



PROGRAM STUDI
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

MATA KULIAH
Otomata dan teori Bahasa

Pertemuan ke_2

Pengantar Teori bahasa

Tim pengampu





2022

Capaian Pembelajaran

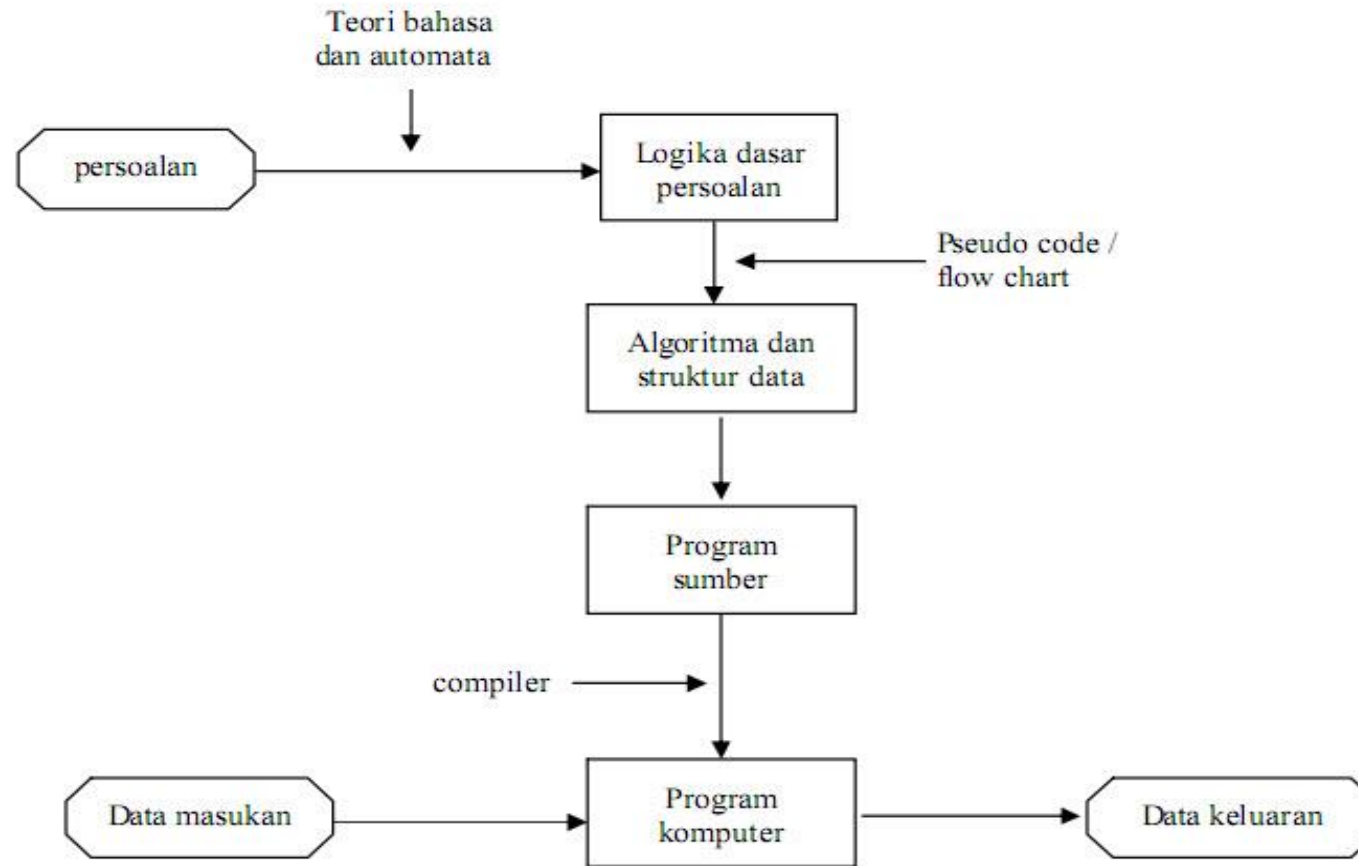
Mahasiswa memahami tipe tipe Bahasa
dan menganalisa tipe tipe bahasa



Contents

-  Definisi dan Pengertian.....●
-  Kedudukan OTB.....●
-  Tata Bahasa.....●
-  Klasifikasi Tata Bahasa.....●

Otomata & Teori Bahasa?



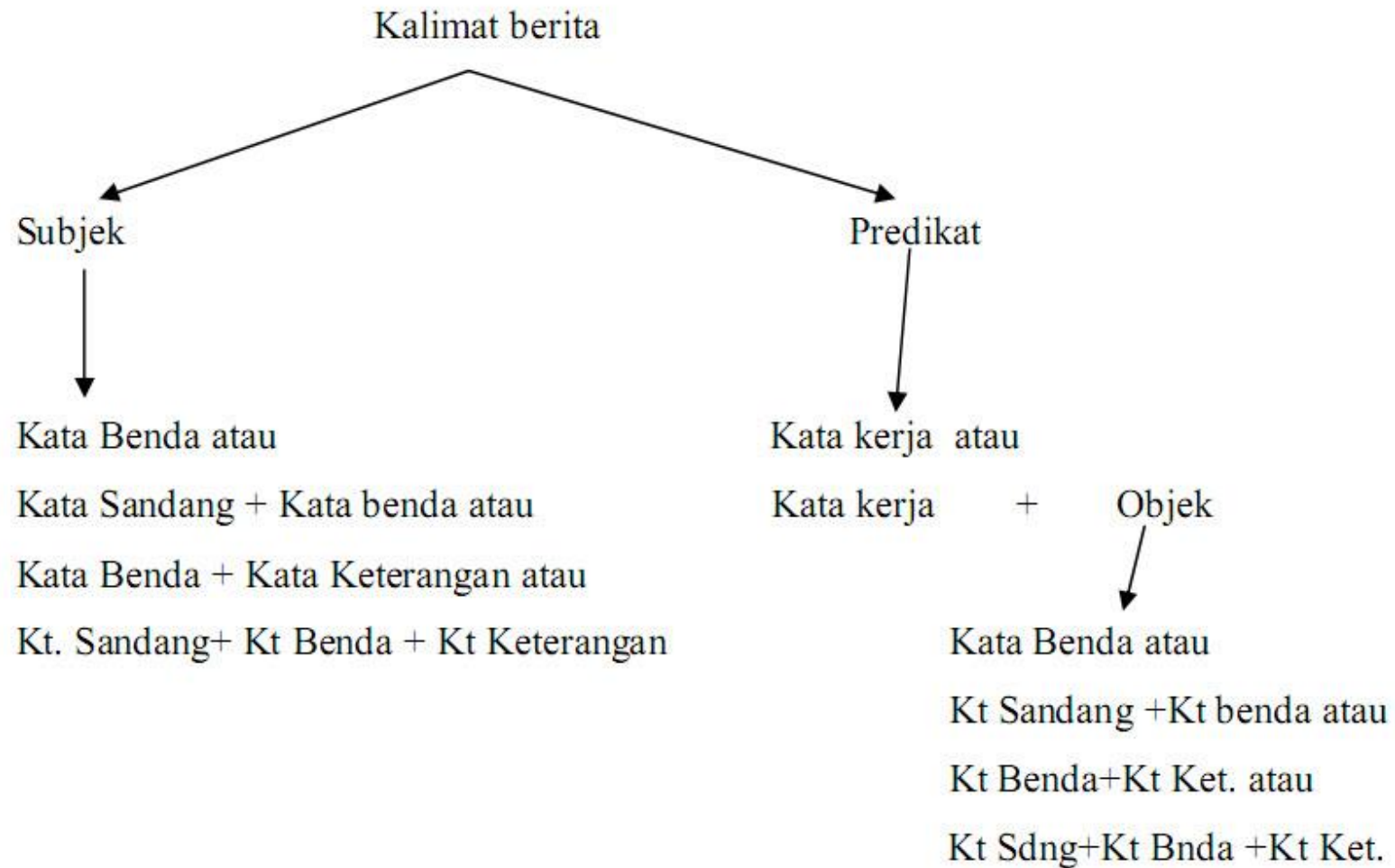
Kedudukan OTB

- Ilmu komputer memiliki dua komponen utama:
 - Pertama: model dan gagasan mendasar mengenai komputasi
 - Kedua: teknik rekayasa untuk perancangan sistem komputasi, meliputi perangkat keras dan perangkat lunak, khususnya penerapan rancangan dari teori.
- Otomata dan Teori Bahasa merupakan bagian pertama

Tata Bahasa

- Penulisan suatu kalimat dalam sebuah bahasa, akan mengikuti suatu aturan tertentu yang berlaku pada bahasa tersebut
- Aturan tersebut dikenal sebagai Tata Bahasa (Grammar)

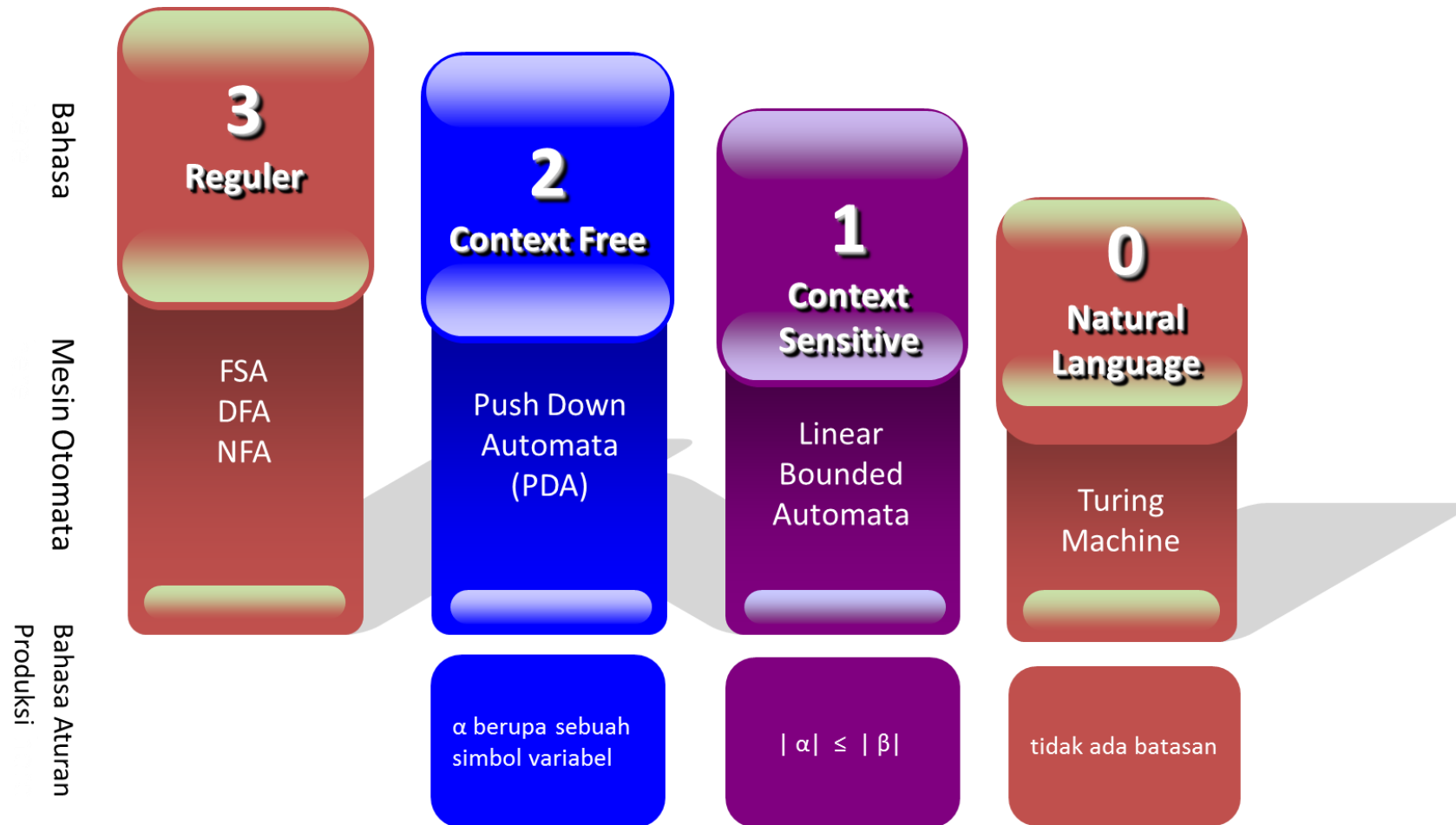
Tata Bahasa



Klasifikasi Tata Bahasa

- Tata bahasa (grammar): kumpulan dari himpunan-himpunan variabel, simbol-simbol terminal, simbol awal yang dibatasi oleh aturan-aturan produksi.
- Pada tahun 1959 seorang ahli bernama Noam Chomsky melakukan penggolongan tingkatan bahasa menjadi empat, yang disebut dengan Hirarki Chomsky.

Klasifikasi Tata Bahasa



Konsep Bahasa

- Simbol
- Abjad/alfabet
- String/kata/untai
- String kosong
- Bahasa (Language)
- Bahasa Kosong
- Bahasa Universal dari Σ



Konsep Bahasa dalam Teori Otomata

Alphabet (Abjad)

- Sebuah alphabet adalah himpunan berhingga dan tak kosong dari simbol. Alphabet disimbolkan oleh Σ .
- Contoh:
 - $\Sigma = \{0, 1\}$ alphabet biner
 - $\Sigma = \{a, b, \dots, z\}$, himpunan semua huruf kecil.
 - Himpunan semua karakter ASCII.

Konsep dalam Teori Otomata (lanjutan)

String

- Sebuah string (atau *word*) adalah deretan simbol berhingga yang dipilih dari alphabet.
- Contoh : 011011 dan 1111 adalah string dari alphabet biner $\Sigma = \{0, 1\}$.
- String kosong adalah string dimana tidak ada kemunculan simbol. (String tersebut dinotasikan oleh ε).

Konsep dalam Teori Otomata (lanjutan)

- Panjang dari string adalah banyaknya posisi untuk simbol dalam string.
- Contoh, 01101 memiliki panjang 5.
- Umumnya panjang dari string adalah banyaknya simbol dalam string.
- Pernyataan tersebut tidak sepenuhnya benar, sebagai contoh terdapat 2 simbol dalam string 01101 yaitu 0 dan 1, tetapi terdapat 5 posisi untuk simbol, dan panjangnya adalah 5.
- Notasi standar untuk panjang string w adalah $|w|$. Contoh: $|011| = 3$ dan $|\epsilon| = 0$.

Konsep dalam Teori Otomata (lanjutan)

- x adalah sebuah substring dari string lain y jika ada string w dan z , keduanya dapat berupa string kosong, sedemikian sehingga $y = wxz$.
- Sebagai contoh, car adalah substring dari $carry$, car , $vicar$.

Konsep dalam Teori Otomata (lanjutan)

Pangkat dari Alphabet

- Jika Σ adalah alphabet, dapat dinyatakan himpunan dari semua string dengan panjang tertentu dari alphabet tersebut dengan menggunakan notasi eksponensial.
- Kita mendefinisikan Σ^k sebagai himpunan dari string dengan panjang k , setiap string tersebut memiliki simbol dalam Σ .

Konsep dalam Teori Otomata (lanjutan), Contoh 1

- Perhatikan bahwa $\Sigma^0 = \{\varepsilon\}$, untuk alphabet apapun.
- Bahwa ε adalah string yang memiliki panjang 0. Jika $\Sigma = \{0, 1\}$ maka
$$\begin{aligned}\Sigma^1 &= \{0, 1\} \\ \Sigma^2 &= \{00, 01, 10, 11\} \\ \Sigma^3 &= \{000, 001, 010, 011, 100, 101, 110, 111\} \text{ dan seterusnya.}\end{aligned}$$
- Semua string pada alphabet Σ dinotasikan Σ^* .
- Contoh: $\{0,1\}^* = \{\varepsilon, 0, 1, 00, 01, 10, 11, 000, \dots\}$ dan $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \dots$

Konsep dalam Teori Otomata (lanjutan)

- Kadang-kadang kita tidak ingin memasukkan string kosong dalam himpunan string.
- Himpunan string-string tak kosong dari alphabet Σ dinotasikan Σ^+ .
- Dengan demikian :
$$\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \dots$$
$$\Sigma^* = \Sigma^+ \cup \{\epsilon\}.$$

Konsep dalam Teori Otomata (lanjutan)

Perangkaian String (*concatenation*)

- Misalkan x dan y adalah string, maka xy menyatakan perangkaian dari x dan y , bahwa string dibentuk dengan membuat salinan dari x dan diikuti oleh salinan dari y .
- Jika x adalah string yang disusun oleh i simbol,
$$x = a_1 a_2 \dots a_i$$

dan y adalah string yang disusun oleh j simbol,
$$y = b_1 b_2 \dots b_j$$

maka xy adalah string dengan panjang $i + j$,
$$xy = a_1 a_2 \dots a_i b_1 b_2 \dots b_j$$
- Contoh: $x = 01101$ dan $y = 110$, maka $xy = 01101110$ dan $yx = 11001101$.
- Untuk suatu string w , persamaan $\varepsilon w = w\varepsilon = w$ dipenuhi. Bahwa ε adalah identitas untuk perangkaian.

Konsep dalam Teori Otomata (lanjutan)

Bahasa

- Himpunan string-string yang semuanya dipilih dari Σ^* , dimana Σ adalah alphabet, dan $L \subseteq \Sigma^*$, maka L adalah bahasa pada Σ .
- Perhatikan bahwa bahasa pada Σ tidak harus meliputi string-string dengan semua simbol dari Σ .
- Dengan demikian, jika L adalah bahasa pada Σ , diketahui bahwa L adalah bahasa pada alphabet yang merupakan superset dari Σ .
- Bahasa umum dapat dipandang sebagai himpunan dari string.

Konsep dalam Teori Otomata (lanjutan), Contoh 2

- Bahasa Inggris, yang merupakan koleksi dari kata-kata dalam bahasa Inggris yang benar.
 - Kata-kata tersebut merupakan string pada alphabet yang mengandung semua huruf.
- Bahasa C atau bahasa pemrograman lainnya.
 - Dalam bahasa tersebut, program yang benar adalah subset dari string-string yang mungkin yang dibentuk dari alphabet.
 - Alphabet tersebut adalah subset dari karakter-karakter ASCII.

Operasi pada String

- Eksponensiasi String

$w^n =$

1. ε , jika $n=0$,
2. $w \cdot w^{n-1}$, jika $n>0$

Konsep dalam Teori Otomata (lanjutan)

Contoh bahasa dalam teori otomata: $\Sigma = \{0, 1\}$

1. Buatlah bahasa, dg aturan : semua string yang berisi n buah 0 dan diikuti oleh n buah 1, untuk $n \geq 0$:
 $L1 = \{\epsilon, 01, 0011, 000111, \dots\}$
2. Buatlah bahasa, dg aturan : semua string dari 0 dan 1 dengan banyaknya 0 sama dengan banyaknya 1.
 $L2 = \{\epsilon, 01, 10, 0011, 0101, 1001, \dots\}$
3. Buatlah bahasa, dg aturan : Himpunan bilangan biner yang memiliki nilai prima
 $L3 = \{10, 11, 101, 111, 1011, \dots\}$
4. Σ^* adalah bahasa universal untuk alphabet Σ
5. ϕ adalah bahasa kosong (bahasa yang tidak memiliki anggota string), merupakan bahasa pada suatu alphabet.
6. $\{\epsilon\}$, bahasa yang hanya mengandung string kosong, juga merupakan bahasa pada suatu alphabet. Perhatikan bahwa $\phi \neq \{\epsilon\}$

Contoh 3

Berikut adalah contoh bahasa pada $\Sigma = \{a, b\}$:

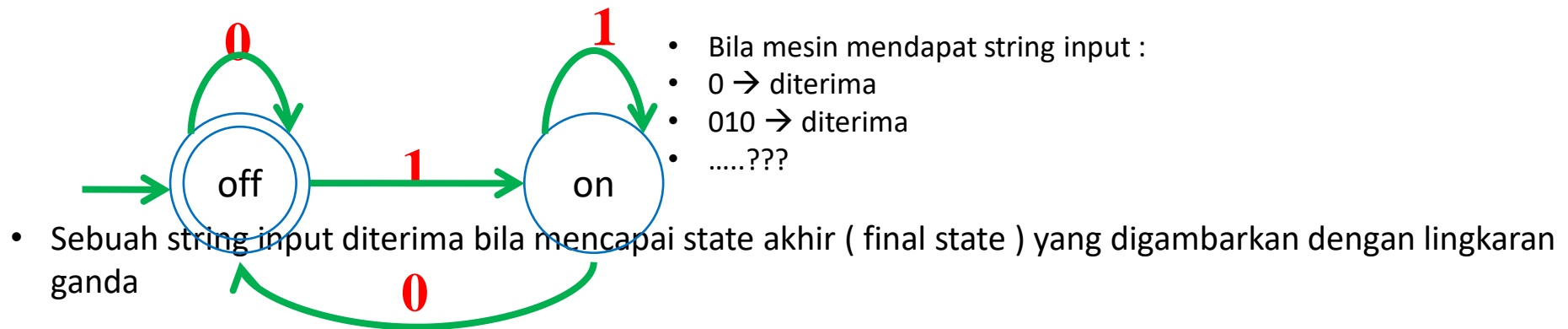
1. $L_1 = \{\varepsilon, a, aa, aab\}$
2. $L_2 = \{x \in \{a, b\}^* \mid |x| \leq 8\}$
3. $L_3 = \{x \in \{a, b\}^* \mid |x| \text{ adalah ganjil}\}$
4. $L_4 = \{x \in \{a, b\}^* \mid n_a(x) \geq n_b(x)\}$
5. $L_5 = \{x \in \{a, b\}^* \mid |x| \geq 2, x \text{ diawali dan diakhiri dengan } b\}$

Perangkaian Bahasa

- Jika L_1 dan L_2 adalah bahasa, L_1 dan $L_2 \subseteq \Sigma^*$. Perangkaian dari L_1 dan L_2 dinotasikan $L_1L_2 = \{xy \mid x \in L_1 \text{ dan } y \in L_2\}$.
- $L_1 = \{\text{hope, fear}\}$
- $L_2 = \{\text{less, fully}\}$
- Sebagai contoh, $L_1 \cdot L_2 = \{\text{hope, fear}\}\{\text{less, fully}\} = \{\text{hopeless, hopefully, fearless, fearfully}\}$.
- Untuk L adalah bahasa, $L\{\varepsilon\} = \{\varepsilon\}L$ karena untuk setiap $x \in L$, $x\varepsilon = \varepsilon x = x$.

Otomata Sederhana

- Contoh mesin otomata sederhana :
 - Otomata pada saklar listrik



Otomata Sederhana

- Finite automaton berikut dapat dinyatakan sebagai bagian dari lexical analyzer.

