and $\overline{B}$ are regular. Therefore, $A - B$ is regular.

(c) Prove that if we add a finite set of strings to a nonregular language, the result is a nonregular language.

Answer: Let A be a nonregular language, and let B be a finite set of strings. We want to add B to A, so we may assume that none of the strings in B are in A, i.e., $A \cap B = \emptyset$. Let C be the language obtained by adding B to A, i.e., $C = A \cup B$. Suppose that C is regular, and we now show this is impossible. Since $A \cap B = \emptyset$, we have that $A = C - B$. Since C and B are regular, the previous part of this problem implies that $C - B$ should be regular, but we assumed that $A = C - B$ is nonregular, so we get a contradiction.

(d) Prove that if we remove a finite set of strings from a nonregular language, the result is a nonregular language.

Answer: Let A be a nonregular language, and let B be a finite set of strings, where $B \subseteq A$. Let C be the language obtained by removing B from A, i.e., $C = A - B$. Suppose that C is regular, and we now show this is impossible. Since we removed B from A to get C, we must have that $C \cap B = \emptyset$, so $A = C \cup B$. Now C is regular by assumption and B is regular since it's finite, so $C \cup B$ must be regular by Theorem 1.25. But we assumed that $A = C \cup B$ is nonregular, so we get a contradiction.

6. Consider the following statement: "If A is a nonregular language and B is a language such that $B \subseteq A$, then B must be nonregular." If the statement is true, give a proof. If it is not true, give a counterexample showing that the statement doesn't always hold.

Answer: The statement is not always true. For example, we know that the language $A = \{ \mathtt{0}^j \mathtt{1}^j \mid j \geq 0\}$ is nonregular. Define the language $B = \{\mathtt{01}\}$, and note that $B \subseteq A$. However, B is finite, so we know that it is regular.