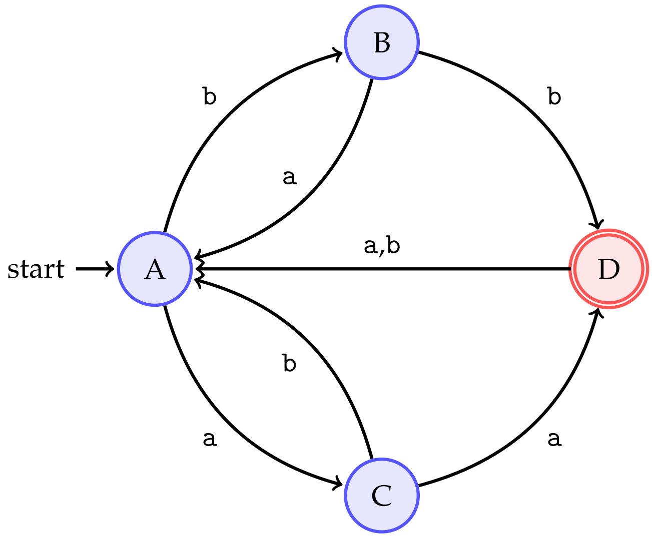
(1) Follow the JFLAP tutorial at http://www.jflap.org/tutorial/fa/createfa/fa.html. Then use JFLAP to draw and simulate some of the DFAs/NFAs discussed in the lecture.

(2) Consider the following DFA:



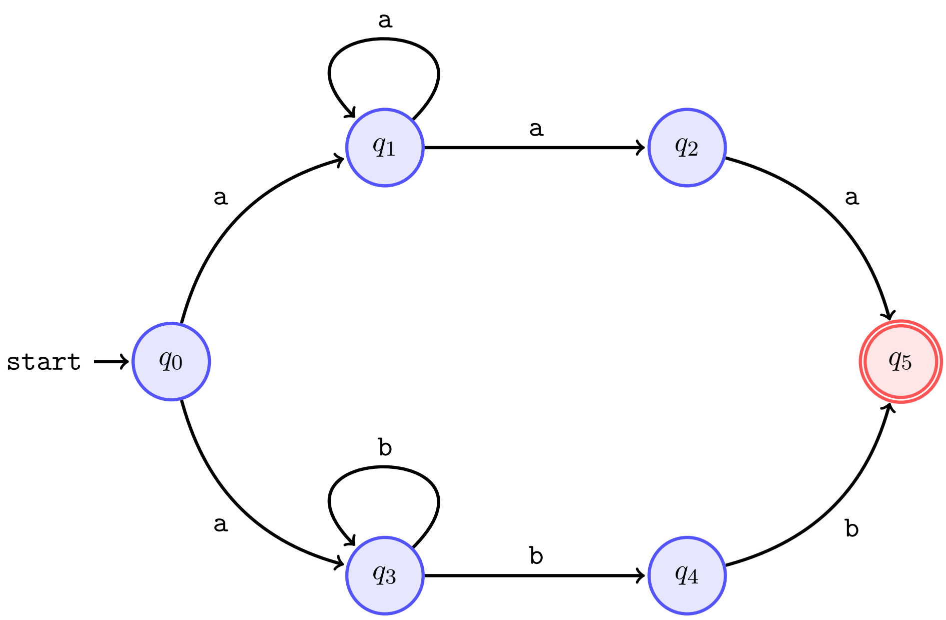
Practice *simulating* the behaviour of the above DFA using the following strings

`abba` `babb` `aaa` `bbabba`

For each string, list the sequence of states visited by the DFA (e.g. $q_0, q_2, q_0, q_1, q_3, \ldots$).

Produce the formal definition of the above DFA. This should consist of: the alphabet Σ, the set of states Q, the transition function δ, in table form, the start state, and the set of final states F.

(3) Consider the following NFA:



Practice *simulating* the behaviour of the NFA using the following strings.

`abbaa` `babb` `aaaba` `abbbbbbbaab`

For each string, list the sets of states visited by the NFA (e.g. $\{q_0\}, \{q_1, q_2\}, \{q_2, q_3\}, \ldots$).

Produce the formal definition $(\Sigma, Q, \delta, q_{\textbf{start}}, F)$ of the above NFA.

(4) The formal description $(Q, \Sigma, \delta, q_{\text{start}}, F)$ of a DFA is given by

$$(\{q_1, q_2, q_3, q_4, q_5\}, \{\mathtt{u}, \mathtt{d}\}, \delta, q_1, \{q_3\}),$$

where δ is given by the following table

	u	d
$\to q_1$	q_1	q_2
q_2	q_1	q_3
$*q_3$	q_2	q_4
q_4	q_3	q_5
q_5	q_4	q_5

Give the *state diagram* of this machine.

(5) Use `JFLAP` to design simple DFAs which recognize the following languages over $\Sigma = \{a, b\}$

a) The language of strings which begin with a.

b) The language of strings which end with b.

c) The language of strings which either begin **or** end with b.

d) The language of strings which begin with a **and** end with b.

e) The language of strings which contain the substring ba.

f) The language of strings with all the a's on the left and b's on the right

g) The language strings consisting of alternating a's and b's.

(6) Use `JFLAP` to produce NFAs to recognize the following languages over $\Sigma = \{\mathtt{0}, \mathtt{1}\}$

a) The language of strings which begin and end with `01`.

b) The language of strings which do not end with `01`.

c) The language of strings which begin and end with different symbols.

d) The language of strings of odd length.

e) The language of strings which contain an even number of `0`'s.

f) The language of binary numbers which are divisible by 4.

(7) If a is a *symbol* from an alphabet Σ then $\mathtt{a}^n$ denotes the string which consists of n successive copies of a.

Similarly, if x is a *string* of symbols then x^n denotes the string which consists of n successive copies of x. For example, $\mathtt{a}^2 = \mathtt{aa}$ and $(\mathtt{ab})^2 = \mathtt{abab}$.

Let $\Sigma = \{\mathtt{0}, \mathtt{1}\}$. Write $\mathtt{0}^4, \mathtt{1}^4, (\mathtt{10})^3, \mathtt{10}^3$ explicitly as strings in the usual form.

(8) If Σ is an alphabet then Σ^n denotes the set of all strings over Σ which have length exactly n symbols.

a) Let $\Sigma = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$. Find Σ^2.

b) Let $\Sigma = \{\mathtt{a}, \mathtt{b}\}$. Find Σ^3.

(9) If Σ is an alphabet then the set of all finite-length strings over it is denoted by Σ^*.

Let $\Sigma_1 = \{\mathtt{a}\}$ and $\Sigma_2 = \{\mathtt{a}, \mathtt{b}\}$. List the strings of length 0, 1, 2, 3, and 4 over these two alphabets. Write these in the form $\Sigma_1^* = \{\ldots\}$ and $\Sigma_2^* = \{\ldots\}$.

Note that

$$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \Sigma^4 \cup \cdots.$$