



نظریه‌ی زبان‌ها و ماشین‌ها

علی شکیبا

دانشگاه ولی‌عصر (عج) رفسنجان

ali.shakiba@vru.ac.ir

فصل ۲: اتوماتای متناهی

پذیرنده‌های متناهی نامعین

- نامعین بودن باعث **قابلیت انتخاب حرکات اتومات**

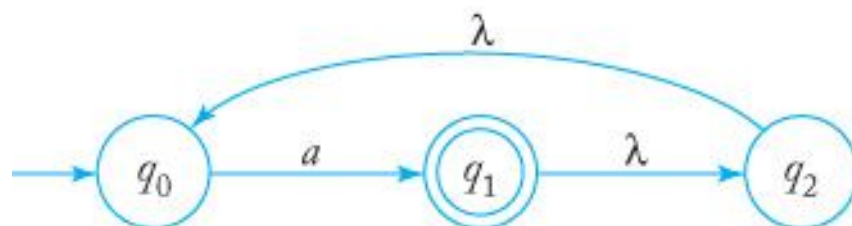
- تابع انتقال پذیرنده‌های متناهی نامعین

$$M = (Q, \Sigma, \delta, q_0, F)$$

به صورت زیر تعریف می‌شود:

$$\delta: Q \times (\Sigma \cup \{\lambda\}) \rightarrow 2^Q$$

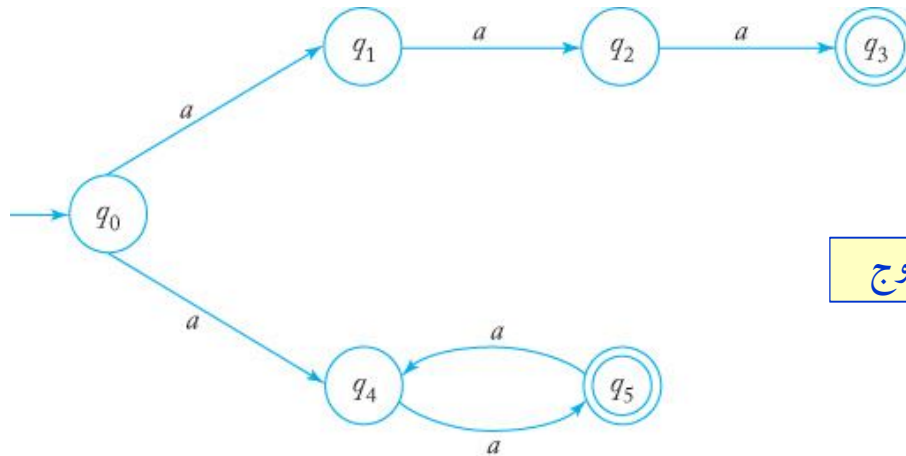
مثال از پذیرنده‌ی متناهی نامعین



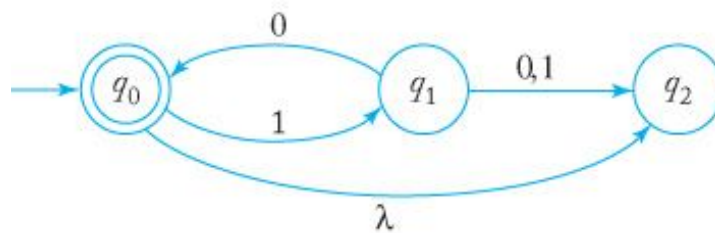
$$\delta^*(q_2, \lambda) = \{q_0, q_2\}$$

$$\delta^*(q_2, aa) = \{q_0, q_1, q_2\}$$

مثال از پذیرنده‌ی متناهی نامعین (ادامه)



پذیرش رشته‌ی aaa یا رشته‌های a با طول زوج



در مورد پذیرش رشته‌ی 01 بحث کنید؟

$$\delta(q_2, 0) = \emptyset$$

زبان پذیرفته شده‌ی یک NFA

- زبان پذیرنده‌ی متناهی نامعین

$$M = (Q, \Sigma, \delta, q_0, F)$$

- عبارت است از تمام رشته‌های w که یک قدم با برجسب w از حالت شروع به حداقل یک حالت پایانی وجود داشته باشد:

$$L(M) = \{w \in \Sigma^* : \delta^*(q_0, w) \cap F \neq \emptyset\}$$

تفاوت زبان‌های DFA و NFA

- آیا NFA ها از DFA ها قوی تر هستند؟
- به عبارت دیگر؛ آیا زبانی وجود دارد که بتوان برای آن یک NFA ترسیم کرد؛ اما هیچ DFA ای برای آن وجود نداشته باشد؟
- آیا DFA ها از NFA ها قوی تر هستند؟
- DFA ها را می‌توان حالت خاصی از NFA ها دانست؛
- بنابراین؛ هر DFA خود یک NFA است.

تفاوت زبان‌های NFA و DFA (ادامه)

- آیا زبانی وجود دارد که بتوان برای آن یک NFA ترسیم کرد؛ اما هیچ DFA ای برای آن وجود نداشته باشد؟
 - خیر! DFA و NFA از لحاظ قدرت پذیرش زبان‌ها یکسان هستند.
- به ازای هر زبانی که یک NFA آن را پذیرش کند؛ DFA ای وجود دارد که آن زبان را پذیرش می‌کند.
 - بنابراین؛ زبان‌های پذیرش شده توسط NFA ها نیز در زمره‌ی زبان‌های منظم قرار می‌گیرند.
- در ادامه؛ الگوریتمی جهت تبدیل یک NFA به DFA معادل ارائه می‌کنیم.

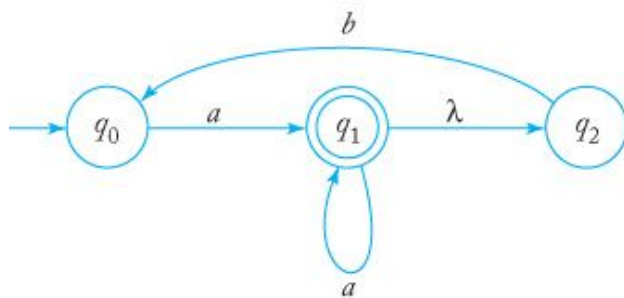
چه وقت دو اتومات هم‌ارز هستند؟

- دو اتومات M_1 و M_2 را هم‌ارز نامند هرگاه داشته باشیم:

$$L(M_1) = L(M_2).$$

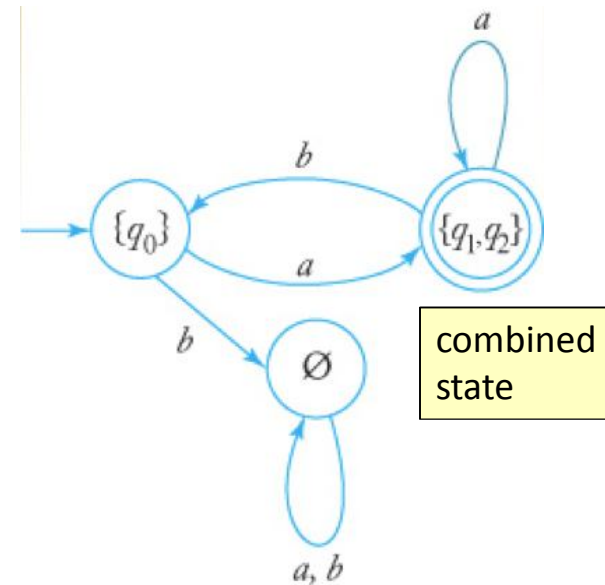
- به عبارت دیگر؛ هر دو زبان یکسانی را بپذیرند.

مثال از تبدیل یک NFA به DFA معادل



NFA	DFA
$\delta(q_0, a) = \{q_1, q_2\}$	$\delta(\{q_0\}, a) = \{q_1, q_2\}$
$\delta(q_0, b) = \emptyset$	$\delta(\{q_0\}, b) = \emptyset$
$\delta(q_1, a) = \{q_1, q_2\}$	$\delta(\{q_1, q_2\}, a) = \{q_1, q_2\}$
$\delta(q_2, b) = \{q_0\}$	$\delta(\{q_1, q_2\}, b) = \{q_0\}$

Label the states of the new DFA after the set of states that the NFA can be in after each transition.



combined state

Each state of the DFA must have a transition on each symbol. Refer back to the NFA to derive the transition.

چرا NFA را به DFA تبدیل می کنیم؟

- طراحی NFA به مراتب از DFA ساده تر است.
- اما، اجرای یک NFA از DFA کارآتر نیست.
- یک NFA ممکن است لازم باشد بیش از یک مسیر را برای بررسی پذیرش یا عدم پذیرش یک رشته؛ طی کند؛ اما ...
- در DFA به ازای هر رشته دقیقاً یک مسیر نیاز به بررسی دارد.

الگوریتم تبدیل یک NFA به DFA معادل

1. گراف انتقال G_D را با راس آغازین $\{q_0\}$ در نظر بگیرید.
2. مادامی که همه یال‌ها در نظر گرفته نشده‌اند؛ مراحل زیر را تکرار کنید:
 - (a) هر یک از رئوس $\{q_i, q_j, \dots, q_\ell\}$ را که برای نماد $a \in \Sigma$ یالی از آن خارج نشده است را در نظر بگیرید.
 - (b) مقدار $A = \delta^*(q_i, a) \cup \delta^*(q_j, a) \cup \dots \cup \delta^*(q_\ell, a)$ را محاسبه کنید. در صورتی که G_D فاقد راسی با برچسب A بود؛ آن را اضافه کنید.
 - (c) یالی با برچسب a از راس $\{q_i, q_j, \dots, q_\ell\}$ به راس با برچسب A اضافه کنید.
3. اگر برچسب راس شامل حالتی نهایی مانند $q \in F$ بود؛ آن راس نیز یک راس نهایی است.
4. اگر NFA رشته‌ی λ را بپذیرد؛ آنگاه حالت $\{q_0\}$ نیز یک حالت نهایی است.