CMSC 141 Automata and Language Theory

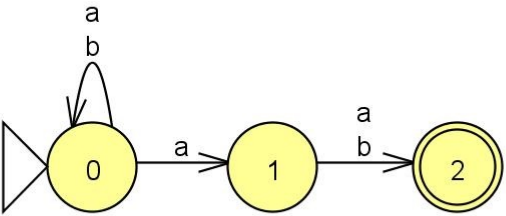
Regular Languages

Mark Froilan B. Tandoc

August 27, 2014

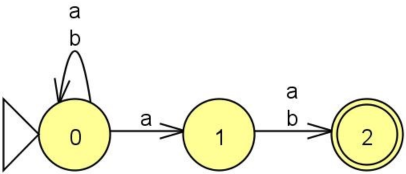
NFA $\rightarrow$ DFA Conversion

$L = \{w|w \text{ have an a as the second last symbol }\}$
$L = \{aa, ab, aaa, aab, baa, bab, aaaa, aaab, \ldots\}$

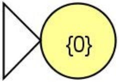


δ	a	b
0	{0,1}	{0}
1	{2}	{2}
2	$\emptyset$	$\emptyset$

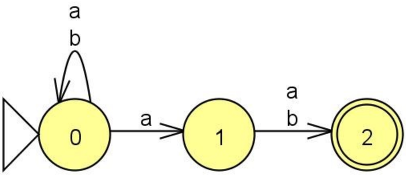
NFA DFA Conversion



δ	a	b
0	$\{0,1\}$	$\{0\}$
1	$\{2\}$	$\{2\}$
2	$\emptyset$	$\emptyset$

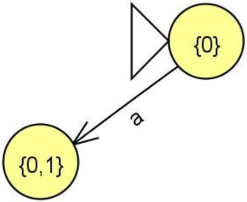


NFA DFA Conversion

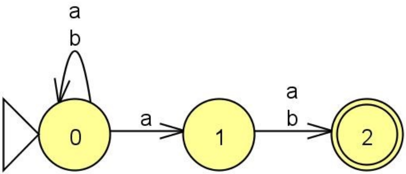


δ	a	b
0	{0,1}	{0}
1	{2}	{2}
2	$\emptyset$	$\emptyset$

$$\delta'(\{0\}, a) = \delta(0, a) \\ = \{0, 1\}$$

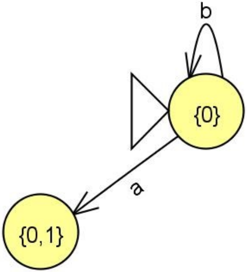


NFA DFA Conversion

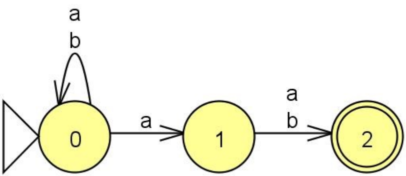


δ	a	b
0	{0,1}	{0}
1	{2}	{2}
2	$\emptyset$	$\emptyset$

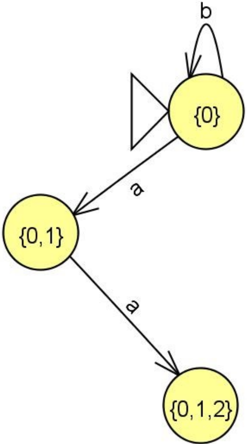
$$\begin{aligned}\delta'(\{0\}, b) &= \delta(0, b) \\ &= \{0\}\end{aligned}$$



NFA DFA Conversion

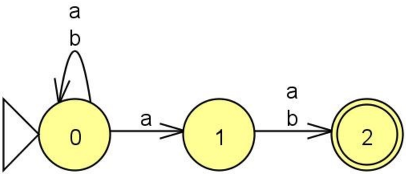


δ	a	b
0	$\{0,1\}$	$\{0\}$
1	$\{2\}$	$\{2\}$
2	$\emptyset$	$\emptyset$



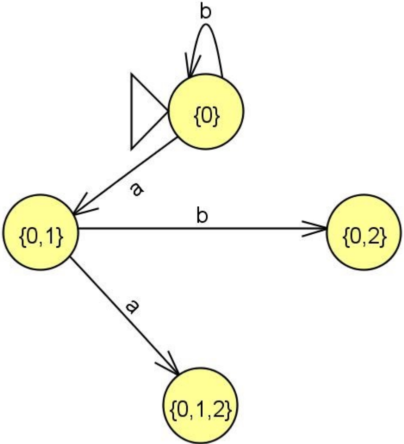
$$\begin{aligned}\delta'(\{0,1\},a) &= \delta(0,a)\cup\delta(1,a)\\ &= \{0,1\}\cup\{2\}\\ &= \{0,1,2\}\end{aligned}$$

NFA DFA Conversion

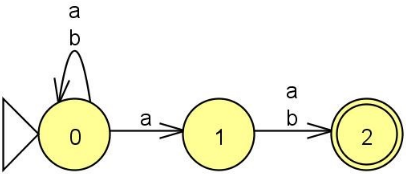


δ	a	b
0	{0,1}	{0}
1	{2}	{2}
2	$\emptyset$	$\emptyset$

$$\begin{aligned}\delta'(\{0,1\}, b) &= \delta(0, b) \cup \delta(1, b)\\ &= \{0\} \cup \{2\}\\ &= \{0, 2\}\end{aligned}$$

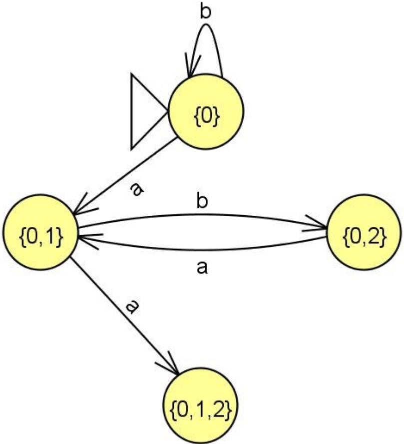


NFA DFA Conversion

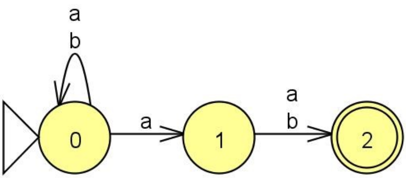


δ	a	b
0	$\{0,1\}$	$\{0\}$
1	$\{2\}$	$\{2\}$
2	$\emptyset$	$\emptyset$

$$
\begin{aligned}
\delta'(\{0,2\}, a) &= \delta(0, a) \cup \delta(2, a) \\
&= \{0,1\} \cup \emptyset \\
&= \{0,1\}
\end{aligned}
$$

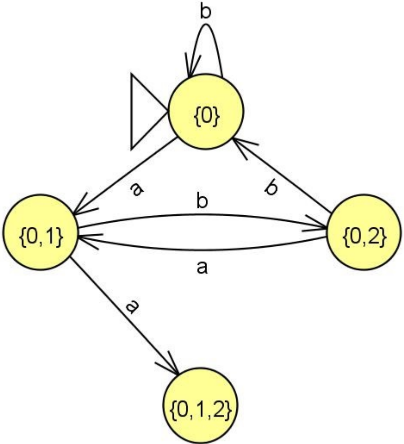


NFA DFA Conversion

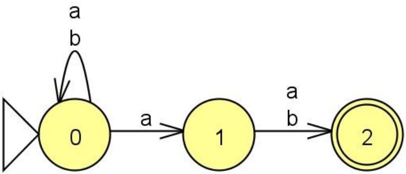


δ	a	b
0	$\{0,1\}$	$\{0\}$
1	$\{2\}$	$\{2\}$
2	$\emptyset$	$\emptyset$

$$\begin{aligned}\delta'(\{0,2\}, b) &= \delta(0, b) \cup \delta(2, b) \\ &= \{0\} \cup \emptyset \\ &= \{0\}\end{aligned}$$

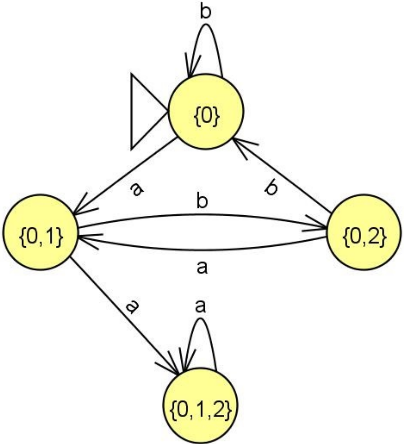


NFA DFA Conversion

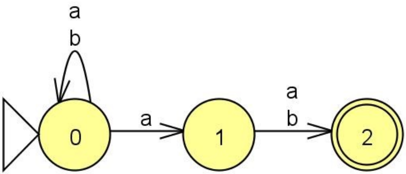


δ	a	b
0	{0,1}	{0}
1	{2}	{2}
2	$\emptyset$	$\emptyset$

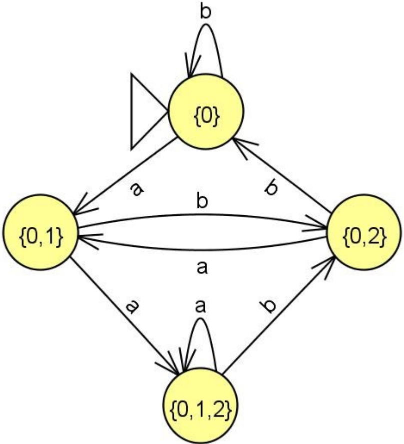
$$\begin{aligned}\delta'(\{0,1,2\},a) &= \delta(0,a)\cup\delta(1,a)\cup\delta(2,a)\\ &= \{0,1\}\cup\{2\}\cup\emptyset\\ &= \{0,1,2\}\end{aligned}$$



NFA DFA Conversion

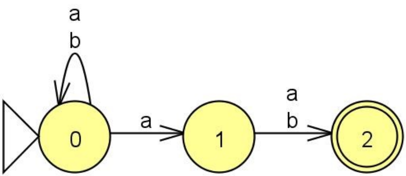


δ	a	b
0	$\{0,1\}$	$\{0\}$
1	$\{2\}$	$\{2\}$
2	$\emptyset$	$\emptyset$

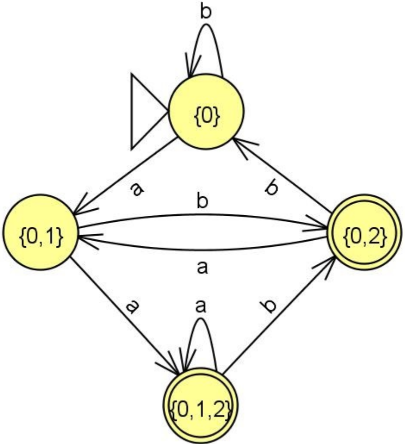


$$
\begin{aligned}
\delta'(\{0,1,2\}, b) \\
&= \delta(0,b) \cup \delta(1,b) \cup \delta(2,b) \\
&= \{0\} \cup \{2\} \cup \emptyset \\
&= \{0,2\}
\end{aligned}
$$

NFA DFA Conversion



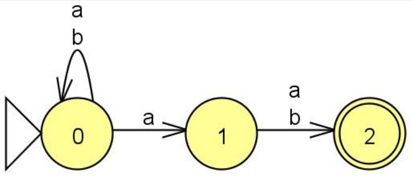
δ	a	b
0	{0,1}	{0}
1	{2}	{2}
2	$\emptyset$	$\emptyset$



Bad case for NFA $\to$ DFA conversion

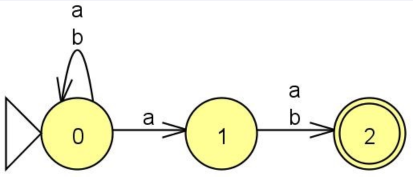
Bad case for NFA $\rightarrow$ DFA conversion

DFA

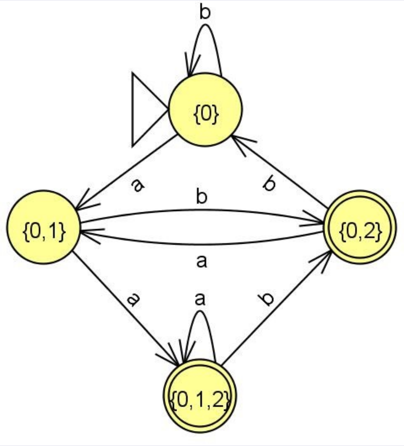


Bad case for NFA → DFA conversion

DFA

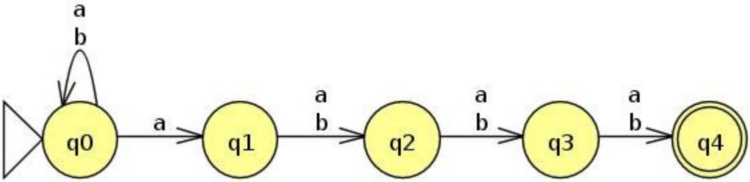


NFA

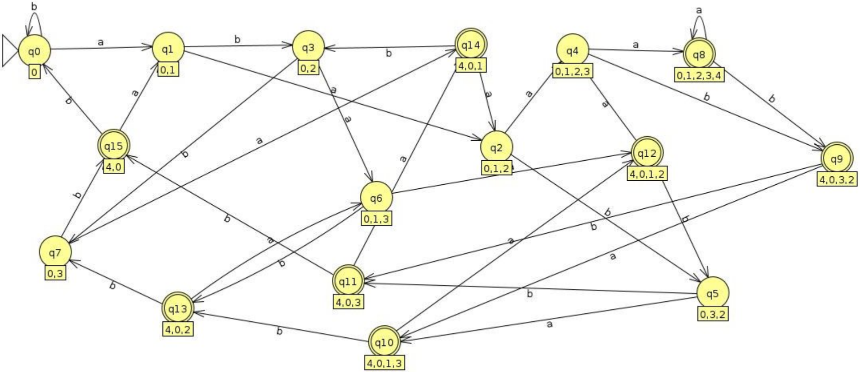


Bad case for NFA $\rightarrow$ DFA conversion

Generalizing our example for arbitrary n

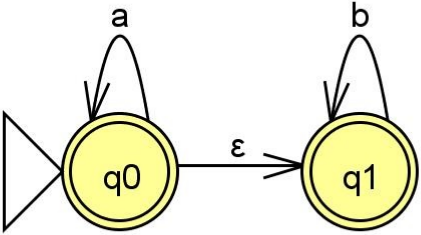


Bad case for NFA $\to$ DFA conversion



εNFAs

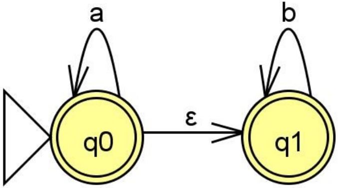
NFAs also has the capability of having transitions using ε.

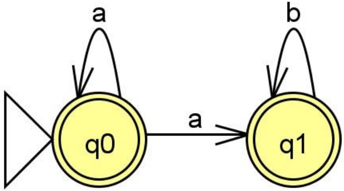


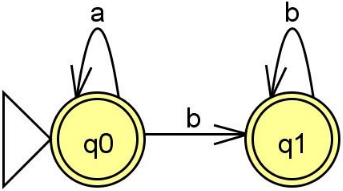
Without reading any input, the machine can split to multiple states.

εNFAs

Does these NFAs accept the same language? Why or why not?

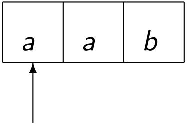


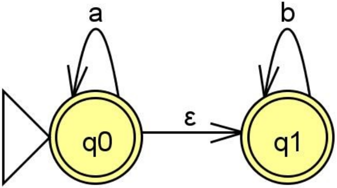


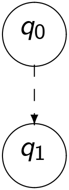


Sample Run

$L = \{w|w$ is a's followed by the b's $\}$
Test if "aab" belong to the language

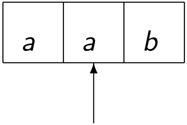


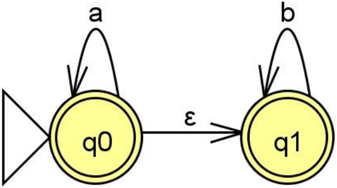


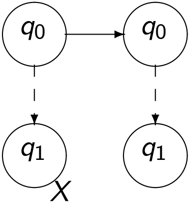


Sample Run

$L = \{w | w$ is a's followed by the b's $\}$
Test if "aab" belong to the language

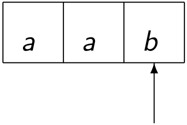


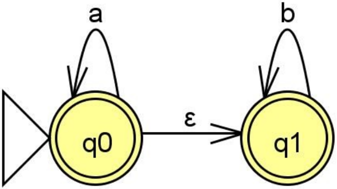


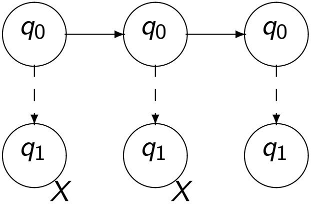


Sample Run

$L = \{w | w \text{ is a's followed by the b's } \}$
Test if "aab" belong to the language



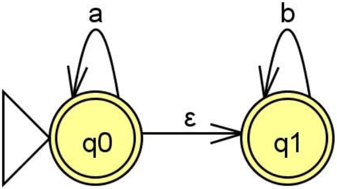


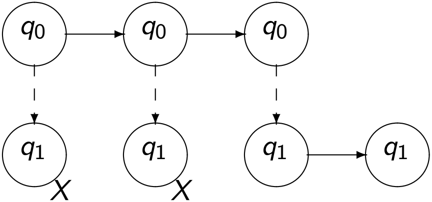


Sample Run

$L = \{w | w$ is a's followed by the b's $\}$
Test if "aab" belong to the language

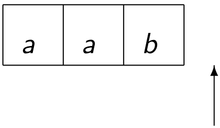
a	*a*	*b*

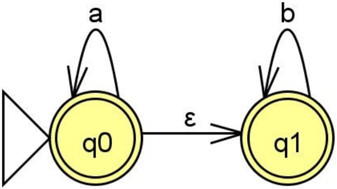


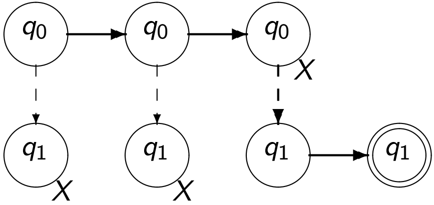


Sample Run

$L = \{w|w$ is a's followed by the b's $\}$
Test if "aab" belong to the language







References

- Previous slides on CMSC 141
- M. Sipser. Introduction to the Theory of Computation. Thomson, 2007.
- J.E. Hopcroft, R. Motwani and J.D. Ullman. Introduction to Automata Theory, Languages and Computation. 2nd ed, Addison-Wesley, 2001.
- E.A. Albacea. Automata, Formal Languages and Computations, UPLB Foundation, Inc. 2005
- JFLAP, www.jflap.org