

نمونه ورودی پروژه اول

ورودی اولیه برنامه:

```
{q0,q1,q2,q3,q4,q5,q6}  
{a,b}  
9  
q0,q1,a  
q1,q1,b  
q1,q2,  
q2,q3,a  
q3,q2,a  
q3,q4,b  
q2,q5,b  
q5,q6,a  
q6,q1,b  
{q1,q3,q6}
```

تست توابع مربوط به NFA:

IsAcceptByNFA('') → False

IsAcceptByNFA('abb') → True

IsAcceptByNFA('abaa') → False

IsAcceptByNFA('abab') → True

IsAcceptByNFA('aaaaaa') → True

FindRegExp() → $a(b^*(bab)^*)^* + a(b^*(bab)^*)a(aa)^*$ or any equivalence regex

CreateEquivalentDFA() → DFA object

States: $[\epsilon, [q0], [q1, q2], [q1, q2, q4], [q1, q2, q5], [q2], [q3], [q3, q6], [q4], [q5], [q6]]$

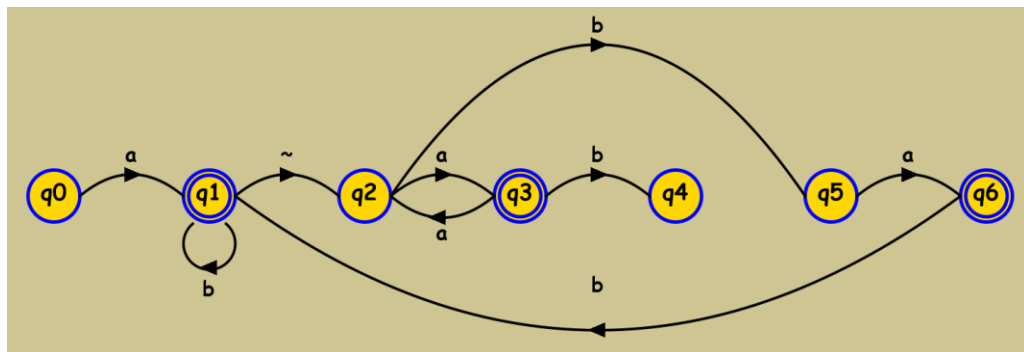
Initial State: $q0$

Final States: $[q1, q2], [q1, q2, q4], [q1, q2, q5], [q3], [q3, q6], [q6]$

Transitions: $\{ \epsilon = \{a=, b=\}, q2 = \{a=q3, b=q5\}, q1, q2 = \{a=q3, b=q1, q2, q5\}, q3 = \{a=q2, b=q4\}, q4 = \{a=, b=\}, q5 = \{a=q6, b=\}, q6 = \{a=, b=q1, q2\}, q1, q2, q5 = \{a=q3, q6, b=q1, q2, q5\}, q1, q2, q4 = \{a=q3, b=q1, q2, q5\}, q0 = \{a=q1, q2, b=\}, q3, q6 = \{a=q2, b=q1, q2, q4\} \}$

ShowSchematicNFA()

- \sim means λ .



تست توابع مربوط به DFA:

تمامی رشته‌هایی که برای NFA تست شد، برای DFA معادل هم باید نتیجه یکسان دهد.

IsAcceptByNFA('') \rightarrow False

IsAcceptByNFA('abb') \rightarrow True

IsAcceptByNFA('abaa') \rightarrow False

IsAcceptByNFA('abab') \rightarrow True

IsAcceptByNFA('aaaaaa') \rightarrow True

MakeSimpleDFA() \rightarrow DFA object

States: [[q0], [q1, q2, q5], [q2], [q3], [q3, q6], [q5], [q6], [, q4], [q1, q2, q1, q2, q4]]

Initial State: q0

Final States: [[q1, q2, q1, q2, q4], [q1, q2, q5], [q3], [q3, q6], [q6]]

Transitions: {q2={a=q3, b=q5}, q3={a=q2, b=q4}, q5={a=q6, b=q4}, q6={a=q4, b=q1,q2,q1,q2,q4}, q1,q2,q5={a=q3,q6, b=q1,q2,q5}, q1,q2,q1,q2,q4={a=q3, b=q1,q2,q5}, q0={a=q1,q2,q1,q2,q4, b=q4}, q3,q6={a=q2, b=q1,q2,q1,q2,q4}, ,q4={a=q4, b=q4} }

ShowSchematicDFA()

- So complicated generated by the program :))

