



BAB 5: FINITE AUTOMATA

Bab 5: Otomata (Automata) Hingga

Agenda.

- Otomata (Automata) Hingga
 - Deterministic Finite Automata – DFA (Otomata Hingga Deterministik)
 - Equivalen 2 DFA
 - Finite State Machine – FSA (Mesin Stata Hingga)
- Non-Deterministic Finite Automata – NFA (Otomata Hingga Non-Deterministik)
- NFA to DFA Conversion
- DFA to Grammar Conversion
- Grammar to NFA Conversion
- Ekuivalensi NFA – RE (Regular Expression)

Automata Hingga | 3

Otomata (Automata) Hingga

Definisi.

- Otomata Hingga (AH)/Automata Hingga (AH)/Finite Automata (FA) 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

Automata Hingga | 4

Otomata (Automata) Hingga

- Terdiri dari 2 jenis:
 1. Deterministic FA (DFA), dimana transisi stata FS merupakan akibat dari pembacaan sebuah simbol bersifat tertentu; dan
 2. Non-Deterministic FA (NFA), dimana transisi stata FS merupakan akibat dari pembacaan sebuah simbol bersifat tak tentu.

Otomata (Automata) Hingga

Deterministic Finite Automata (DFA).

- **Contoh kasus:** Diketahui sebuah DFA sebagai berikut $F(K, V_T, M, S, Z)$, dimana:

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

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

$$S = \{q_0\}$$

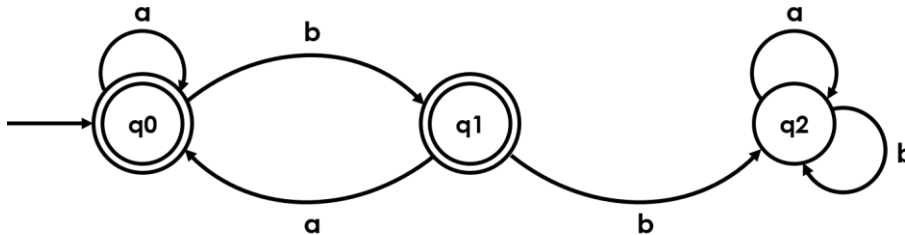
$$Z = \{q_0, q_1\}$$

M diberikan dalam tabel berikut:

	a	b
q0	q0	q1
q1	q0	q2
q2	q2	q2

Otomata (Automata) Hingga

- Ilustrasi *graph* untuk DFA F adalah sebagai berikut:



Problem: Telusurilah, apakah string *abababaa* diterima DFA F?

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

Automata Hingga | 7

Otomata (Automata) Hingga

- Contoh kasus lain:**
Diketahui table M sebuah DFA Y di samping ini dengan nilai $Z = \{q_0 \text{ dan } q_2\}$.
Berdasarkan tabel M tersebut, maka tentukan:
a. pasangan *tuple* dan
b. graph otomata untuk DFA tersebut.

	a	b	c
q0	q0	q2	q3
q1	q0	q2	q1
q2	q1	q3	q1
q3	q3	q0	q2

Automata Hingga | 8

Otomata (Automata) Hingga

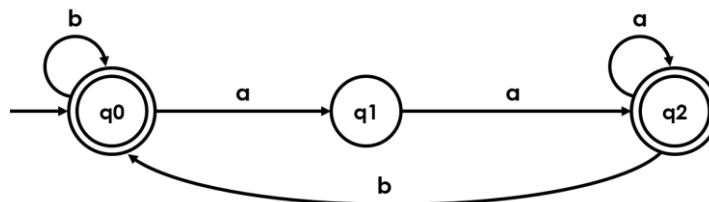
Latihan 1.

Problem 1: Lakukanlah penelusuran menggunakan gambar *graph* sebelumnya untuk menguji apakah string-string *berikut* diterima DFA Y?

1. $a^2b^3c^2$
2. $abcabcabc$
3. $a^3b^3c^3$

Otomata (Automata) Hingga

Problem 2: Diketahui sebuah *graph* DFA X sebagai berikut, dimana setiap stata hanya memiliki satu buah nilai input dan tidak ada stata yang disinggahi > 1 kali secara berurutan untuk setiap nilai inputnya!



Berdasarkan gambar *graph* di atas, maka tentukanlah:

1. K , V_T , S , Z , dan tabel M ;
2. apakah string-string *berikut* diterima DFA X?
 1. $aaabbaaa$
 2. $aabbaabaa$

Otomata (Automata) Hingga

Problem 3: Rancanglah sebuah DFA Y agar dapat menerima string-string berikut, dimana setiap stata hanya memiliki satu buah nilai input dan tidak ada stata yang disinggahi > 1 kali secara berurutan untuk setiap nilai input yang sama!

1. abaaba
2. abbaabba

Otomata (Automata) Hingga

Equivalensi 2 DFA.

- Dua buah DFA dikatakan equivalen jika keduanya dapat menerima bahasa yang sama.
- Misalkan kedua DFA tersebut adalah A dan A' dan bahasa yang diterima adalah bahasa L yang dibangun oleh alfabet $V_T = \{a_1, a_2, a_3, \dots, a_n\}$.
- Untuk menentukan equivalensi 2 buah DFA, maka algoritma yang digunakan adalah sebagai berikut:

Otomata (Automata) Hingga

1. Berikan nama kepada semua stata masing-masing FDA dengan nama berbeda. Misalkan nama-nama tersebut adalah: S, A_1, A_2, \dots untuk FDA A , dan : S', A_1', A_2', \dots untuk FDA A' .
2. Buat tabel $(n+1)$ kolom, yaitu kolom-kolom: $(v, v'), (v_{a_1}, v_{a_1}'), \dots, (v_{a_n}, v_{a_n}')$, yaitu pasangan terurut (stata FDA A , stata FDA A').
3. Isikan (S, S') pada baris pertama kolom (v, v') , dimana S dan S' masing-masing adalah *stata awal* masing-masing FDA.

Automata Hingga | 13

Otomata (Automata) Hingga

4. Jika terdapat *edge* dari S ke A_1 dengan label a_1 dan jika terdapat *edge* dari S' ke A_1' juga dengan label a_1 , isikan pasangan terurut (A_1, A_1') sebagai pada baris pertama kolom (v_{a_1}, v_{a_1}') . Lakukan hal yang sama untuk kolom-kolom berikutnya.
5. Perhatikan nilai-nilai pasangan terurut pada baris pertama. Jika terdapat nilaipasangan terurut pada kolom (v_{a_1}, v_{a_1}') s/d (v_{a_n}, v_{a_n}') yang *tidak sama* dengan nilai pasangan terurut (v, v') , tempatkan nilai tersebut pada kolom (v, v') *baris-baris* berikutnya. Lakukan hal yang sama seperti yang dilakukan pada langkah (4). Lanjutkan dengan langkah (5).

Automata Hingga | 14

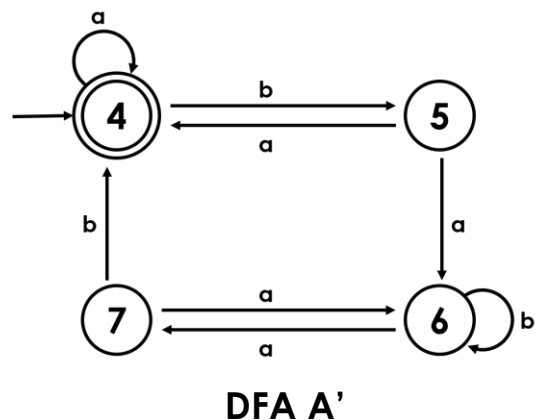
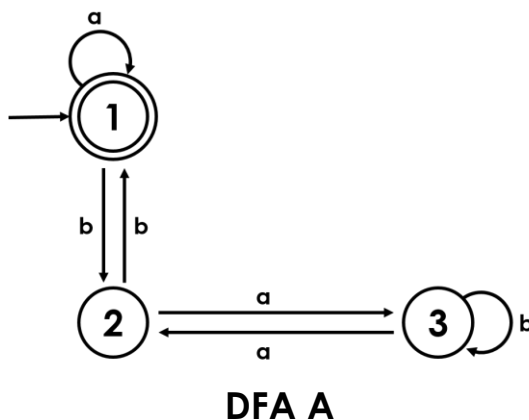
Otomata (Automata) Hingga

6. Jika selama proses di atas dihasilkan sebuah nilai pada kolom (v, v') , dengan komponen v merupakan *stata penerima* sedangkan komponen v' bukan, atau sebaliknya, maka *kedua DFA tersebut tidak ekuivalen*. Proses dihentikan.
7. Jika kondisi (6) tidak dipenuhi dan jika tidak ada lagi pasangan terurut baru yang harus ditempatkan pada kolom (v, v') maka proses dihentikan dan *kedua DFA tersebut ekuivalen*.

Automata Hingga | 15

Otomata (Automata) Hingga

- **Contoh kasus:** Periksalah equivalensi kedua DFA berikut.

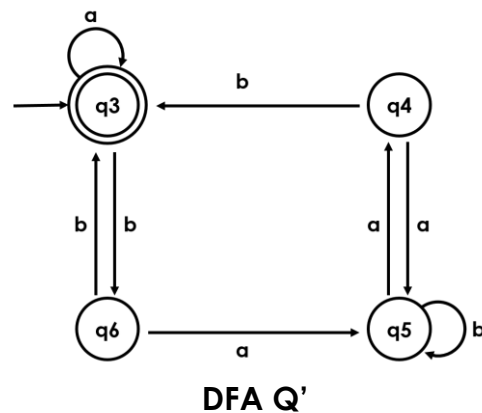
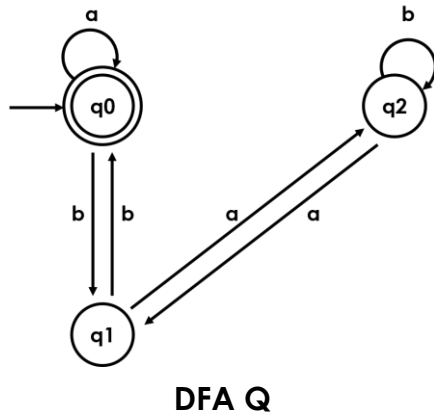


Automata Hingga | 16

Otomata (Automata) Hingga

Latihan 2.

Problem 1: Periksa apakah ekuivalensi kedua DFA berikut.



Automata Hingga | 17

Otomata (Automata) Hingga

Problem 2: Gambarkan dua buah DFA Q dan Q' yang ekuivalen dan buktikan ekuivalen kedua DFA tersebut dalam sebuah tabel jika input yang diberikan adalah a dan b dan setiap DFA memiliki minimum 3 state!

Automata Hingga | 18

Otomata (Automata) Hingga

Finite State Machine (FSM).

- Varian automata hingga yang memiliki output.
- FSM didefinisikan sebagai pasangan 6 tuple $F(K, V_T, S, Z, f, g)$ dimana:

K : Himpunan hingga state

V_T : Himpunan hingga simbol/terminal input

$S \in K$: State awal

Z : Himpunan hingga simbol/terminal output

f : Fungsi input

g : Fungsi output

Automata Hingga | 19

Otomata (Automata) Hingga

- **Contoh kasus:** Diketahui sebuah FSM memiliki 2 simbol *input*, 3 *state*, dan 3 simbol *output* sebagai berikut:

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

$S = q_0$

$V_T = \{a, b\}$

$Z = \{x, y, z\}$

Fungsi f :

$f(q_0, a) = q_1$ $f(q_0, b) = q_2$

$f(q_1, a) = q_2$ $f(q_1, b) = q_1$

$f(q_2, a) = q_0$ $f(q_2, b) = q_1$

Fungsi g :

$g(q_0, a) = x$ $g(q_0, b) = y$

$g(q_1, a) = x$ $g(q_1, b) = z$

$g(q_2, a) = z$ $g(q_2, b) = y$

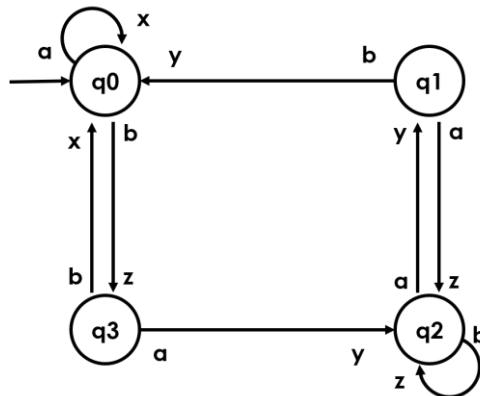
Problem: Sajikan FSM tersebut dalam bentuk graph dan tabel!

Automata Hingga | 20

Otomata (Automata) Hingga

Latihan 3.

Problem 1: Tentukan pasangan tupel FSM berikut.



Automata Hingga | 21

Otomata (Automata) Hingga

Problem 2: Perhatikan tabel berikut, kemudian gambarkan graph FSM otomata dan tuliskan pasangan 6 tuple berdasarkan tabel FSM tersebut.

	a	b	c
q0	q1, x	q2, y	q0, z
q1	q4, y	q2, x	q3, y
q2	q3, x	q4, z	q0, z
q3	q1, y	q0, z	q2, x

Automata Hingga | 22

Otomata (Automata) Hingga

Finite State Machine (FSM) Binary Adder.

- Sifat penjumlahan biner bergantung pada statusnya: *carry* atau *not carry*.
- Pada status *not carry* berlaku:
 $0 + 0 = 0$, $1 + 0 = 0 + 1 = 1$, $1 + 1 = 0$
- Pada status *carry* berlaku:
 $0 + 0 = 1$, $1 + 0 = 0 + 1 = 0$, $1 + 1 = 1$
- Pada status *not carry* menjadi *blank* (*b*), sedangkan pada status *carry* menjadi 1.

Automata Hingga | 23

Otomata (Automata) Hingga

- **Contoh kasus:** Diketahui nilai setiap tupel untuk FSM ini adalah:

$K = N$ (not carry), C (carry), dan S (stop)

$S = N$

$V_T = \{00, 01, 10, 11, b\}$

$Z = \{0, 1, b\}$

M diberikan dalam tabel berikut:

	00	01	10	11	b
N	N, 0	N, 1	N, 1	C, 0	S, b
C	N, 1	C, 0	C, 0	C, 1	S, 1

Automata Hingga | 24

Otomata (Automata) Hingga

- **Contoh kasus:** Hitunglah $1101011 + 0111011$

Jawab:

Input = pasangan digit kedua bilangan, mulai dari LSB

= 11, 11, 00, 11, 01, 11, 10, b

Output = 0, 1, 1, 0, 0, 1, 0, 1 (tuliskan dari kanan → kiri)

State = C, C, N, C, C, C, C, S (tuliskan dari kanan → kiri)

Periksa =
$$\begin{array}{r} 1101011 \\ 0111011 + \\ \hline \end{array}$$

10100110

S CCC CNCC

(baca dari kiri → kanan)

Automata Hingga | 25

Otomata (Automata) Hingga

Latihan 4.

Problem 1:

Menggunakan mesin FSM binary adder, maka hitunglah:

(a) $10010110 + 01010111$

(b) $10111010 + 11110100$

Automata Hingga | 26

Otomata (Automata) Hingga

Non-Deterministic Finite Automata (NFA).

- Non-Deterministik Finite Automata (NFA)/Automata Hingga Non-Deterministik (AHN) adalah sebuah mesin automata yang memiliki > 1 stata yang dituju apabila menerima sebuah nilali input
- NFA merupakan varian mesin automata yang ambigu sehingga perlu dikonversikan ke mesin automata deterministik (DFA)

Automata Hingga | 27

Otomata (Automata) Hingga

NFA to DFA Conversion

- **Contoh kasus:** Diketahui sebuah NFA sebagai berikut $F(K, V_T, M, S, Z)$, dimana:

$$K = \{A, B, C\}$$

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

$$S = A$$

$$Z = \{C\}$$

M diberikan dalam tabel berikut:

	a	b
A	{A,B}	{C}
B	A	B
C	B	{A,B}

Automata Hingga | 28

Otomata (Automata) Hingga

Algoritma konversi:

1. Tetapkan: $S' = S$ dan $V_T' = V_T$, kemudian salinkan tabel DFA F sebagai tabel DFA F' , sehingga $K' = K$ dan $M' = M$.
2. Setiap stata q yang merupakan *nilai* (atau *peta*) dari fungsi dan $q \notin K$, ditetapkan sebagai elemen baru dari K' . Tempatkan q tersebut pada kolom stata M' , lakukan pemetaan berdasarkan fungsi M .
3. Ulangi langkah (3) sampai tidak diperoleh stata baru.
4. Elemen Z' adalah semua stata yang mengandung stata elemen Z .

Automata Hingga | 29

Otomata (Automata) Hingga

Latihan 5.

- **Problem 1:** Konversikan NFA berikut menjadi DFA jika diketahui sebuah NFA sebagai berikut $F(K, V_T, M, S, Z)$, dimana:

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

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

$$S = \{q_0\}$$

$$Z = \{q_2\}$$

M diberikan dalam tabel berikut:

	a	b	c
q0	{q0,q1}	{q0,q2}	{q0}
q1	{q1,q2}	{q2}	{q1}
q2	{q1}	{q0,q1}	{q2}

Automata Hingga | 30

Otomata (Automata) Hingga

- Problem 2:** Konversikan NFA berikut menjadi DFA jika diketahui sebuah NFA sebagai berikut $F(K, V_T, M, S, Z)$, dimana:

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

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

$$S = \{q_0\}$$

$$Z = \{q_4\}$$

M diberikan dalam tabel berikut:

	a	b	c
q0	{q0,q1}	{q0,q2}	{q0,q3}
q1	{q1,q4}	{q1}	{q1}
q2	{q2}	{q2,q4}	{q2}
q3	{q3}	{q3}	{q3,q4}
q4	∅	∅	∅

Automata Hingga | 31

Otomata (Automata) Hingga

Problem 3: Telusurilah, apakah kalimat-kalimat berikut diterima NFA: ab dan abc

Latihan 6.

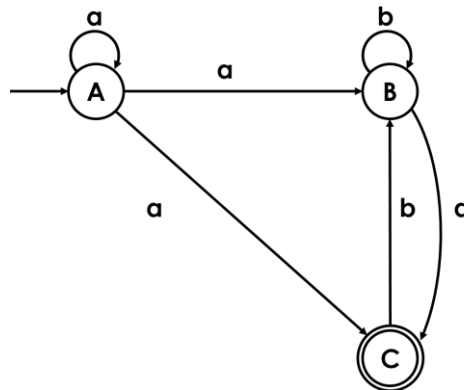
Problem 1: Berdasarkan tabel M pada contoh kasus di atas, telusurilah apakah kalimat-kalimat berikut diterima NFA: aabc, aabb, dan abcabc.

Automata Hingga | 32

Otomata (Automata) Hingga

Latihan 7.

Problem 1: Konversikan otomata NFA berikut menjadi otomata DFA!



Automata Hingga | 33

Bab 5: Otomata (Automata) Hingga

Agenda.

- Otomata (Automata) Hingga
 - Deterministic Finite Automata – DFA (Otomata Hingga Deterministik)
 - Equivalen 2 DFA
 - Finite State Machine – FSA (Mesin Stata Hingga)
- Non-Deterministic Finite Automata – NFA (Otomata Hingga Non-Deterministik)
- NFA to DFA Conversion
- DFA to Grammar Conversion
- Grammar to NFA Conversion
- Ekuivalensi NFA – RE (Regular Expression)

Automata Hingga | 34

Otomata (Automata) Hingga

DFA to Grammar Conversion.

- Contoh kasus:** Diketahui sebuah DFA sebagai berikut $F(K, V_T, M, S, Z)$, akan dibentuk GR $G = (V_T', V_N, S', Q)$ dimana:

$K = \{S, A, B, C\}$ M diberikan dalam

$V_T = \{0, 1\}$ tabel berikut:

$S = S$

$Z = \{S\}$

K	0	1
S	B	A
A	C	S
B	S	C
C	A	B

Automata Hingga | 35

Otomata (Automata) Hingga

Algoritma konversi:

- Tetapkan: $V_T' = VT$, $S' = S$, $V_N = K$
- Jika $A_p, A_q \in K$ dan $a \in VT$, maka:

$M(A_p, a) = A_q$ ekuivalen dengan produksi:

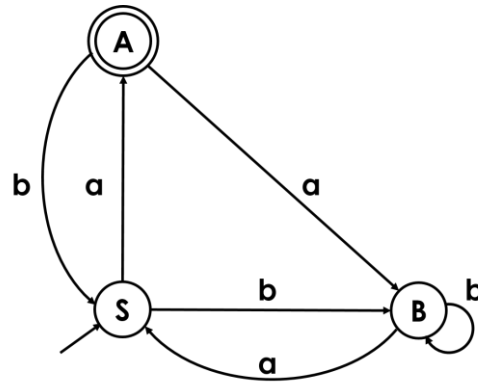
 - $A_p \rightarrow aA_q$, jika $A_q \notin Z$
 - $A_p \rightarrow a$, jika $A_q \in Z$

Automata Hingga | 36

Otomata (Automata) Hingga

Latihan 7.

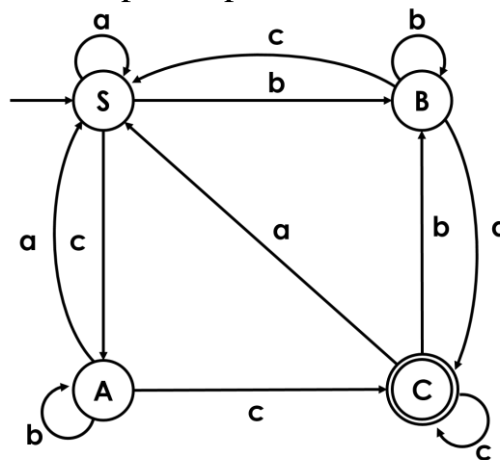
Problem 1: Tentukan himpunan produksi otomata DFA berikut!



Automata Hingga | 37

Otomata (Automata) Hingga

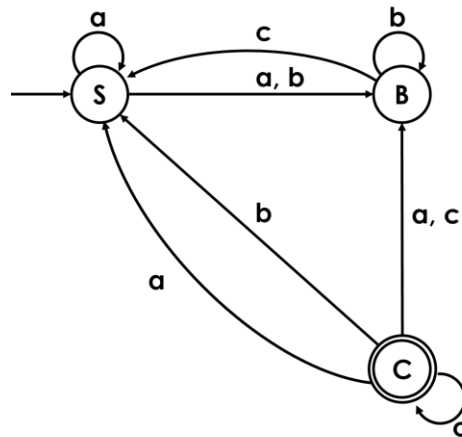
Problem 2: Tentukan himpunan produk GR X untuk otomata DFA X berikut!



Automata Hingga | 38

Otomata (Automata) Hingga

Problem 3: Tentukan himpunan produk GR Q untuk otomata berikut!



Automata Hingga | 39

Otomata (Automata) Hingga

Grammar to NFA Conversion.

Algoritma konversi grammar ke AHN:

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

Automata Hingga | 40

Otomata (Automata) Hingga

Contoh kasus:

Diketahui sebuah grammar $G = (V_T, V_N, S, Q)$ akan dibentuk $F = (K, V_T', M, S', Z)$, dimana $V_T = \{a, b\}$, $V_N = \{S, A, B\}$, $S = S$, dan $Q = \{S \rightarrow aS, S \rightarrow bA, A \rightarrow aA, A \rightarrow aB, B \rightarrow b\}$

Latihan 8.

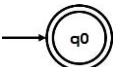

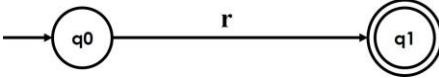
Problem 1: Tentukan $F = (K, V_T', M, S', Z)$, jika diketahui sebuah grammar $G = (V_T, V_N, S, Q)$ memiliki $V_T = \{a, b\}$, $V_N = \{S, A, B\}$, $S = S$, dan $Q = \{S \rightarrow aS, S \rightarrow bA, A \rightarrow a, A \rightarrow aB, B \rightarrow bS, B \rightarrow bB, B \rightarrow b\}$

Problem 2: Gambarkan NFA F diatas!

Automata Hingga | 41

Otomata (Automata) Hingga

Ekuivalensi NFA – RE (Regular Expression).

Jenis ER	Simbol ER	NFA
Simbol Hampa	ε	
ER Hampa	ϕ atau $\{\}$	
ER Umum	r	

Automata Hingga | 42

Otomata (Automata) Hingga

Jenis ER	Simbol ER	NFA
Alternation	$r1 \mid r2$	
Concatenation	$r1 r2$	
Kleene Closure	r^*	

Automata Hingga | 43

Otomata (Automata) Hingga

Contoh kasus:

Tentukan NFA untuk ekspresi regular $r = 0(1|23)^*$

Latihan 9.

Problem 1: Tentukan NFA untuk ekspresi regular $r = 0 \mid (23)^*$!

Automata Hingga | 44

