NFA with ∈ moves is exactly same as NFA without ∈ moves.

But differece exist in the transition function δ. δ must include information about ∈ transitions.

> *NFA with ∈-Moves has 6 tuples (Q, Σ, δ, q0, F).*
>
> *Where,*
>
> - *Q = finite set of states.*
> - *Σ = finite set input symbols.*
> - *δ = transition function that maps $Q \times (\Sigma \cup \{\in\})$ to 2^Q.*
> - *q0 = initial state.*
> - *F = set of final states.*

The non-deterministic finite automaton can be extended to include the transitions on null/empty input ∈.

For example,

NFA with ∈

In this NFA with epsilon,

- It accept an input string ‘aabc’.

- Or string as number of a’s followed by number of b’s followed by number of c’s.
- The string ‘aabc’ is accepted by the NFA by following the path with labels a, a, ∈, b, ∈, c.

Transition table for above NFA.

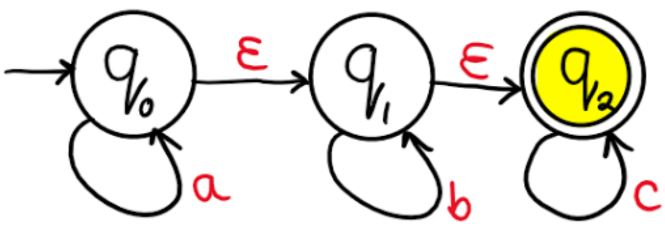
State	a
→ q0	{q0}
q1	Φ
q2	φ

∈-closure

∈-closure of a state q is a set of states following by all transitions of q that are labeled as ∈.

- ∈-closure (q0) = (q0, q1, q2)
- ∈-closure (q1) = (q1, q2)
- ∈-closure (q2) = (q2)

NFA with ∈ to NFA without ∈



NFA with ∈

Transition diagram

Transition table NFA with ∈

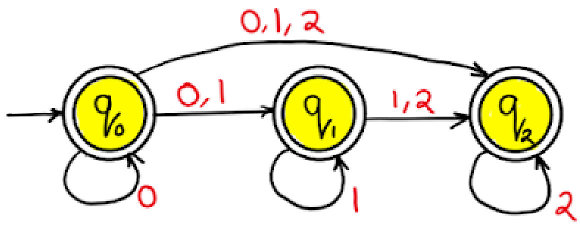
State	a	
→ q0	{q0}	
q1	Φ	[illegible]
q2	φ	

First find out ∈ closure:∈-closure

- (q0) = (q0, q1, q2)
- ∈-closure (q1) = (q1, q2)
- ∈-closure (q2) = (q2)

Transition table NFA without ∈

State	a	b	c
→q0	{q0, q1, q2}	{q1, q2}	{q2}
q1	Φ	{q1, q2}	{q2}
q2	ϕ	ϕ	{q2}



NFA without ∈

Transition diagram

Reference:

- Introduction to the Theory of Computation” by Michael Sipser.

Related posts:

1. Definition of Deterministic Finite Automata
2. Notations for DFA
3. How do a DFA Process Strings?
4. DFA solved examples
5. Definition Non Deterministic Finite Automata
6. Moore machine