Example: Moore Machine$_{JP}$

Define a Moore machine that facilitates counting the number of times the substring **aab** occurs in an input string over Σ = {a, b}. That is, the machine should output **1** if it has just read two **a** symbols followed by a **b**; otherwise it should output **0**.

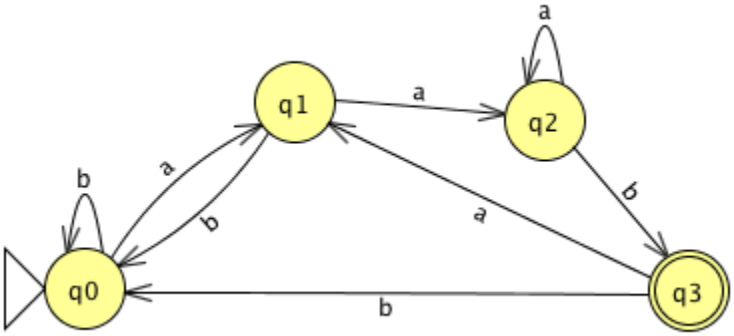
Recall that an Moore is defined as a 5-tuple (Q, Σ, δ, q0, F) where

- Q is a finite set of states, each of which specifies an output symbol
- Σ is a finite alphabet of symbols for forming the input string
- Γ is the finite set of symbols in the output alphabet
- δ is the transition function, δ: Q × Σ → Q
- q0 is the start state (q0 □ Q)

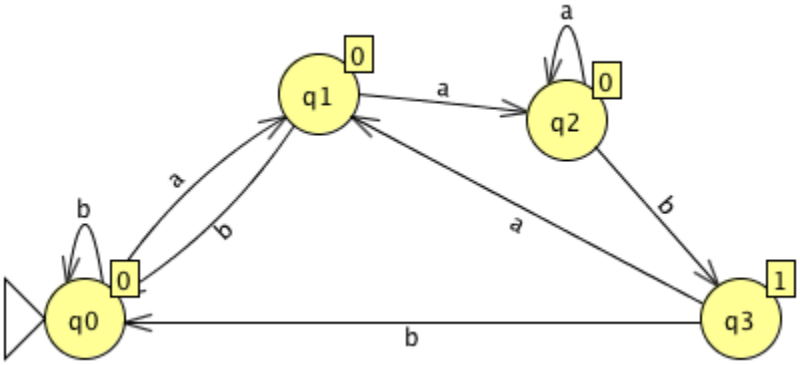
Sample Solution

One approach is to consider a DFA that recognizes all strings that end with the specified substring. Then associate all states with output **0** except that machine's accept state which should be associated with the output **1**.

Here is a DFA that recognizes the set of all strings {a, b}* that end with **aab** (see `DFA_aab.jff`).



Here is the Moore machine, based on that DFA, that prints a **0** except for state q3 which prints a **1**. (see `MOORE_aab.jff`).



Now step through input strings and observe the output. For example, the input string **aaababaabb** produces output **00001000010** indicating that there are exactly two occurrences of substring aab in the input.