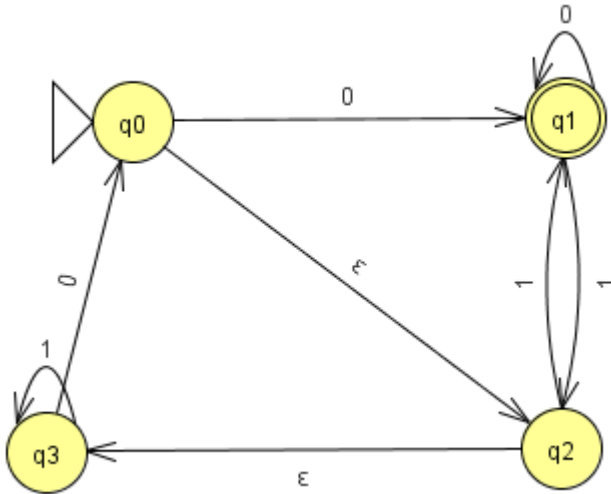


## NFA to DFA conversion

Convert the following NFA to an equivalent DFA

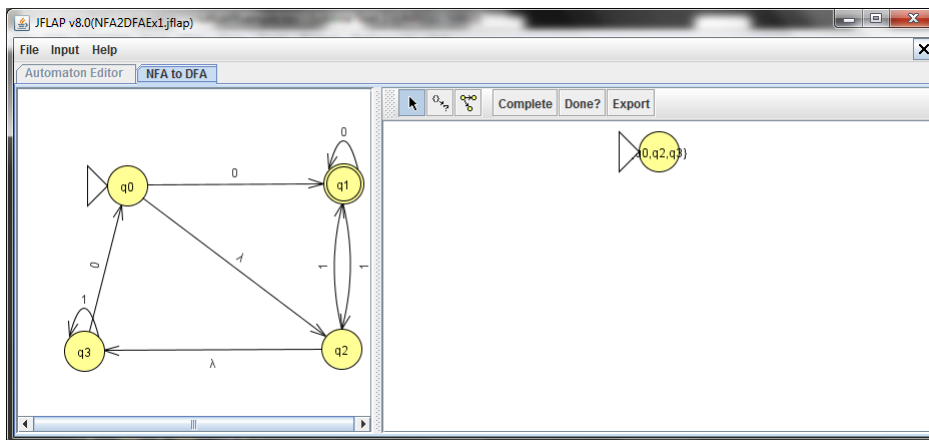


### Solution

Remember that every state of the equivalent DFA is a subset of states from the NFA. For the complete algorithm refer to Sipser.

- The first step is to figure out what the start state of the equivalent DFA is. The start state of the NFA is  $q_0$ . Use the  $\epsilon$  transitions to determine the states that can be reached from  $q_0$  using only  $\epsilon$  transitions. Here you see that  $q_2$  can be reached via an  $\epsilon$  transition from  $q_0$  and from  $q_2$  you can get to  $q_3$  with an  $\epsilon$  transition. Therefore, the start state for the DFA is the subset  $\{q_0, q_2, q_3\}$ .

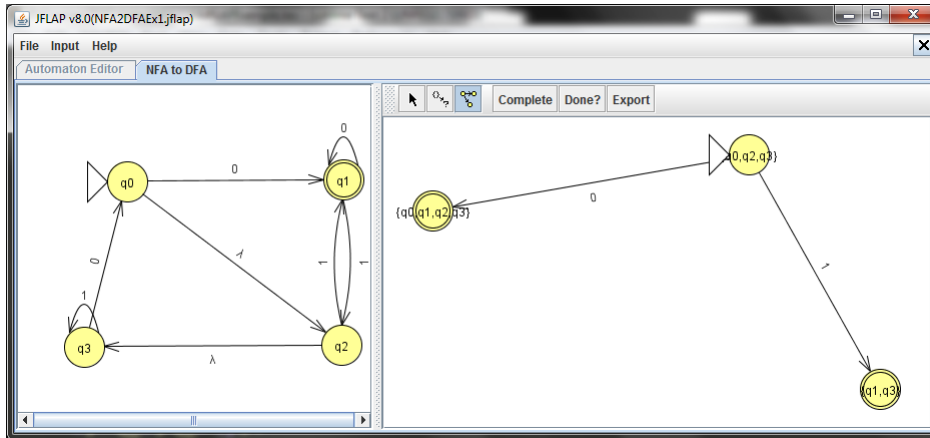
In JFLAP click convert to DFA under the Convert menu. This will immediately create the start state of the DFA.



- The next step would be to see what subset of states this state would transition to upon a 0. In the NFA,  $q_0$  goes to  $q_1$ .  $q_2$  goes nowhere on a 0. And finally  $q_3$  goes to  $q_0$  on a

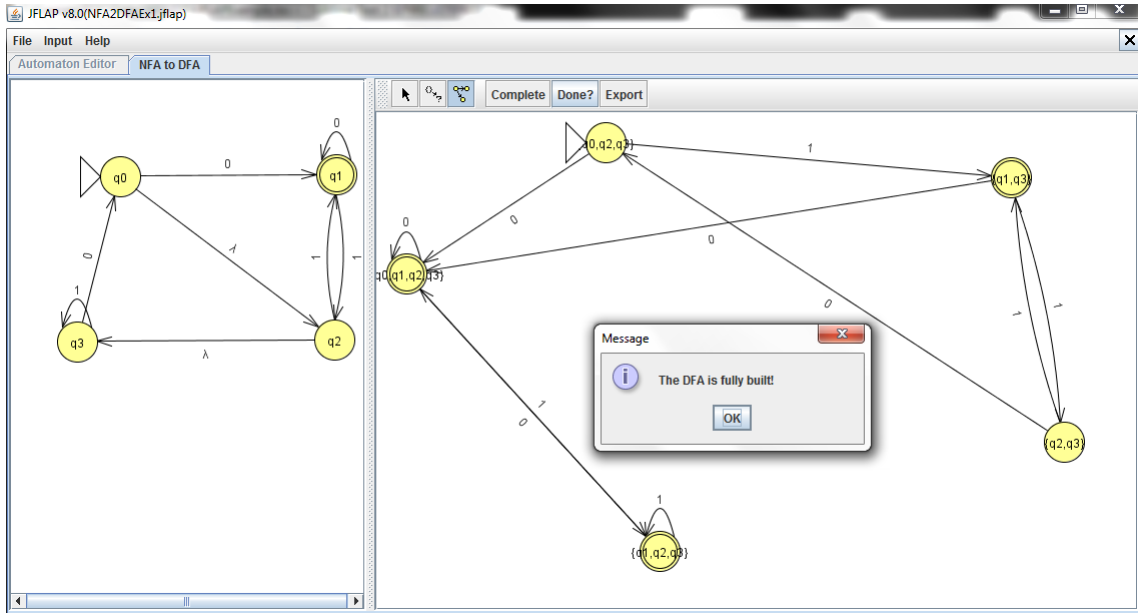
0. So currently we have the subset of states  $\{q_0, q_1\}$ . But due to  $\epsilon$  transitions the state will be  $\{q_0, q_1, q_2, q_3\}$ .

In JFLAP click the state expander button, then click the state that you want to expand. Doing this will create 2 transitions, one for 0 and one for 1.



- Here's a quick explanation of why the transition from  $\{q_0, q_2, q_3\}$  on a 1 is  $\{q_1, q_3\}$ . In the NFA  $q_0$  goes nowhere on a 1,  $q_2$  goes to  $q_1$  on a 1. Finally  $q_3$  stays at  $q_3$  on a 1. Note that neither  $q_1$  nor  $q_3$  have an  $\epsilon$  transitions coming out of them. So the subset is  $\{q_1, q_3\}$ .
- Repeat the process of clicking the state expander button and then clicking the states. Do this for each of the states.
- At some point you will have clicked the state expander button for each of the states present in your current DFA and no new states will be created (Remember that there are only a finite number of subsets of the states of the NFA).

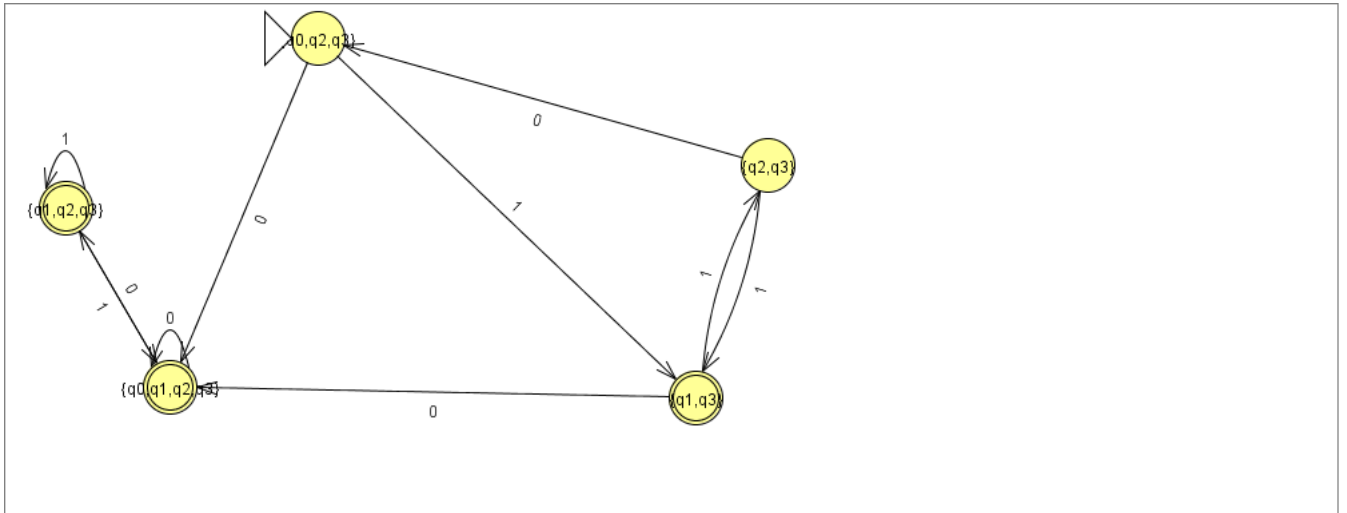
At this point you will be done in JFLAP. To confirm this, click the Done button. You should observe the following figure



- The last part would be to determine the final states. Note that this will already be done in JFLAP for you. The logic is that if a state in the DFA contains any of the final states of the NFA, then it will be a final state of the DFA. For instance the state represented by the subset  $\{q_1, q_3\}$  is a final state in the DFA because  $q_1$  is a final state in the original NFA.

Therefore among the states in our DFA, the ones that are final states are  $\{q_1, q_3\}$ ,  $\{q_0, q_1, q_2, q_3\}$  and  $\{q_1, q_2, q_3\}$ .

The final solution is



To check the correctness of this, you can use JFLAP's Compare Equivalence option which can be found under the Test menu when you are looking at any finite automaton.