Thompson's construction

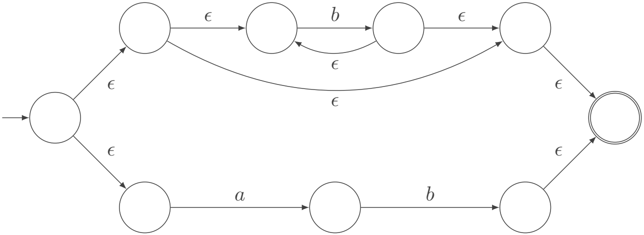
ian.mcloughlin@gmit.ie

Thompson's construction

Algorithm to construct an Non-deterministic Finite Automaton (NFA) from a regular expression.

NFA will recognise the same language as the regular expression.

Example: $a.b|b^*$



Fragments

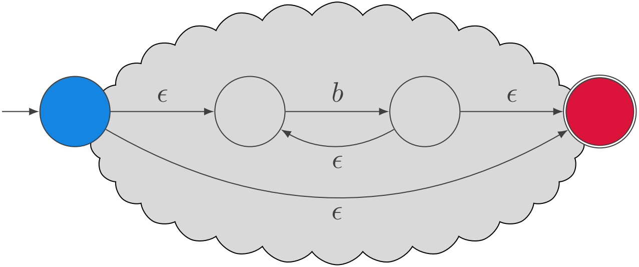
Assume the regular expression is in postfix.

Stack of fragments of the overall NFA.

Normal characters push to the stack.

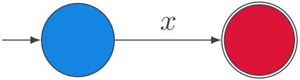
Special characters pop from and push to the stack.

Example fragment

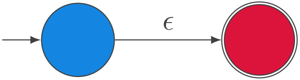


Non-special characters

For a normal, non-special character x push the following fragment to the stack.

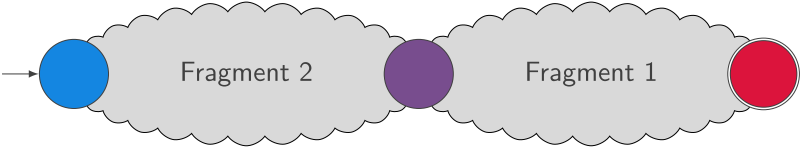


We should include the empty regular expression ϵ too.



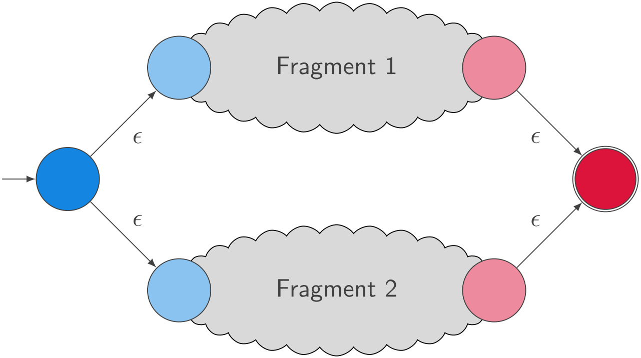
Concatenation $N.M$

When you see a ., pop two fragments from the stack and push the following instead.



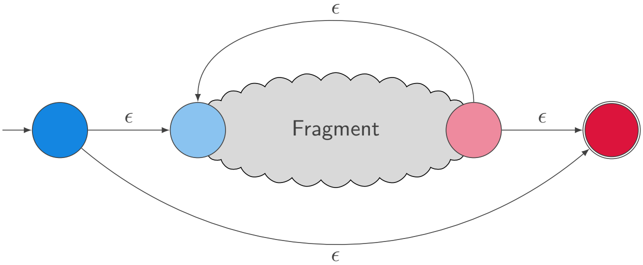
Union $N|M$

When you see a $|$, pop two fragments from the stack and push the following instead.



Kleene star N^*

When you see a *, pop a fragment from the stack and push the following instead.



Data structures

Recall the definition of an NFA.

Q is a finite set of *states*,
Σ is a finite set called the *alphabet*,
δ is the *transition function* ($Q \times \Sigma_\epsilon \to \mathcal{P}(Q)$),
q_0 is the *start state* ($\in Q$), and
F is the set of *accept states* ($\subseteq Q$).

Notes

- Only need to know δ, q_0 and F, and $|F| = 1$.
- Nothing points at q_0 and q_f points at nothing.
- From every state is a single symbol arrow, or one or two ϵ arrows.