

BAHASA REGULER

1. Ekspresi Regular

- Bahasa regular adalah penyusun ekspresi regular (ER)
- Ekspresi regular terdiri dari kombinasi simbol-simbol atomik menggunakan 3 operator : concate, alternate, dan closure/repetisi.
- Pada kasus scanner, simbol-simbol atomik adalah karakter-karakter di dalam program sumber.
- Dua buah ekspresi regular adalah ekuivalen jika keduanya menyatakan bahasa yang sama
- Operasi Regular :
- Katenasi /konkatenasi atau sequencing disajikan dengan physical adjacency
 - ekspresi regular ' $\langle \text{letter} \rangle \langle \text{digit} \rangle$ ' bentuk penyajian sederhana (diasumsikan sebagai definisi yang jelas dari letter dan digit) komposisi terurut dari letter diikuti dengan digit
 - " $\langle \rangle$ " dan " $\langle \rangle$ " digunakan untuk mengidentifikasi simbol-simbol yang merepresentasikan simbol-simbol spesifik (menggunakan ekspresi regular)
 - Kita bisa menggunakan " $::=$ " (ekivalensi) untuk menggabungkan ekspresi regular yang didefinisikan dengan $\langle \text{letter} \rangle$ dan $\langle \text{digit} \rangle$
 - Concatenation adalah penyambungan dua buah string.
Diberikan dua string : $x = abc$, dan $y = 123$
Contoh : $\text{concate}(xy) = x.y = xy = abc123$, $x. \epsilon = x = abc$
 $A = \{cat, dog\}$ dan $B = \{house\}$, maka $A . B = \{cathouse, doghouse\}$
 $A.(\epsilon) = (\epsilon) . A = A$
bersifat asosiatif : $x(yz) = (xy)z$
- Alternasi membolehkan pilihan dari beberapa pilihan dan biasanya disajikan dengan operator '|'
 - $\langle \text{digit} \rangle ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$
contoh yang menggunakan juga operator ekivalensi
 - Bentuk tulisan cepat tertentu juga biasanya digunakan dengan alternasi (khususnya ellipsis)
 - $\langle \text{letter} \rangle ::= a | b | \dots | z | A | B | \dots | Z$
Can use the ellipses (" \dots ") when a sequence is well defined
 - Alternation adalah pilihan satu di antara dua buah string.
Contoh : $\text{alternate}(xy) = x | y = abc$ atau 123
 - Tiga sifat aljabar alternation :
 - ◆ Operasi alternation bersifat komutatif : $x | y = y | x$
 - ◆ Operasi alternation bersifat asosiatif : $x | (y | z) = (x | y) | z$
 - ◆ Elemen identitas operasi alternation adalah dirinya sendiri : $x | x = x$
 - Sifat distributif concatenation terhadap alternation : $x (y | z) = xy | xz$
- Repetisi membolehkan ekspresi dari konstruksi yang diulang beberapa kali
 - Terdapat 2 operator yang digunakan yaitu superscript '+' dan superscript '*'
 - $\langle \text{word} \rangle ::= \langle \text{letter} \rangle^+$
 - this implies a word consists of one or more letters (* would imply zero or more letters and a word must have at least one letter so we use +)
 - * yaitu karakter asterik (penutup bintang atau *star/ Kleene closure*) artinya bisa tidak muncul, bisa juga muncul berhingga kali (0 s/d n).
 - + yaitu penutup plus atau *plus/positive closure* artinya minimal muncul satu kali (1 s/d n).
 - Kleene Closure : $x^* = \epsilon | x | xx | xxx | \dots = \epsilon | x | x^2 | x^3 | \dots$
 - Positive Closure : $x^+ = x | xx | xxx | \dots = x | x^2 | x^3 | \dots$
 - + atau \cup yaitu union atau gabungan
- Beberapa kesamaan :
 - ◆ Kesamaan ke-1 : $(x^*)^* = (x^*)$
 - ◆ Kesamaan ke-2 : $\epsilon | x^+ = x^+ | \epsilon = x^+$

- ◆ Kesamaan ke-3 : $(x | y)^* = \epsilon | x | y | xx | yy | xy | yx | \dots =$ semua string yang merupakan concatenation dari nol atau lebih x, y , atau keduanya.

Contoh :

$$L_1 = \{a^n b a^m \mid n \geq 1, m \geq 1\} \Leftrightarrow er_1 = a^+ b a^+$$

$$L_2 = \{a^n b a^m \mid n \geq 0, m \geq 0\} \Leftrightarrow er_2 = a^* b a^*$$

Perhatikan bahwa kita tidak bisa membuat ekspresi regular dari bahasa

$L_3 = \{a^n b a^n \mid n \geq 1\}$ atau $L_4 = \{a^n b a^n \mid n \geq 0\}$, karena keduanya tidak dihasilkan dari grammar regular.

Kesamaan 2 ekspresi regular :

$$(a b)^* a = a (b a)^*$$

Bukti :

$$\begin{aligned} (a b)^* a &= (\epsilon | (ab) | (abab) | \dots) a = (\epsilon a | (aba) | (ababa) | \dots) = (a | (aba) | (ababa) | \dots) \\ &= a (\epsilon | (ba) | (baba) | \dots) = a (b a)^* \end{aligned}$$

Latihan 1. Buktikan kesamaan ekspresi regular berikut :

1. $(a^* | b)^* = (a | b)^*$
2. $(a | b^*)^* = (a | b)^*$
3. $(a^* b)^* a^* = a^* (b a^*)^*$
4. $(a a^*)(\epsilon | a) = a^*$
5. $a(b a | a)^* b = a a^* b (a a^* b)^*$

2. AUTOMATA HINGGA (AH)

- AH didefinisikan sebagai pasangan 5 tupel : (K, V_T, M, S, Z) .
 - K : himpunan hingga stata,
 - V_T : himpunan hingga simbol input (alfabet)
 - M : fungsi transisi, menggambarkan transisi stata AH akibat pembacaan simbol input.
Fungsi transisi ini biasanya diberikan dalam bentuk tabel.
 - $S \in K$: stata awal
 - $Z \subset K$: himpunan stata penerima
- Ada dua jenis automata hingga : deterministik (AHD, DFA = *deterministic finite automata*) dan non deterministik (AHN, NFA = *non deterministik finite automata*).
 - AHD : transisi stata AH akibat pembacaan sebuah simbol bersifat tertentu.
 $M(\text{AHD}) : K \times V_T \rightarrow K$
 - AHN : transisi stata AH akibat pembacaan sebuah simbol bersifat tak tentu.
 $M(\text{AHN}) : K \times V_T \rightarrow 2^K$

2.1. Automata Hingga Deterministik (AHD)

Berikut ini sebuah contoh AHD $F(K, V_T, M, S, Z)$, dimana :

$$K = \{q_0, q_1, q_2\}$$

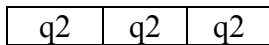
$$V_T = \{a, b\}$$

$$S = q_0$$

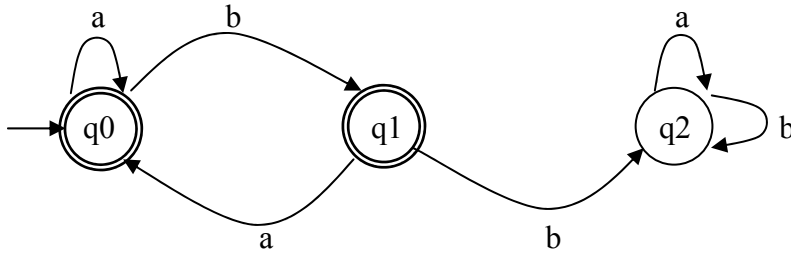
$$Z = \{q_0, q_1\}$$

M diberikan dalam tabel berikut :

	a	b
q0	q0	q1
q1	q0	q2



Ilustrasi graf untuk AHD F adalah sebagai berikut :
 Lambang stata awal adalah node dengan anak panah.
 Lambang stata akhir/penerima adalah node ganda.



Contoh kalimat yang diterima AHD : a, b, aa, ab, ba, aba, bab, abab, baba

Contoh kalimat yang tidak diterima AHD : bb, abb, abba

AHD ini menerima semua kalimat yang tersusun dari simbol a dan b yang tidak mengandung substring bb.

Contoh :

Telusurilah, apakah kalimat-kalimat berikut diterima AHD :

abababaa, aaaabab, aaabbaba

Jawab :

i) $M(q_0, abababaa) \Rightarrow M(q_0, bababaa) \Rightarrow M(q_1, ababaa) \Rightarrow M(q_0, babaa) \Rightarrow M(q_1, abaa) \Rightarrow M(q_0, baa) \Rightarrow M(q_1, aa) \Rightarrow M(q_0, a) \Rightarrow q_0$

Tracing berakhir di q0 (stata penerima) \Rightarrow kalimat abababaa diterima

ii) $M(q_0, aaaabab) \alpha M(q_0, aaabab) \alpha M(q_0, aabab) \alpha M(q_0, abab)$

$\Rightarrow M(q_0, bab) \Rightarrow M(q_1, ab) \Rightarrow M(q_0, b) \alpha q_1$

Tracing berakhir di q1 (stata penerima) \Rightarrow kalimat aaaababa diterima

iii) $M(q_0, aaabbaba) \Rightarrow M(q_0, aabbaba) \Rightarrow M(q_0, abbaba) \Rightarrow M(q_0, bbaba)$

$\Rightarrow M(q_1, bbaba) \Rightarrow M(q_2, baba) \Rightarrow M(q_2, aba) \Rightarrow M(q_2, ba) \Rightarrow M(q_2, a) \alpha q_2$

Tracing berakhir di q2 (bukan stata penerima) \Rightarrow kalimat aaabbaba ditolak

Kesimpulan : sebuah kalimat diterima oleh AHD jika tracingnya berakhir di salah satu stata penerima.

2.2. Automata Hingga Nondeterministik (AHN)

Berikut ini sebuah contoh AHN $F(K, V_T, M, S, Z)$, dimana :

$K = \{q_0, q_1, q_2, q_3, q_4\}$

M diberikan dalam tabel berikut :

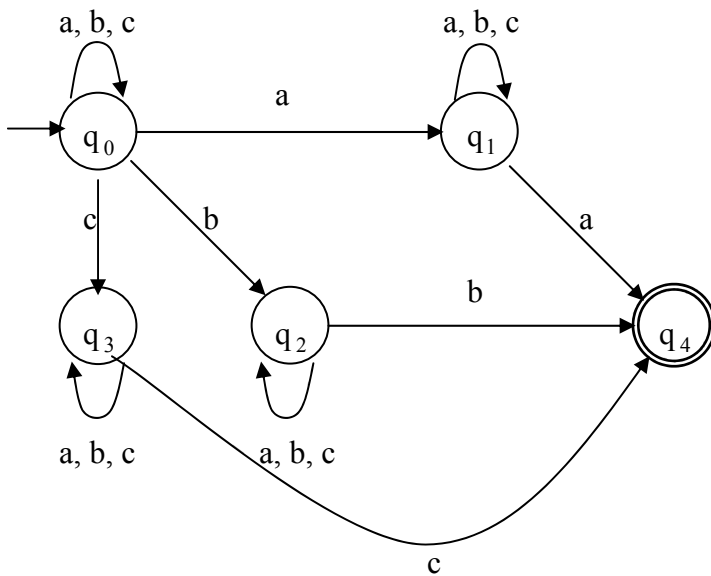
$V_T = \{a, b, c\}$

$S = q_0$

$Z = \{q_4\}$

	a	b	c
q ₀	{q ₀ , q ₁ }	{q ₀ , q ₂ }	{q ₀ , q ₃ }
q ₁	{q ₁ , q ₄ }	{q ₁ }	{q ₁ }
q ₂	{q ₂ }	{q ₂ , q ₄ }	{q ₂ }
q ₃	{q ₃ }	{q ₃ }	{q ₃ , q ₄ }
q ₄	∅	∅	∅

Ilustrasi graf untuk AHN F adalah sebagai berikut :



Contoh kalimat yang diterima AHN di atas : aa, bb, cc, aaa, abb, bcc, cbb

Contoh kalimat yang tidak diterima AHN di atas : a, b, c, ab, ba, ac, bc

Fungsi transisi M sebuah AHN dapat diperluas sebagai berikut :

1. $M(q, \varepsilon) = \{q\}$ untuk setiap $q \in K$
2. $M(q, tT) = \cup M(p_i, T)$ dimana $t \in V_T$, T adalah V_T^* , dan $M(q, t) = \{p_i\}$
3. $M(\{q_1, q_2, \dots, q_n\}, x) = \cup M(q_i, x)$, untuk $x \in V_T^*$

Sebuah kalimat di terima AHN jika :

- salah satu tracing-nya berakhir di stata penerima, atau
- himpunan stata setelah membaca string tersebut mengandung stata penerima

Contoh :

Telusurilah, apakah kalimat-kalimat berikut diterima AHN : ab, abc, aabc, aabb

Jawab :

$$iv) M(q_0, ab) \Rightarrow M(q_0, b) \cup M(q_1, b) \Rightarrow \{q_0, q_2\} \cup \{q_1\} = \{q_0, q_1, q_2\}$$

Himpunan stata tidak mengandung stata penerima \Rightarrow kalimat ab *tidak diterima*

$$v) M(q_0, abc) \Rightarrow M(q_0, bc) \cup M(q_1, bc) \Rightarrow \{M(q_0, c) \cup M(q_2, c)\} \cup M(q_1, c)$$

$$\Rightarrow \{\{q_0, q_3\} \cup \{q_2\}\} \cup \{q_1\} = \{q_0, q_1, q_2, q_3\}$$

Himpunan stata tidak mengandung stata penerima \Rightarrow kalimat abc *tidak diterima*

$$vi) M(q_0, aabc) \Rightarrow M(q_0, abc) \cup M(q_1, abc) \Rightarrow \{M(q_0, bc) \cup M(q_1, bc)\} \cup M(q_1, bc)$$

$$\Rightarrow \{\{M(q_0, c) \cup M(q_2, c)\} \cup M(q_1, c)\} \cup M(q_1, c)$$

$$\Rightarrow \{\{\{q_0, q_3\} \cup \{q_2\}\} \cup \{q_1\}\} \cup \{q_1\} = \{q_0, q_1, q_2, q_3\}$$

Himpunan stata tidak mengandung stata penerima \Rightarrow kalimat aabc *tidak diterima*

$$vii) M(q_0, aabb) \Rightarrow M(q_0, abb) \cup M(q_1, abb) \Rightarrow \{M(q_0, bb) \cup M(q_1, bb)\} \cup M(q_1, bb)$$

$$\Rightarrow \{\{M(q_0, b) \cup M(q_2, b)\} \cup M(q_1, b)\} \cup M(q_1, b)$$

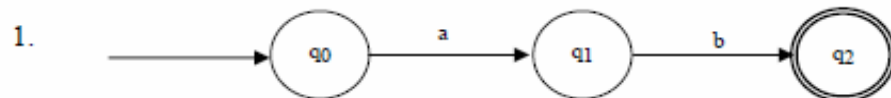
$$\Rightarrow \{\{\{q_0, q_2\} \cup \{q_2, q_4\}\} \cup \{q_1\}\} \cup \{q_1\} = \{q_0, q_1, q_2, q_4\}$$

Himpunan stata tidak mengandung stata penerima \Rightarrow kalimat aabb *diterima*

Contoh Ekspresi Reguler (ER)

- ER = 010*
String yang dibangkitkan / muncul = 01, 010, 0100, 01000 (jumlah nol di ujung kanan bisa tidak muncul, bisa juga muncul berhingga kali)
- ER = ab*cc
String yang dibangkitkan / muncul : abcc, acc, abbcc (b bisa tidak muncul/muncul berhingga kali)
- ER = a*d
String yang dibangkitkan / muncul = d, ad, aad, aaad
- ER = a+d
String yang dibangkitkan / muncul = ad, aad, aaad
- ER = a* \cup b* (\cup berarti atau)
String yang dibangkitkan / muncul = ϵ , a, b, aa, bb, aaa, bbb
- ER = a \cup b
String yang dibangkitkan / muncul = a, b
- ER = (a \cup b)*
String yang dibangkitkan / muncul = ϵ , a, b, aa, bb, aaa, bbb, ab, abb, aab, ba
- ER = 01* + 0 atau 01* \cup 0
String yang dibangkitkan / muncul = 0, 01, 011, 0111

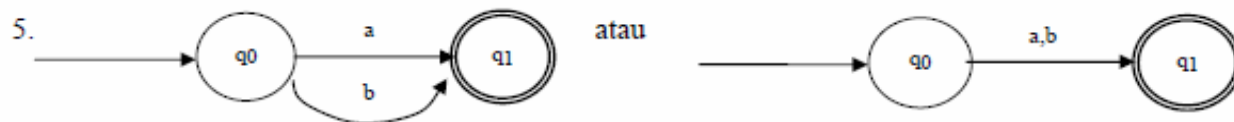
Hubungan ER dan FSA



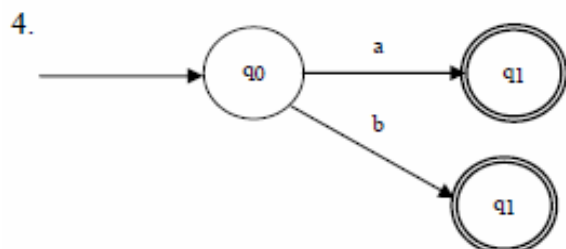
NFA dengan ER : ab



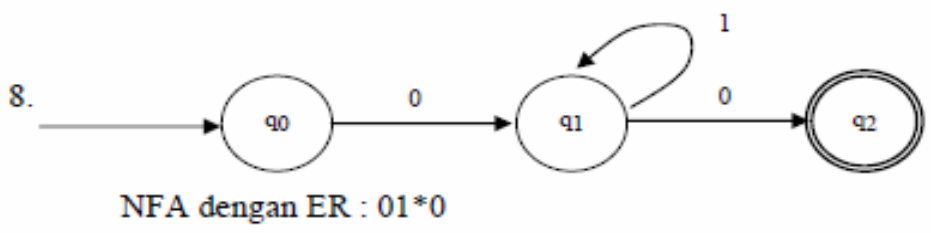
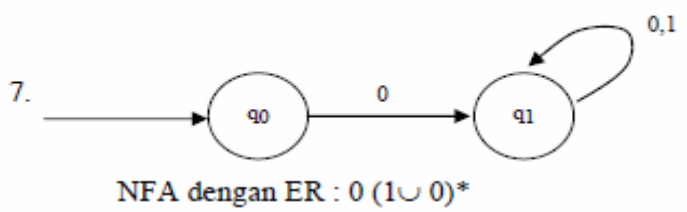
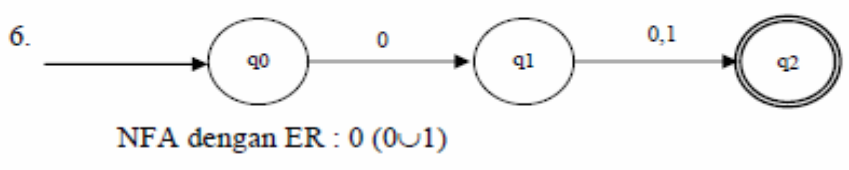
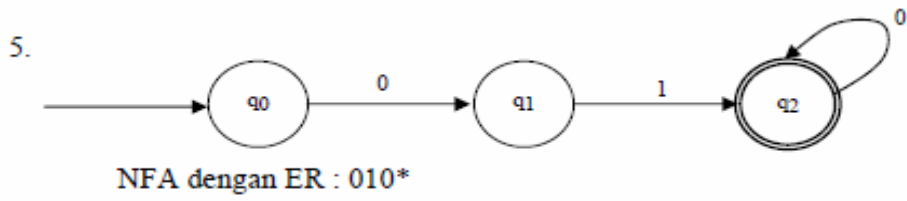
NFA dengan ER : a*b



NFA dengan ER : a \cup b



NFA dengan ER : a \cup b

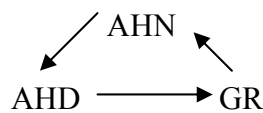


3. Ekuivalensi AHN, AHD, dan GR

AHD bisa dibentuk dari AHN.

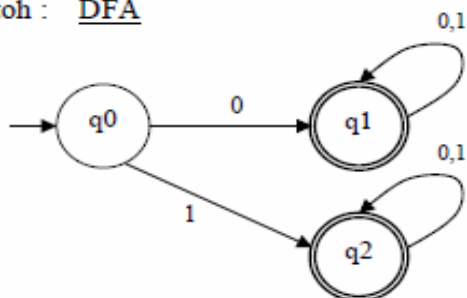
GR bisa dibentuk dari AHD.

AHN bisa dibentuk dari GR.

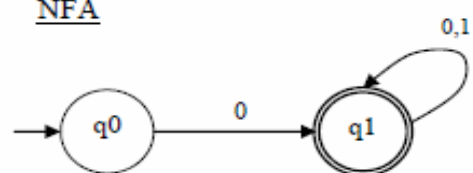


- Finite Automata dikatakan ekuivalen apabila menerima bahasa yang sama.
- Dari sebuah mesin NFA dapat dibuat mesin DFA yang ekuivalen yaitu mampu menerima bahasa yang sama.

Contoh : DFA



NFA



Kedua mesin diatas bisa menerima bahasa yang sama, yaitu dalam ekspresi reguler $0(0 \cup 1)^*$

3.1. Pembentukan AHD dari AHN

Diberikan sebuah AHN $F = (K, V_T, M, S, Z)$. Akan dibentuk sebuah AHD $F' = (K', V_{T'}, M', S', Z')$ dari AHN F tersebut. Algoritma pembentukannya adalah sbb. :

1. Tetapkan : $S' = S$ dan $V_{T'} = V_T$
2. Copykan tabel AHN F sebagai tabel AHD F' . Mula-mula $K' = K$ dan $M' = M$
3. Setiap stata q yang merupakan *nilai* (atau *peta*) dari fungsi M dan $q \notin K$, ditetapkan sebagai elemen baru dari K' . Tempatkan q tersebut pada kolom Stata M' , lakukan pemetaan berdasarkan fungsi M .
4. Ulangi langkah (3) sampai tidak diperoleh stata baru.
5. Elemen Z' adalah semua stata yang mengandung stata elemen Z .

Contoh :

Berikut ini diberikan sebuah AHN $F = (K, V_T, M, S, Z)$ dengan :

$K = \{A, B, C\}$, $V_T = \{a, b\}$, $S = A$, $Z = \{C\}$, dan M didefinisikan sebagai berikut :

Stata K AHN F	Input	
	a	b
A	[A,B]	C
B	A	B
C	B	[A,B]

Tentukan AHD hasil transformasinya.

Jawab :

Berdasarkan algoritma di atas, maka :

1. $S' = S = A$, $V_{T'} = V_T = \{a, b\}$.
2. Hasil copy tabel AHN F menghasilkan tabel AHD F' berikut :

Stata K' AHD F'	Input	
	a	b
A	[A,B]	C
B	A	B
C	B	[A,B]

3. Pada tabel AHD F' di atas terdapat stata baru yaitu [A,B]. Pemetaan [A,B] adalah : $M([A,B],a) = M(A,a) \cup M(B,a) = [A,B] \cup A = [A,B]$, dan $M([A,B],b) = M(A,b) \cup M(B,b) = C \cup B = [B,C]$, sehingga diperoleh tabel berikut :

Stata K' dari AHD F'	Input	
	a	b
A	[A,B]	C
B	A	B
C	B	[A,B]
[A,B]	[A,B]	[B,C]

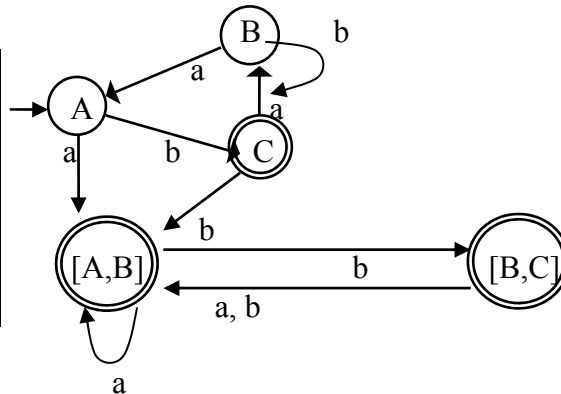
4. Langkah (3) di atas menghasilkan stata baru yaitu [B,C]. Setelah pemetaan terhadap [B,C] diperoleh tabel berikut :

Stata K' dari AHD F'	Input	
	a	b
A	[A,B]	C
B	A	B
C	B	[A,B]
[A,B]	[A,B]	[B,C]
[B,C]	[A,B]	[A,B]

5. Setelah langkah (4) di atas tidak terdapat lagi stata baru.

Dengan demikian AHD F' yang dihasilkan adalah : $AHD F' = (K', V_T', M', S', Z')$, dimana : $K' = \{A, B, C, [A,B], [B,C]\}$, $V_T' = \{a, b\}$, $S' = A$, $Z' = \{C, [B,C]\}$. Fungsi transisi M' serta graf dari AHD F' adalah sebagai berikut :

Stata K' dari AHD F'	Input	
	a	b
A	[A,B]	C
B	A	B
C	B	[A,B]
[A,B]	[A,B]	[B,C]
[B,C]	[A,B]	[A,B]



3.2. Pembentukan GR dari AHD

Diketahui sebuah AHD $F = (K, V_T, M, S, Z)$. Akan dibentuk GR $G = (V_T', V_N, S', Q)$.

Algoritma pembentukan GR dari AHD adalah sebagai berikut :

1. Tetapkan $V_T' = V_T$, $S' = S$, $V_N = S$
2. Jika $A_p, A_q \in K$ dan $a \in V_T$, maka :

$$M(A_p, a) = A_q \text{ ekuivalen dengan produksi : } \begin{cases} A_p \rightarrow aA_q, & \text{jika } A_q \notin Z \\ A_p \rightarrow a, & \text{jika } A_q \in Z \end{cases}$$

Contoh : Diketahui sebuah AHD F dengan $Z = \{S\}$ dan fungsi transisi M sebagai berikut :

Stata K AHD F	Input	
	0	1
S	B	A
A	C	S
B	S	C
C	A	B

Dengan algoritma di atas maka diperoleh $Q(GR)$ sbb. :

$$\begin{aligned} M(S,0) = B &\Leftrightarrow S \rightarrow 0B & M(S,1) = A &\Leftrightarrow S \rightarrow 1A \\ M(A,0) = C &\Leftrightarrow A \rightarrow 0C & M(A,1) = S &\Leftrightarrow A \rightarrow 1 \\ M(B,0) = S &\Leftrightarrow B \rightarrow 0 & M(B,1) = C &\Leftrightarrow B \rightarrow 1C \\ M(C,0) = A &\Leftrightarrow C \rightarrow 0A & M(C,1) = B &\Leftrightarrow C \rightarrow 1B \end{aligned}$$

GR yang dihasilkan adalah $G(V_T', V_N, S', Q)$, dengan $V_T' = \{0,1\}$, $V_N = \{S, A, B, C\}$, $S' = S$, dan $Q = \{S \rightarrow 0B, S \rightarrow 1A, A \rightarrow 0C, B \rightarrow 1C, C \rightarrow 0A, C \rightarrow 1B, A \rightarrow 1, B \rightarrow 0\}$

3.3. Pembentukan AHN dari GR

Diketahui GR $G = (V_T, V_N, S, Q)$. Akan dibentuk AHN $F = (K, V_T', M, S', Z)$.

Algoritma pembentukan AHN dari GR :

1. Tetapkan $V_T' = V_T$, $S' = S$, $K = V_N$
2. Produksi $A_p \rightarrow a A_q$ ekuivalen dengan $M(A_p, a) = A_q$
Produksi $A_p \rightarrow a$ ekuivalen dengan $M(A_p, a) = X$, dimana $X \notin V_N$
3. $K = K \cup \{X\}$
4. $Z = \{X\}$

Contoh

Diketahui GR $G = (V_T, V_N, S, Q)$ dengan : $V_T = \{a, b\}$, $V_N = \{S, A, B\}$, $S = S$, dan $Q = \{S \rightarrow aS, S \rightarrow bA, A \rightarrow aA, A \rightarrow aB, B \rightarrow b\}$

Terapkan algoritma di atas untuk memperoleh AHN F sebagai berikut :

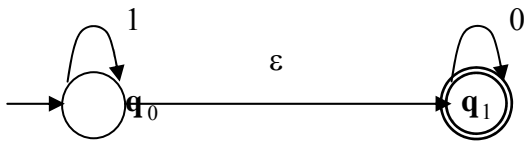
1. $V_T' = V_T = \{a, b\}$, $S' = S$, $K = V_N = \{S, A, B\}$
 2. $S \rightarrow aS \Leftrightarrow M(S,a) = S$, $S \rightarrow bA \Leftrightarrow M(S,b) = A$,
 $A \rightarrow aA \Leftrightarrow M(A,a) = A$, $A \rightarrow aB \Leftrightarrow M(A,a) = B$,
 $B \rightarrow b \Leftrightarrow M(B,b) = X$
- AHN yang diperoleh : $F(K, V_T', M, S', Z)$, dengan
 $K = \{S, A, B, X\}$, $V_T' = \{a, b\}$, $S' = S$, $Z = \{X\}$,

Tabel M :

Stata K AHN F	Input	
	a	b
S	S	A
A	[A,B]	ϕ
B	ϕ	X
X	ϕ	ϕ

4. AHN Dengan Transisi Hampa

Perhatikan AHN berikut.



AHN di atas mengandung ruas dengan bobot ϵ . AHN demikian dinamakan AHN dengan transisi ϵ , atau singkatnya AHN- ϵ . AHN- ϵ di atas menerima bahasa $L = \{1^i 0^j \mid i, j \geq 0\}$

4.1. Ekuivalensi Ahn- ϵ Dengan ER (Ekspresi Regular)

Jenis ER	Simbol ER	AHN
Simbol hampa	ϵ	
ER hampa	ϕ atau $\{\}$	
ER umum	r	
Alternation	$r_1 \mid r_2$	
Concatenation	$r_1 r_2$	
Kleene Closure	r^*	

Pengantar Tekom

Contoh :

Tentukan AHN untuk ekspresi reguler $r = 0(1 | 23)^*$

Jawab :

