

Teori Bahasa & OTOMATA

BAB I. PENDAHULUAN

A. KEDUDUKAN TEORI BAHASA DAN OTOMATA PADA ILMU KOMPUTER

Ilmu komputer mempunyai 2 komponen utama :

- Model dan gagasan mendasar mengenai komputasi.
- Teknik rekayasa untuk perancangan sistem komputasi, meliputi perangkat keras dan perangkat lunak, khususnya penerapan rancangan dari teori.

Teori bahasa dan otomata merupakan bagian pertama.

Secara teoritis ilmu komputer diawali dari sejumlah perbedaan disiplin ilmu.

- **Teknik elektro:** Mengembangkan switching sebagai tool untuk mendesain hardware.
- **Matematika :** Bekerja berdasarkan logika.
- **Ahli Bahasa :** Menyelidiki tata bahasa untuk natural language.
- **Ahli Biologi :** Mempelajari neural network.

Spesifikasi dari sebuah bahasa pemrograman :

- Himpunan simbol-simbol (alphabet) yang bisa dipakai untuk membentuk program yang benar.
- Himpunan program yang benar secara sintaktik.
- Makna dari program tersebut.

B. Konsep Bahasa dan Otomata

1. Teori Bahasa adalah konsep-konsep pada "string alfabeta" dalam penyambungan karakter-karakter alfabeta untuk membentuk suatu makna (bahasa).

2. Alfabet adalah himpunan simbol (karakter) tak kosong yang berhingga. Alfabet digunakan untuk membentuk kata-kata (string-string) di bahasa. Bahasa dimulai dengan alfabet. Alfabet dilambangkan dengan Σ

3. String adalah deretan simbol dari alfabet dimana pengulangan simbol diijinkan.

Contoh :

$$V = \{a,b,c,d\}$$

String pada alfabet V antara lain \rightarrow 'a','abcd','bbba'

4. panjang string adalah jumlah simbol di dalam string bukan pada alfabet dan pengulangan kemunculan simbol dihitung. Panjang string dilambangkan **$|w|$**

Contoh:

$$|\epsilon| = 0$$

$$|a| = 1$$

$$|aa| = 2$$

$$|aaa| = 3$$

$$|aaab| = 4$$

5. *Empty string(null string)* adalah string yang tidak mengandung simbol apapun. Lambangnya ϵ atau λ

6. *Regular expression* adalah cara untuk mengekspresikan bahasa dengan hanya menggunakan operasi :

- Concatenation
- Superscript
- Kleene closure
- Positif closure

Penyambungan (Concatenation - o)

Penyambungan dilakukan pada 2 karakter atau lebih membentuk 1 barisan karakter (string simbol).

Contoh :

'a' o 'b' = 'ab'

'ab' o 'baab' = 'abbaab'

Superscript

Penyambungan dapat dianggap sebagai perkalian karena biasanya penulisannya adalah bila x dan y string, maka $x \circ y$ adalah xy . sehingga pemangkatan dapat digunakan

$$VoV = VV = V^2 \text{ ----> Panjang string} = 2$$

Kleene closure

$$V^* = \{\epsilon\} \cup V^+$$

adalah string pada V , termasuk string kosong dimana ϵ string kosong (string tanpa simbol)

ϵ mempunyai sifat identitas, yaitu:

$$\epsilon \circ x = x$$

$$x \circ \epsilon = x$$

Positive closure

$$V^+ = V^1 \cup V^2 \cup V^3 \cup \dots$$

adalah himpunan string pada V , tidak ada string kosong didalamnya.

$$V^0 = \{\epsilon\}$$

adalah himpunan yang isinya hanya string kosong, dimana String kosong ϵ tidak sama dengan himpunan kosong

- Otomata merupakan suatu sistem yang terdiri atas sejumlah berhingga state, dimana state menyatakan informasi mengenai input yang lalu, dan dapat pula dianggap sebagai memori mesin.
- Input pada mesin otomata dianggap sebagai bahasa yang harus dikenali oleh mesin.
- Selanjutnya mesin otomata membuat keputusan yang mengindikasikan apakah input itu diterima atau tidak.
- Sebuah string input diterima bila mencapai state akhir / final state yang digambarkan dengan lingkaran ganda.

C. Hirarki Chomsky

- Tata bahasa (grammar) bisa didefinisikan secara formal sebagai kumpulan dari himpunan-himpunan variabel, simbol-simbol terminal, simbol awal yang dibatasi oleh aturan-aturan produksi.

- Aturan produksi merupakan pusat dari tata bahasa, yang menspesifikasikan bagaimana suatu tata bahasa melakukan transformasi suatu string ke bentuk lainnya.
- Semua aturan produksi dinyatakan dalam bentuk : “ $\alpha \rightarrow \beta$ ” (alpha menghasilkan betha atau alpha menurunkan betha)
- α menyatakan simbol-simbol pada ruas kiri aturan produksi.
- β menyatakan simbol-simbol pada ruas kanan aturan produksi

- ***Simbol variabel / non terminal*** adalah simbol yang masih bisa diturunkan lagi dan dinyatakan dengan **huruf besar**.
- ***Simbol terminal*** sudah tidak bisa diturunkan lagi, dan dinyatakan dengan **huruf kecil**.
- Dengan menerapkan aturan produksi, suatu tata bahasa bisa menghasilkan sejumlah **string**.
- Himpunan semua string adalah bahasa yang didefinisikan oleh tata bahasa tersebut.

Contoh Aturan Produksi

➤ $T \rightarrow a$

dibaca “T menghasilkan a”

➤ $E \rightarrow T \mid T + E$

dibaca “E menghasilkan T” **atau**
“E menghasilkan T dan E”

Simbol $|$ menyatakan ‘atau’, digunakan untuk mempersingkat penulisan aturan produksi yang mempunyai ruas kiri yang sama.

Penggolongan empat tingkatan bahasa berdasarkan *hirarki Comsky* dapat dilihat pada tabel berikut :



Bahasa	Mesin Otomata	Batasan Aturan Produksi.
Reguler Type 3	Finite State Automata , meliputi : <ul style="list-style-type: none"> • DFA • NFA 	α adalah sebuah simbol variabel . β maksimal memiliki sebuah simbol variabel yang bila ada terletak di posisi paling kanan, boleh tidak ada .
Bebas Konteks (Context Free) Type 2	Push Down Automata	α adalah sebuah simbol variabel .
Context Sensitive Type 1	Linier Bounded Automata	$ \alpha \leq \beta $
Unrestricted (Phase Structure) Natural Language Type 0	Mesin Turing	Tidak ada batasan.

Tipe 0 /Unrestricted /Natural Language

Aturan :

- Simbol pada Sebelah kiri harus minimal ada sebuah simbol variabel

Contoh :

Abcdef \rightarrow g (Diterima)

aBCdE \rightarrow GHIJKL (Diterima)

abcdef \rightarrow GHIJKL (Ditolak, karena simbol pada sebelah kiri tidak ada sebuah simbol variabel)

Tipe 0 /Unrestricted /Natural Language

Tentukan apakah produksi-produksi berikut memenuhi aturan tata bahasa Regular

1. $A \rightarrow b$
2. $B \rightarrow bdB$
3. $B \rightarrow C$
4. $B \rightarrow bC$
5. $B \rightarrow Ad$
6. $B \rightarrow bcdef$
7. $B \rightarrow bcdefg$
8. $A \rightarrow aSa$
9. $A \rightarrow aSS$
10. $A \rightarrow \epsilon$

Tipe 1/ Contexts Sensitive

- Panjang string pada ruas kiri \leq panjang string pada ruas kanan $|\alpha| \leq |\beta|$.

Misal :

$Ab \rightarrow DeF$

$CD \rightarrow eF$

exception : $S \rightarrow \epsilon$

CSG (Tipe 1) \rightarrow semua aturan harus memenuhi $|\beta| \geq |\alpha|$, kecuali aturan khusus:

- $S \rightarrow \epsilon$ (jika S = start symbol, dan S tidak pernah muncul di sisi kanan produksi).

Jadi, ϵ hanya boleh dipakai untuk start symbol dengan syarat tersebut.

Tipe 1/ Contexts Sensitive

Tentukan apakah produksi-produksi berikut memenuhi aturan tata bahasa *context sensitive*.

1. $B \rightarrow bcdefG$
2. $A \rightarrow aSa$
3. $A \rightarrow aSS$
4. $A \rightarrow BCDEF$
5. $Ad \rightarrow dB$
6. $A \rightarrow \epsilon$
7. $AB \rightarrow \epsilon$
8. $ad \rightarrow b$
9. $ad \rightarrow \epsilon$
10. $abC \rightarrow DE$
11. $abcDef \rightarrow ghijkl$
12. $AB \rightarrow cde$

Tipe 2 / Bebas Konteks/ Context Free

- Ruas kiri harus tepat satu simbol variabel

Misal :

$B \rightarrow CDeFG$

$D \rightarrow BcDe$

Tipe 2 / Bebas Konteks/ Context Free

Tentukan apakah aturan produksi-produksi berikut memenuhi aturan tata bahasa bebas konteks.

1. $A \rightarrow aSa$
2. $A \rightarrow Ace$
3. $A \rightarrow ab$
4. $A \rightarrow \epsilon$
5. $B \rightarrow bcdef$
6. $B \rightarrow bcdefG$
7. $A \rightarrow aSa$
8. $A \rightarrow aSS$

Tipe 3/Reguler

- Ruas kanan **maksimal** memiliki **sebuah simbol variabel** yang terletak di paling kanan, simbol terminal bisa berapa saja/ tak terbatas, tetapi bila terdapat simbol variabel harus terletak paling kanan.

Misal :

$$\begin{aligned} A &\rightarrow e \\ A &\rightarrow fgh \\ A &\rightarrow eH \\ C &\rightarrow D \end{aligned}$$

Tipe 3/Reguler

Tentukan apakah produksi-produksi berikut memenuhi aturan tata bahasa Regular

1. $A \rightarrow b$
2. $B \rightarrow bdB$
3. $B \rightarrow C$
4. $B \rightarrow bC$
5. $B \rightarrow Ad$
6. $B \rightarrow bcdef$
7. $B \rightarrow bcdefg$
8. $A \rightarrow aSa$
9. $A \rightarrow aSS$
10. $A \rightarrow \epsilon$

Catatan :

- **Aturan produksi seperti :**

$$\epsilon \rightarrow Abd$$

bukan aturan produksi yang legal, karena simbol ϵ tidak boleh berada pada ruas kiri

- **Aturan produksi yang ruas kirinya hanya memuat simbol terminal saja, seperti :**

$$a \rightarrow bd$$

$$ab \rightarrow bd$$

bukan aturan produksi yang legal, karena ruas kiri juga harus memuat simbol yang bisa diturunkan.