Tata Bahasa (grammer) didefinisikan dengan empat (4) tupel G = ({V, T, P, S}) dimana :

V = Himpunan simbol variabel / non terminal

T = Himpunan simbol terminal

P = Kumpulan aturan produksi

S = Simbol awal

V.1 Aturan Produksi Bahasa Reguler

Kita masih ingat dengan aturan produksi dari bahasa regular (tipe 3) yaitu :

$$\alpha \rightarrow \beta$$

α adalah sebuah simbol variabel.

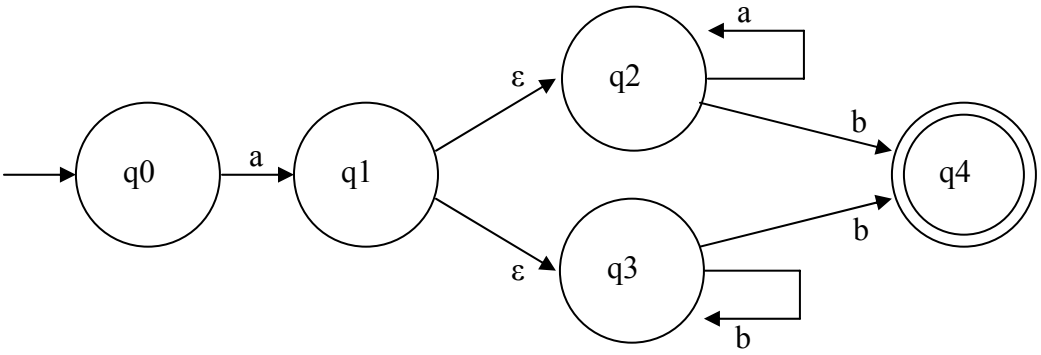
β maksimal memiliki sebuah simbol variabel yang bila ada terletak diposisi paling kanan.

Batasannya bertambah lagi, dimana ruas kanan maksimal memiliki sebuah simbol variabel yang terletak paling kanan. Artinya bisa memiliki simbol terminal dengan jumlah tidak dibatasi, tetapi bila terdapat simbol variabel maka simbol variabel tersebut hanya berjumlah satu (1) dan terletak paling kanan.

V.2 Mengkonstruksi Aturan Produksi dari Suatu Finite State Automata

Dalam mengkonstruksi aturan produksi tata bahasa regular dari suatu FSA , perlu kita ingat yang menjadi perhatian adalah state-state yang bisa menuju ke state akhir.

Contoh 1 : Mesin FSA



Pada mesin FSA contoh 1, memiliki simbol input ‘a’ dan ‘b’.

- Misal kita identikan state awal qo dengan simbol awal S.

  δ (q0, a) = q1

  Dapat ditulis :

  S → aE

  Dimana E kita identikan dengan q1.

- Dari q1 terdapat transisi :

  δ (q1, ε) = q2 dan δ (q1, ε) = q3

  Dapat ditulis :

  E → A E → B

  Dimana A kita identikan dengan q2 dan B kita identikan dengan q3.

- Selanjutnya dapat kita lihat, dari state q2 dengan input ‘a’ kembali ke state q2 dan dari state q3 dengan input ‘b’ kembali ke state q3.

  δ (q2, a) = q2 dan δ (q3, b) = q3

  Dapat ditulis :

  A → aA B → bB

- Selanjutnya, dari state q2 dengan input ‘b’ menuju state q4 dan dari state q3 dengan input ‘b’ menuju ke state q4. Sementara q4 adalah himpunan state akhir dan dari state q4 tidak ada lagi busur keluar, maka :

  δ (q2, b) = q4 dan δ (q3, b) = q4

  Dapat ditulis :

  A → b B → b

- Kumpulan aturan produksi yang kita peroleh bisa ditulis sebagai berikut :

  S → aE

  E → A | B

  A → aA | b

  B → bB | b

Secara formal dapat ditulis :

V = {S, E, A, B}

T = {a, b}

P = { S → aE , E → A | B , A → aA | b , B → bB | b }

S = S

V.3 Finite State Automata untuk Suatu Tata Bahasa Reguler

Jika sebelumnya dari suatu diagram transisi FSA dapat dibuat aturan-aturan produksi tata bahasa regularnya, maka sebaliknya bisa juga mengkonstruksi diagram transisi FSA untuk suatu tata bahasa regular yang diketahui aturan-aturan produksinya.

Contoh 2 : Tata bahasa regular

S → aB | bA | ε

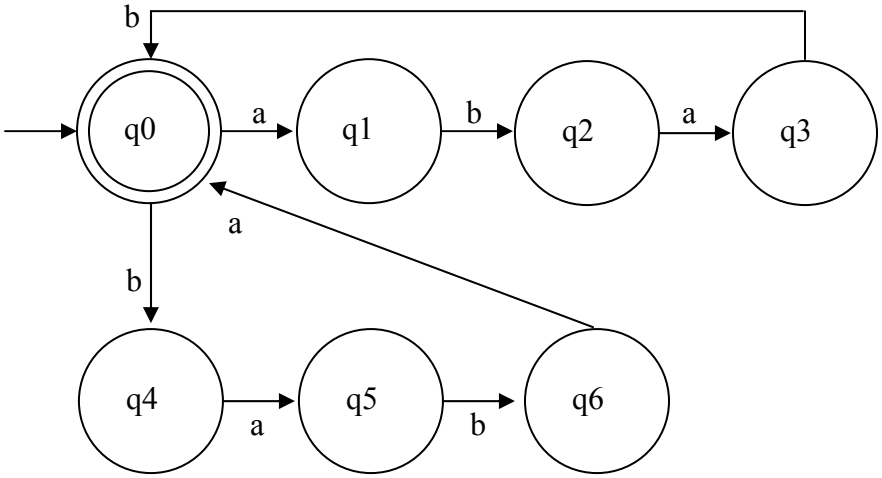
A → abaS

B → babS

Kita dapat langsung gambar atau rancang diagram transisi FSA nya!

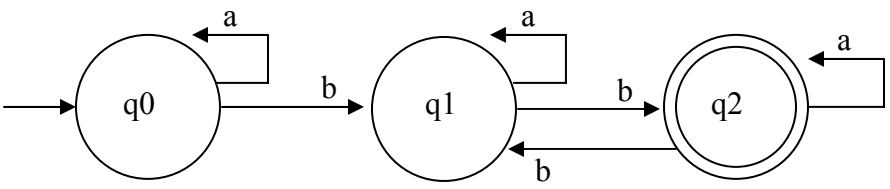
S identik dengan q0 ; A identik dengan q1; dan B identik dengan q4.

Lengkapnya adalah sebagai berikut :



Latihan 5

1. Diketahui FSA sebagai berikut :



   Kontruksi tata bahasa regular nya !

2. Buatlah FSA dari aturan-aturan produksi tata bahasa regular berikut ini :

   S → 0A

   A → 10A | ε

3. Buatlah FSA dari aturan-aturan produksi tata bahasa regular berikut ini :

   A → 00B | 11D | 0

   B → 00D | 11C | 1

   C → 00B | 11D | 0

   D → 00D | 1