Languages and Computation (COMP 2049) Lab 02

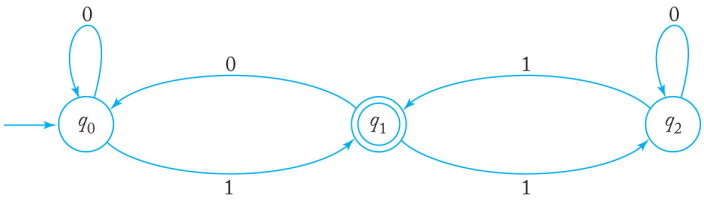
Grammars and Finite Automata

(1) Assume that we are designing a programming language, and the floating-point numbers in the language must be formed according to the following rules:

- Each number may be signed or unsigned.
  - unsigned as in `3.14`, signed as in `+3.14` or `-3.14`;
- The numerical part (also called the value field) must start with a non-empty sequence of digits.
  - For instance, in the number `+322.432`, the value field is `322.432`, which starts with the sequence of digits `322`.
- The value field may optionally include a decimal point '.', in which case it must be followed by some other digits;
  - `3` and `3.14` are acceptable, but `3.` is not acceptable.
- There may be an optional exponent field, in which case, it must contain the letter 'e', followed by a (signed or unsigned) integer.
  - For instance, `3.14e+38` or `-1.2e24` are acceptable, but `7.17e` and `7.17e-` are not acceptable.

Design a grammar that generates these numbers.

(2) Give a description of the language accepted by the following deterministic finite automaton (DFA):



(3) Assume that $\Sigma = \{a, b\}$. Show that the following language is regular by drawing a DFA that accepts it:

$$L = \{w \in \Sigma^* \mid w \text{ does not contain the substring } ab\}.$$