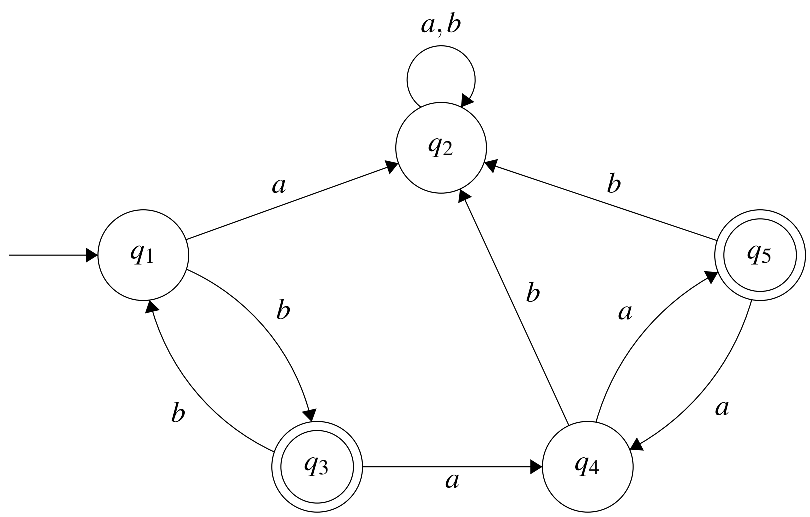
# Finite Automata Homework 3

Due: Thursday, Sep 24
Alexander Powell

1. Let $D = \{w|w$ contains an even number of as and an odd number of bs and does not contain the substring ab$\}$. Give a DFA with five states that recognizes $D$ and a regular expression that generates $D$.

   The DFA that accepts the language $D$ is displayed below.

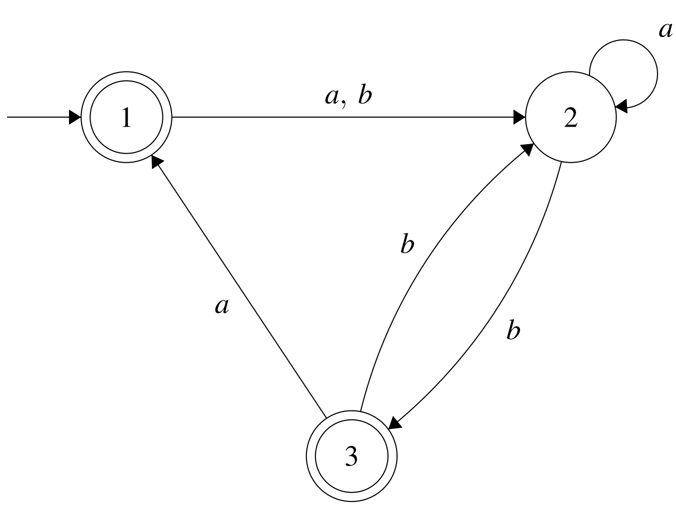



   The regular expression that generates $D$ is written as

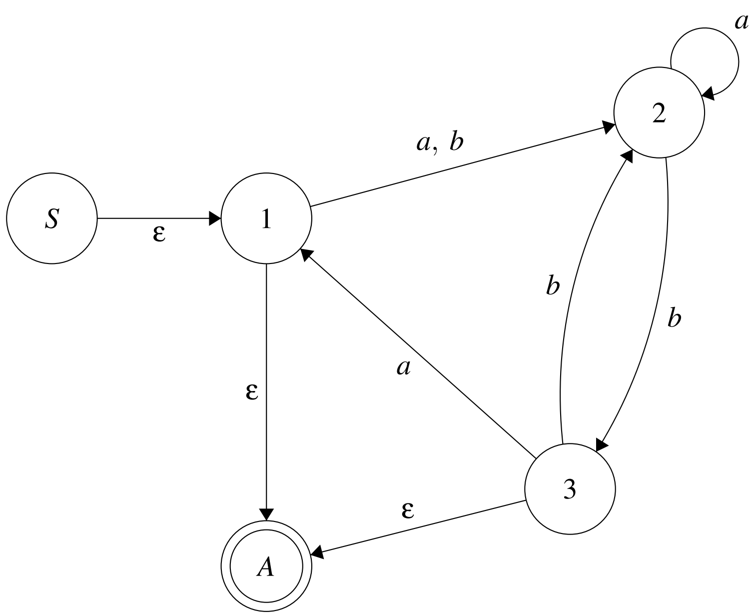
$$b(bb)^*(aa)^*$$

2. Use the procedure described in Lemma 1.60 to convert the following finite automata to regular expressions.

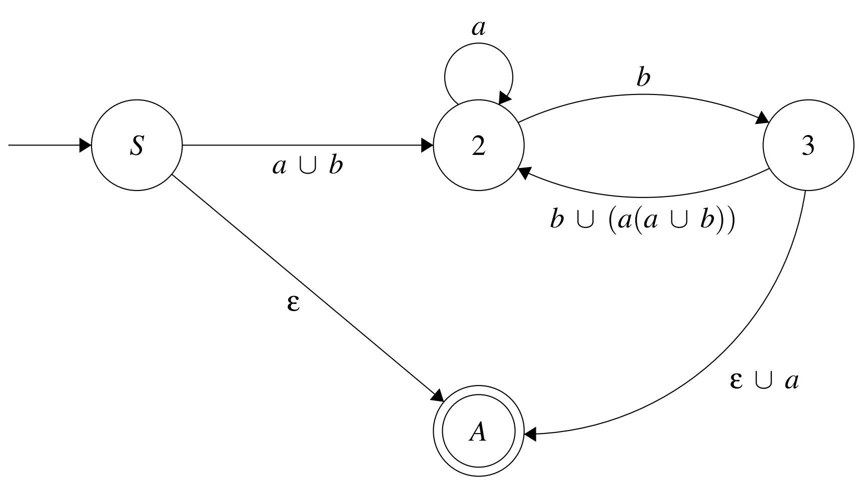
   The given finite automata is shown below.



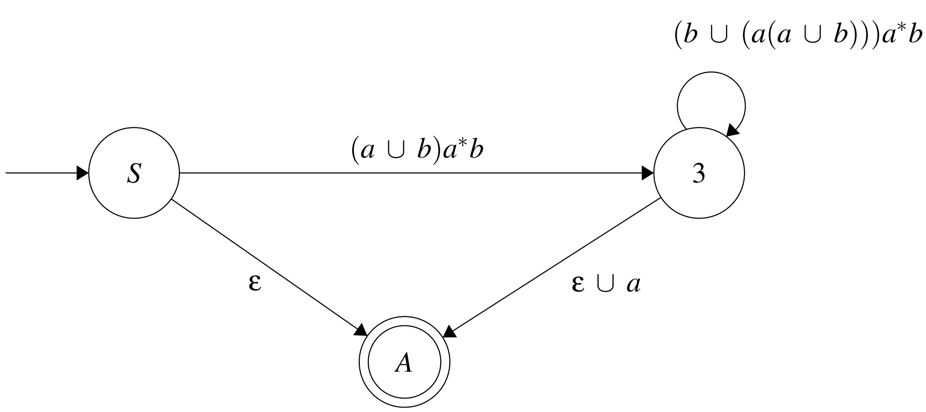
By converting to a GNFA, we get the following structure:



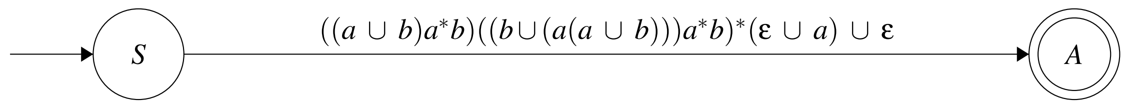

After removing state 1, we get a structure that looks like the following:




After removing state 2, we get a structure that looks like the following:



Finally, after removing state 3, we get the finite automaton below:




So the regular expression that generates the language can be written as

$$((a \cup b)a^*b)((b \cup (a(a \cup b)))a^*b)^*(\varepsilon \cup a) \cup \varepsilon$$

3. Let $C_n = \{x|x$ is a binary number that is a multiple of $n\}$. Show that for each $n \geq 1$, the language $C_n$ is regular.

   *Proof.* To prove that $C_n$ is a regular language, we need to show that a DFA can be constructed that accepts $C_n$. Because each time the automaton reads a digit, the previous string is shifted left by one index. Because we are dealing with binary numbers, whenever a 0 is appended to the existing string, the new value is $n$ times the old value $(\mod 3)$ and whenever a 1 is appended, the new value is 2 times the old value plus 1 $(modn)$.

   Given this behavior, we can describe the function applied to the existing states in the definition of a finite automata to be:

$$\delta(q_i, 0) = q_{((2i) modn)},$$
$$\delta(q_i, 1) = q_{((2i+1) modn)}$$

   Also, we know that for any $n \geq 0$, the finite automaton will have $n$ different states, where the states $q_1, q_2, \ldots, q_{n-1}$ are the remainder states and the last state is the same as the first state, or $q_0 = q_n$ and it is the start and the only accept state. In this way, we can see that a deterministic finite automata can be generated for any $C_n$ where $n \geq 0$ and thus $C_n$ is regular. □

4. Let $\Sigma = \{0, 1\}$ and let

   $D = \{w|w$ contains an equal number of occurrences of the substrings 01 and 10$\}$.

   Thus $101 \in D$ because 101 contains a single 01 and a single 10, but $1010 \notin D$ because 1010 contains two 10s and one 01. Show that $D$ is a regular language.

   To show that $D$ is a regular language we simply need to construct a DFA, NFA, or regular expression that generates all strings in the language $D$. Below is the DFA that accepts the language $D$: