

Introduction

Convert simple regular expressions to deterministic finite automaton. (Regex => NFA => DFA)

Supported grammars

- $r = (s)$
 - $r = st$
 - $r = s[t]$
 - $r = s^*$
 - $r = s^+$
 - $r = s?$
 - $r = \epsilon$
- (Copy this character to input if needed)

Examples

- $(a|b)^*$
- $(a^*|b^*)^*$
- $((c|a)b^+)^*$
- $(a|b)^*abb(a|b)^*$

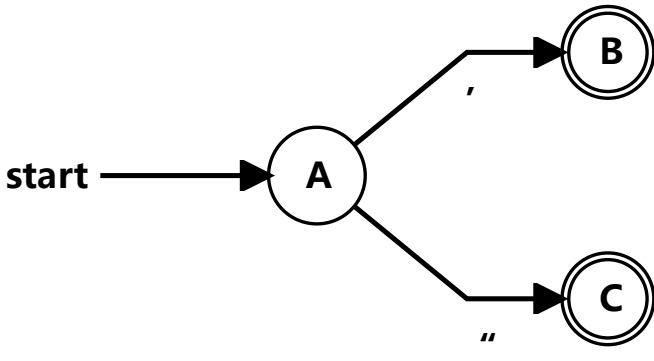
Input:

"|"

CONVERT

NFA: <https://cyberzhg.github.io/toolbox/regex2nfa?regex=4oCcOKAmQ==> (<https://cyberzhg.github.io/toolbox/regex2nfa?regex=4oCcOKAmQ==>)
Min-DFA: https://cyberzhg.github.io/toolbox/min_dfa?regex=4oCcOKAmQ== (https://cyberzhg.github.io/toolbox/min_dfa?regex=4oCcOKAmQ==)

NFA STATE	DFA STATE	TYPE	'	"
{0,1,3}	A		B	C
{4,5}	B	accept		
{2,5}	C	accept		



URL: <https://cyberzhg.github.io/toolbox/nfa2dfa?regex=4oCcOKAmQ==>