

## MODUL VIII

# TEORI BAHASA DAN AUTOMATA

Tujuan :

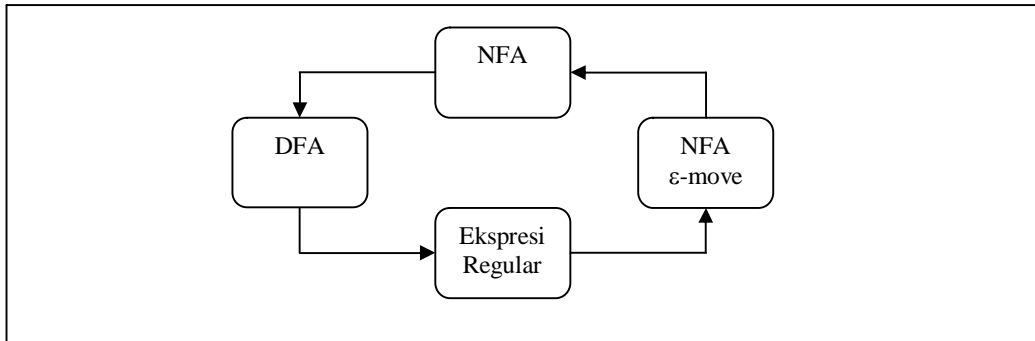
Mahasiswa memahami ekspresi reguler dan dapat menerapkannya dalam berbagai penyelesaian persoalan.

Materi :

- Hubungan antara DFA, NFA, dan ekspresi reguler
- Aturan Produksi Suatu FSA
  - Aturan Produksi Bahasa Regular
  - Mengkonstruksi Aturan Produksi dari Suatu *Finite State Automata*
  - Finite State Automata untuk Suatu Tata Bahasa Regular

## HUBUNGAN ANTARA DFA, NFA, DAN EKSRESI REGULAR

Hubungan antara *Non-deterministic Finite Automata*, *Deterministic Finite Automata*, dan ekspresi regular bisa digambarkan seperti gambar berikut ini.



Gambar 19. Hubungan antara DFA, NFA, dan ekspresi regular

Kita lihat contoh-contoh permasalahan lebih lanjut.

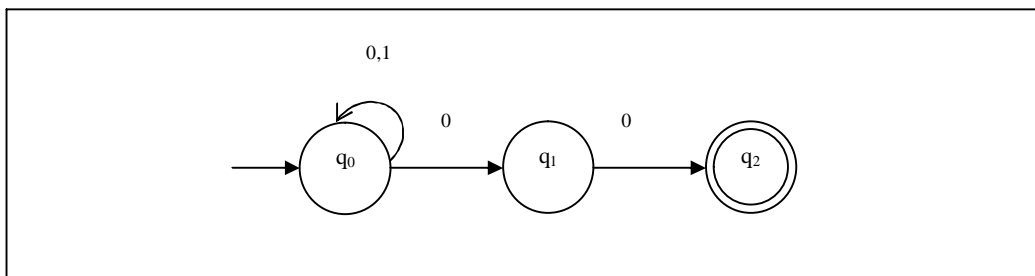
Kita ingin membuat mesin *Deterministic Finite Automata* yang menerima bahasa yang berupa semua *string* yang berakhiran dengan '00'. Diketahui  $\Sigma = \{0,1\}$

Pertama kita buat ekspresi regularnya:

$$(0+1)^*00$$

(kita bisa menuliskan juga sebagai  $(0 \cup 1)^*00$ , karena symbol '+' bisa menggantikan '∪').

Dari ekspresi regular tersebut kita lebih mudah membuat *Non-deterministic Finite Automata*, *Deterministic Finite Automata*-nya lebih dahulu, dari pada langsung *Deterministic Finite Automata*. Mesin NFA-nya dapat kita bisa lihat ada gambar 20.

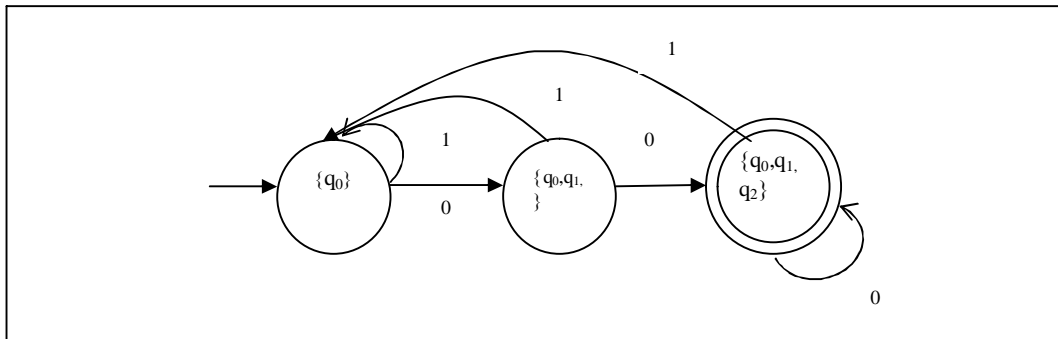


Gambar 20. Mesin NFA

Tabel transisi untuk NFA gambar 20.

$\delta$	0	1
$q_0$	$\{q_0, q_1\}$	$\{q_0\}$
$q_1$	$\{q_2\}$	-
$q_2$	-	-

Akhirnya kita buat *Deterministic Finite Automata* pada gambar 21 yang ekivalen dengan *Non-deterministic Finite Automata* tersebut. Bisa kita cek dengan untai yang harus diterima oleh mesin itu, seperti : 00, 100, 000, 0100, 0000, 1100



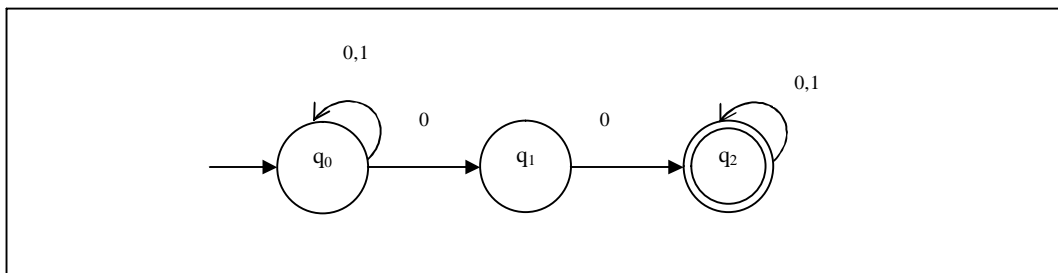
Gambar 21. Mesin DFA

Contoh berikutnya: Kita ingin membuat mesin *Deterministic Finite Automata* yang menerima bahasa berupa semua *string* yang memuat minimal dua nol berturutan ('00'). Diketahui  $\Sigma = \{0,1\}$

Perhatikan perbedaannya dengan soal sebelumnya. Disini tidak ditentukan letak '00'. Kita buat ekspresi regularnya:

$$(0+1)^*00(0+1)^*$$

*Non-deterministic Finite Automata*-nya bisa dilihat pada gambar 22.

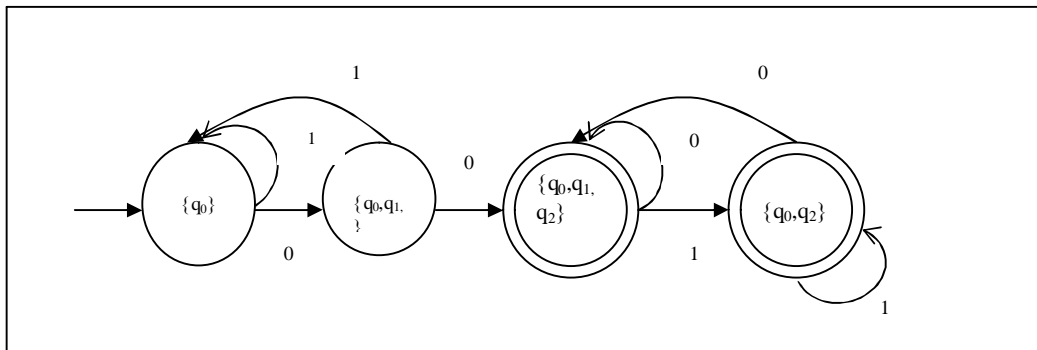


Gambar 22. Mesin NFA

Tabel transisi untuk NFA gambar 22:

$\delta$	0	1
$q_0$	$\{q_0, q_1\}$	$\{q_0\}$
$q_1$	$\{q_2\}$	-
$q_2$	$\{q_2\}$	$\{q_2\}$

Akhirnya kita buat *Deterministic Finite Automata* pada gambar 23 yang ekivalen dengan *Non-deterministic Finite Automata* tersebut. Bisa kita cek dengan untai yang harus bisa diterima oleh mesin itu, seperti: 00, 100, 001, 000, 0100, 1001, 0000, 1100, 0011, 1010011, 110011



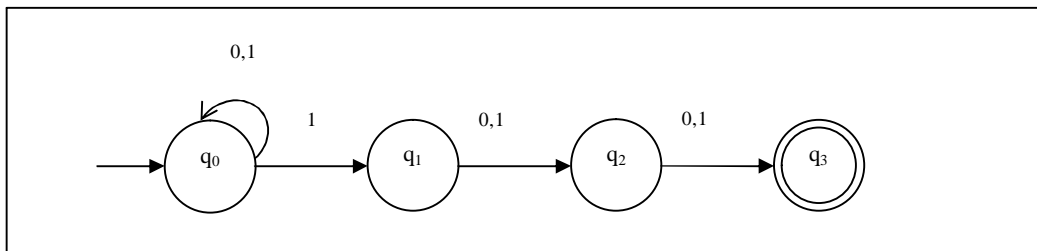
Gambar 23. Mesin DFA

Contoh berikutnya: Kita ingin membuat mesin *Deterministic Finite Automata* yang menerima bahasa berupa semua *string* dimana symbol ketiga dari kanan adalah '1'. Diketahui  $\Sigma = \{0, 1\}$

Kita buat ekspresi reglarnya:

$$(0+1)^*1(0+1)(0+1)$$

*Non-deterministic Finite Automata*-nya bisa dilihat pada gambar 24.



Gambar 24. Mesin NFA

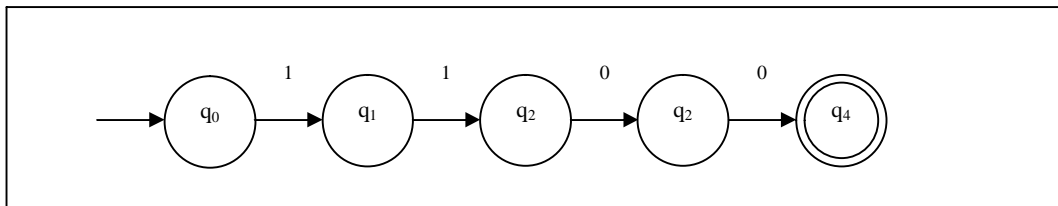
Tabel transisi untuk NFA gambar 24

$\delta$	0	1
$q_0$	$\{q_0\}$	$\{q_0, q_1\}$
$q_1$	$\{q_2\}$	$\{q_2\}$
$q_2$	$\{q_3\}$	$\{q_3\}$
$q_3$	-	-

Akhirnya kita bisa buat *Deterministic Finite Automata* yang ekuivalen dengan *Non-deterministic Finite Automata* tersebut. Gambarnya memang agak rumit karena jumlah *state* yang banyak (silahkan anda buat sendiri untuk latihan). Bisa kita cek dengan untai yang harus bisa diterima oleh mesin itu, seperti: 100, 111, 0100, 1001, 0101, 1100, 0111, 1111, 1101

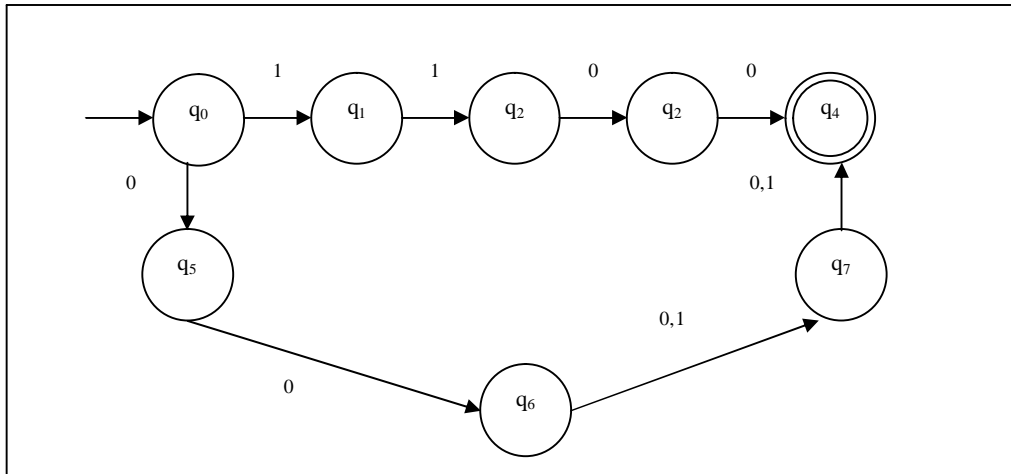
Contoh berikutnya: Kita ingin membuat mesin *Deterministic Finite Automata* yang menerima bahasa yang berupa 4 (empat) symbol yang minimal memuat 2 (dua) buah '0' (yang tidak perlu berturutan). Diketahui  $\Sigma = (0,1)$ . Disini kita agak kesulitan membuat ekspresi regular dari permasalahan tersebut. Maka kita coba langsung mengkonstruksi *Deterministic Finite Automata*-nya dengan jalan melihat semua kemungkinan yang ada.

Kemungkinan pertama adalah dua buah nol terletak di paling ujung. Kita lihat pada gambar 25.



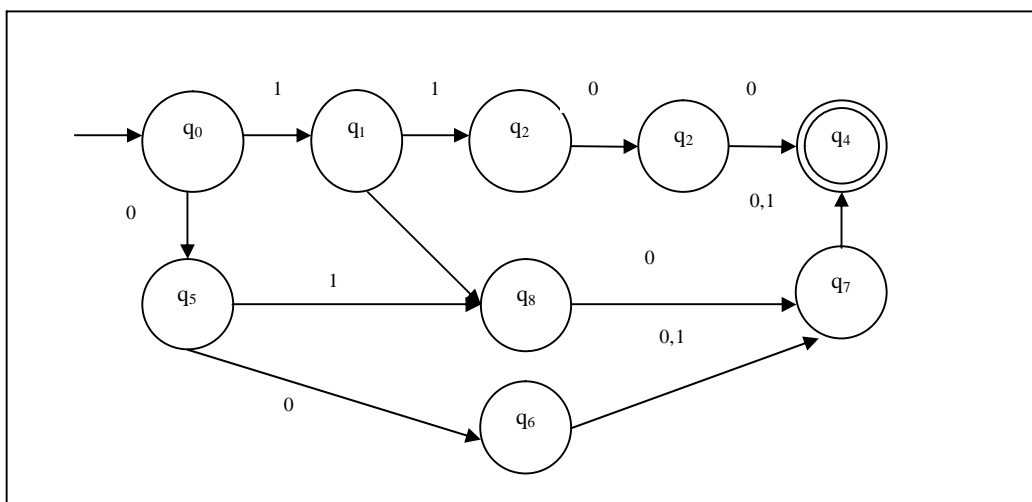
Gambar 25. DFA untuk 1100

Kemungkinan kedua adalah dua buah nol terletak di paling awal, kita tambahkan kemungkinan tersebut sehingga gambarnya dapat dilihat pada gambar 26.



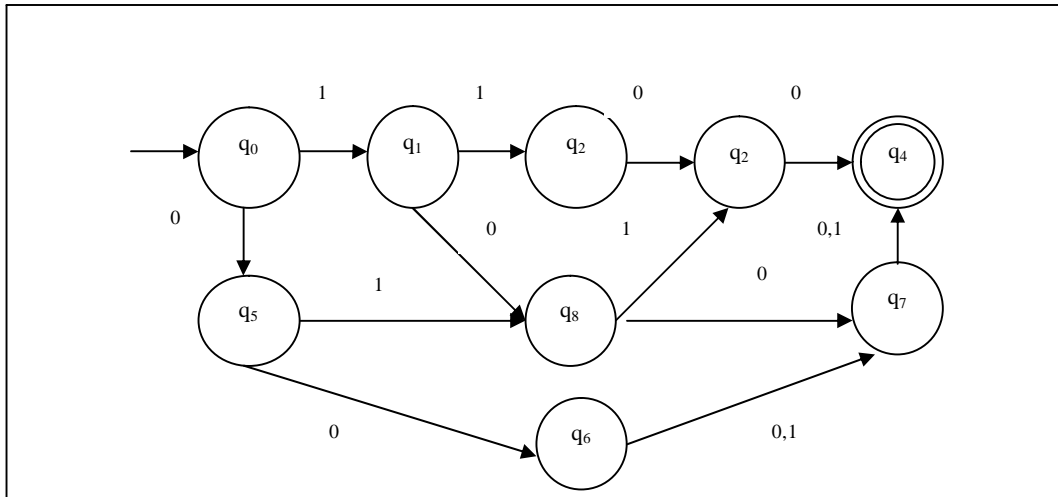
Gambar 26. DFA untuk  $(1100) + (00(1+0)(1+0))$

Selanjutnya kita tambahkan kemungkinan untuk tiga symbol pertama sudah memuat dua buah nol, dapat dilihat pada gambar 27



Gambar 27. DFA untuk  $(1100) + (00(1+0)(1+0))$

Akhirnya kita tambahkan kemungkinan bila tiga symbol pertama baru memuat satu buah nol, kita buat transisi dari state  $q_8$  ke  $q_3$ . Hasilnya dapat dilihat pada gambar 28.



Gambar 28. Hasil akhir mesin DFA

Untuk melengkapkannya menjadi sebuah *Deterministic Finite Automata* dapat pula anda tambahkan transisi ke state  $\emptyset$  :

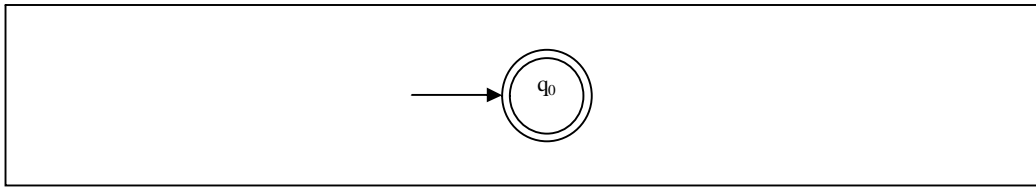
$\delta(q_2, 1) = \emptyset$ ,  $\delta(q_3, 1) = \emptyset$ ,  $\delta(q_4, 0) = \emptyset$ , dan  $\delta(q_4, 1) = \emptyset$ , yang sebenarnya tidak menuju solusi.

Kalau kita jeli sebenarnya dapat dibaca makna dari setiap *state* diatas:

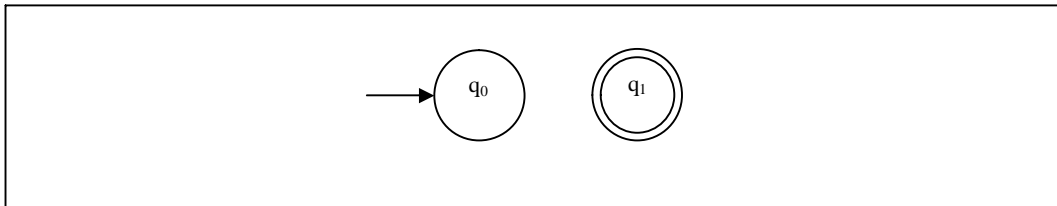
- state  $q_4$  : mesin sudah menerima empat symbol, memuat minimal dua buah nol
- state  $q_7$  : mesin sudah menerima tiga symbol, memuat minimal dua buah nol
- state  $q_3$  : mesin sudah menerima tiga symbol, memuat satu buah nol
- state  $q_6$  : mesin sudah menerima dua symbol, memuat dua buah nol
- state  $q_8$  : mesin sudah menerima dua symbol, memuat satu buah nol
- state  $q_2$  : mesin sudah menerima dua symbol, tidak memuat nol
- state  $q_1$  : mesin sudah menerima satu symbol, tidak memuat nol
- state  $q_5$  : mesin sudah menerima satu symbol, memuat satu buah nol

\*Catatan :

Untuk  $ER = \varepsilon$ , dan  $ER = \emptyset$ , bisa anda lihat otomatanya pada gambar 29 dan 30. Pada gambar 11 tanpa menerima *input* (*input* kosong/*empty*/ $\varepsilon$ ) mencapai *state* akhir. Pada gambar 12 tidak ada jalan untuk mencapai *state* akhir, maka ekspresi reguleranya  $= \emptyset$ .



Gambar 29. FSA untuk  $ER = \varepsilon$



Gambar 30. FSA untuk  $ER = \varepsilon$