Software Specifications
Convert Regex to State Diagram

Cain Susko

February 1, 2022

Queen's University
School of Computing

Regex to ϵ–NFA

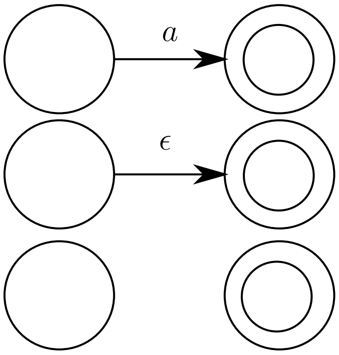
to do this we need 2 algorithms different algorithms

- convert Regex to a State Diagram
- convert State Diagram to Regex

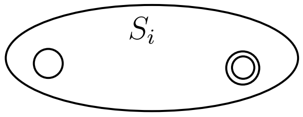
The first algorithm will produce a NFA with ϵ-transitions as well as the following properties:

1. there is exactly one final state and it is not the start state
2. the start state has no incoming transitions
3. the final state has no outgoing transitions

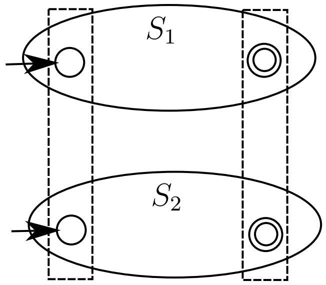
these conditions are needed to guarantee the correctness of the recursive steps of the algorithm
said algorithm is as follows:
The base cases for the algorithm (as its recursive) are $a \in \sum$, *empty string*, and $\emptyset$



Thus, the inductive step is:
suppose that a state diagram S_i has been constructed for regex R_i where $i = 1, 2$

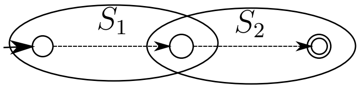


Now, for $R_1 + R_2$:



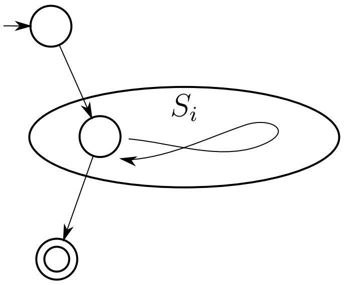
We must merge the start and end states.
note that conditions 1 and 2 (see list above) guarantee that any computation uses only transitions from S_1 or only transitions from S_2

For $R_1 \cdot R_2$:



We must merge the final state of S_1 with the starting state of S_2. This merged state is not final. Conditions 2 and 3 guarantee that any accepting path consists of a string by S_1 concatenated with a string accepted by S_2.

For $R_i *$



We must merge the start and end states for R_i
furthermore the merged state is **not final**, and it cannot be the start state. We must then add a start and end state (with ϵ-transition) to complete the diagram.
we use this algorithm recursively, starting from the smallest part of a Regex, like a single number R_i, and build our way to a more complex diagram.