Challenge Activity 1 – DFAs, NFAs, and ε-NFAs

In order to diminish the size and to speed-up the processing of ε-NFAs (especially the ones obtained by the scheme to convert regular expressions to ε-NFAs), a team intends to remove ε transitions from the ε-NFAs.

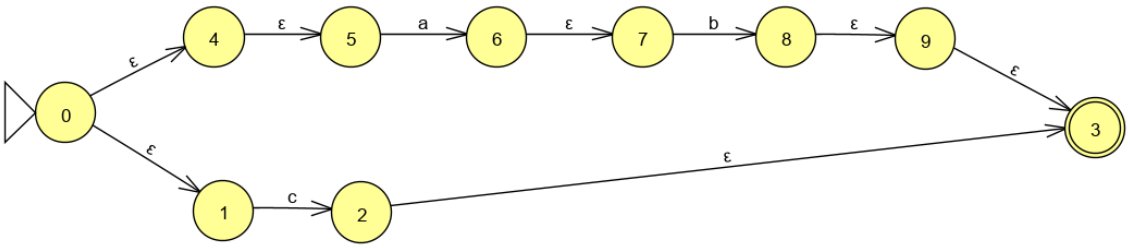
The team suggested the following two-step (A and B, and with first application of A and then of B) scheme to remove the ε-transitions:

A. *While there are modifications in the FA:*
- *For each pair of states (qi, qj) with a single transition, an ε-transition, from qi to qj, only a single input transition to qi and a single output transition from qj, merge the two states in one state qi-qj (the transition from qj becomes transition from qi-qj; and the transition to qi becomes transition to qi-qj).*

B. *While there are modifications in the FA:*
- *for each δ(qi, ε) = {qj} and qi ≠ qj, merge qi-qj (qi becomes qi-qj and the transitions from qi and from qj become transitions from qi-qj; and the transitions to qi and qj become transitions to qi-qj, then qj and its transitions are removed from the FA in the case of the only transition from qi and qj is the ε-transition).*

For both steps A and B, if qi is the initial state then qi-qj becomes the initial state. qi-qj becomes a final state if qi or qj are final states.

2. Consider the following input ε-NFA (obtained from the regular expression c+ab):



a) Apply the A-step of the scheme to the ε-NFA and show the resultant FA. Is the new FA equivalent to the ε-NFA?

Correction: After the A-step the following states are joined: 4-5, 6-7 and 8-9. **YES**, the resultant FA is equivalent to the ε-NFA.

δ(0, ε) = {4-5, 1}

δ(4-5, a) = {6-7}

δ(6-7, b) = {8-9}

δ(8-9, ε) = {3}

δ(1, c) = {2}

δ(2, ε) = {3}

b) Apply the B-step of the scheme to the FA resultant from the A-step and show the new FA. Is the new FA equivalent to the original ε-NFA?

Correction: After the B-step the following states are joined and become a new state: 0-1-4-5, 6-7 and 2-3-8-9; where state 2-3-8-9 is a final state. **TRUE**, the resultant FA is still equivalent to the ε-NFA.

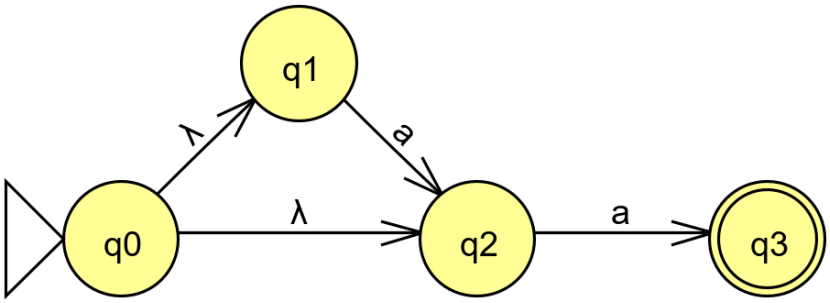
δ(0-1-4-5, a) = {6-7}

δ(0-1-4-5, c) = {2-3-8-9}

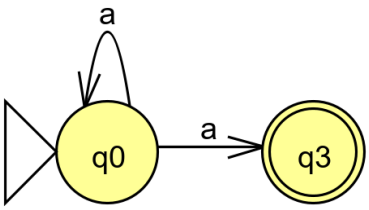
δ(6-7, b) = {2-3-8-9}

c) Is proposed scheme valid for any input ε-NFA?

Correction: FALSE. This scheme will not work for all ε-NFAs. One example is the the following ε-NFA:



Which results in the following FA:



3. The team, instead of proving the validity of the transformation for any ε-NFA, decided, each time the transformation is used, to verify the equivalence between the two FAs (the input and the output ones) by using a program specially developed for that. Suggest the main steps involved in the verification process of that program.

Correction: The verification can be divided in three major steps: NFA-to-DFA, minimization and comparison. In the first step both FAs must be converted into DFAs. This can be achieved by using the "construction of subsets" technique. Then, a minimization is performed by constructing a distinguishable states table to both DFAs at the same time. If, at the end of the process, the two initial sates are equivalent, then both FAs are equivalent.

Another possible way is to build a third DFA from the resulting DFAs of the first step by using the following formula: L3=(L1-L2) U (L2-L1). These operations can be easily performed by means of the and then apply the final state selection based on the operation. If the resulting language is empty (i.e. no acceptance state exists/is reachable) then the FAs are equivalent.