



**PROGRAM STUDI**  
**TEKNIK INFORMATIKA**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS DIAN NUSWANTORO**

MATA KULIAH  
*Otomata dan teori Bahasa*

# Pertemuan ke\_1 PENGANTAR OTOMATA & TEORI BAHASA

*Tim pengampu*

2022

## Referensi

- Firrar Utdirartatmo, *Teori Bahasa dan Otomata*, JJ Learning Yogyakarta, 2001
- Bambang Hariyanto, *Teori Bahasa, Otomata, dan Komputasi serta Terapannya*, Informatika Bandung, 2004
- Dean Kelley, *Otomata dan Bahasa-bahasa Formal*, PT. Prenhallindo, Jakarta, 1999
- pppps: Slide bukan referensi atau materi XD

## Capaian Pembelajaran

Mahasiswa memahami konsep dan istilah yang umum digunakan dalam Teori Bahasa dan Otomata

## Indikator

Mahasiswa mendapatkan:

1. penjelasan mengenai materi yang akan dipelajari selama satu semester.
2. Penjelasan tentang referensi yang digunakan
3. Penjelasan tentang aturan perkuliahan



# MATERI PERTEMUAN

- Pengantar Otomata
- Pendahuluan
- Organisasi Materi
- Referensi
- Evaluasi



# Apa itu Otomata ?

- Otomata adalah MODEL
- Model dari sistem apapun yang akan kita komputasikan
- “Model matematika yang memiliki fungsi dari komputer digital yaitu menerima input, menghasilkan output, bisa memiliki penyimpanan sementara dan mampu membuat keputusan dalam transformasi input ke output”

Tidak ada bidang apapun dalam teknologi informasi yang tidak terkait dengan teori 'dahsyat' ini.

Semua bentuk sistem, diskrit, kontinu, bahkan hybrid (gabungan event diskrit dan kontinu dalam satu sistem) dapat dimodelkan oleh teori ini.

# PENDAHULUAN (1)

Komputasi menjadi isu penting karena mempelajari bagaimana kita dapat **merancang mesin** yang mampu melakukan proses-proses intelektual (yang mulanya hanya dapat dilakukan manusia)

Namun dalam merancang seringkali kita terkendala dengan berbagai macam batasan. Tetapi apakah benar jika batasan-batasan (yang dimiliki komputer) pada dasarnya disebabkan oleh kelemahan programmer (manusia)??? bukan batasan intrinsik yang dimiliki mesin/komputer ?!

Jika Ya, maka kita berharap agar batasan-batasan tersebut dapat tereduksi melalui pengembangan **teori komputasi**.

# PENDAHULUAN (2)

Sub bidang apapun dalam ilmu informatika pasti memiliki 2 komponen :

1. Ide/gagasan dirupakan ke dalam bentuk MODEL KOMPUTASI

- |                        |   |                    |
|------------------------|---|--------------------|
| ■ Neuron Nets          | ➔ | Finite Automata    |
| ■ Sistem Logika Formal | ➔ | Proof Methods      |
| ■ Sistem Tata Bahasa   | ➔ | Psycho-Linguistic: |

1. Apakah arti bahasa itu ?
  2. Bagaimana manusia mengembangkan bahasa ?
  3. Bagaimana manusia memahami bahasa ?
  4. Bagaimana manusia mengajarkan bahasa ke anak-anaknya ?
  5. Bagaimana cara menyatakan gagasan ?
  6. Bagaimana manusia membangun kalimat dari gagasan yang ada dalam pikirannya ?
2. **eknik rekayasa untuk mengimplementasikan model ke dalam sebuah bentuk sistem yang terkomputasi (programming/coding)**

# PENDAHULUAN (3)

**Noam Chomsky,** ▶ membuat model matematis untuk mendeskripsikan bahasa sekaligus menjawab pertanyaan ttg psycho-linguistic  
▶ membuat perangkat formal untuk memodelkan properti bahasa (disebut Grammar)

**McCulloch & Pitts,** merancang Finite Automata untuk memodelkan neuron nets

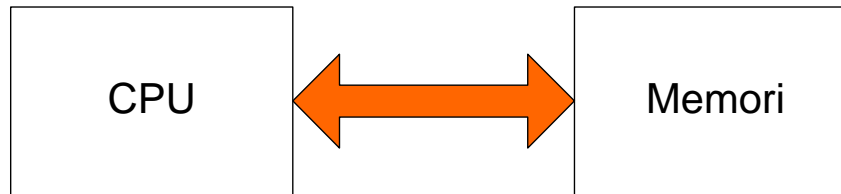
**Stephen Kleene,** menemukan model representasi lain dari automata melalui Regular Expression

**Alan Turing,** menemukan model untuk mengidentifikasi apakah sebuah permasalahan dapat dikomputasi → Mesin Turing

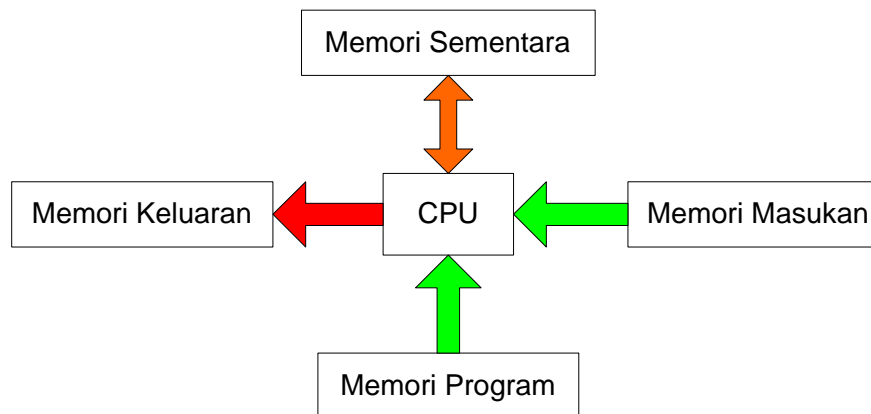


# PENDAHULUAN (4)

**Model Komputasi Awal :**



**Model Komputasi Sekarang :**

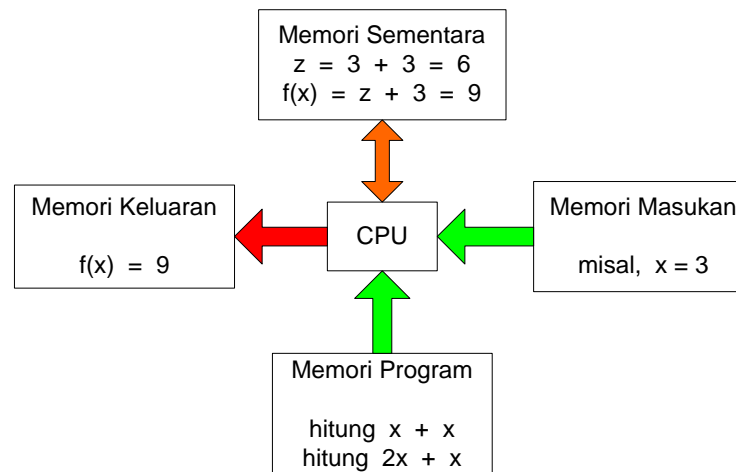
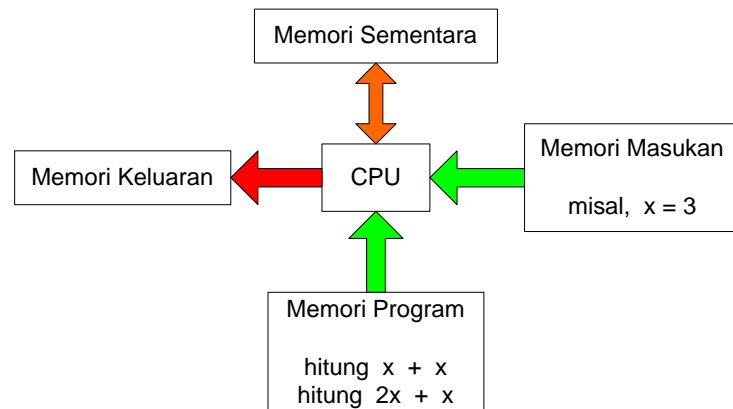
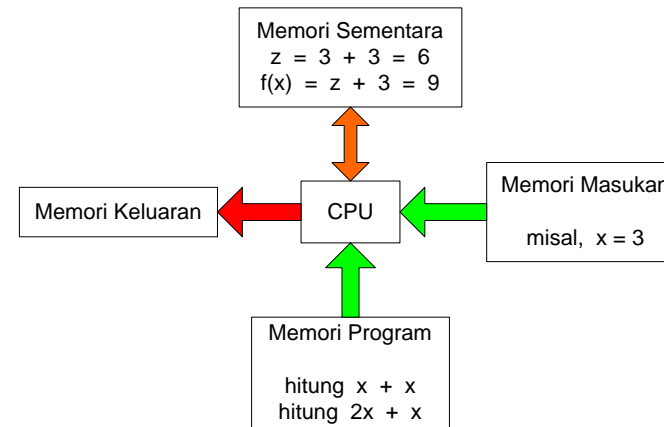
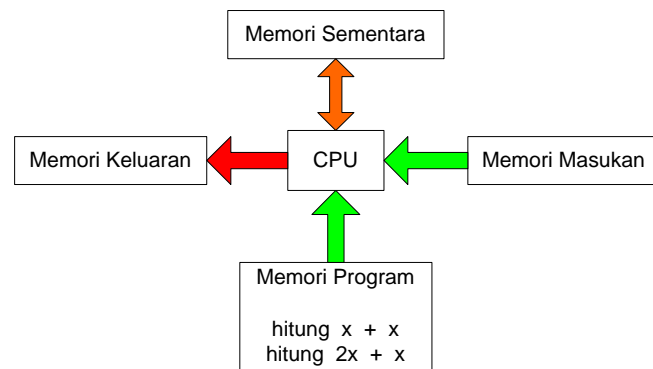


# PENDAHULUAN (5)

Bagaimana proses komputasi untuk :

$$f(x) = x + x + x$$

# PENDAHULUAN (6)



# PENDAHULUAN (7)

3 model mesin komputasi yang akan kita pelajari dalam otomata :

## 1. Finite State Automata (FSA)

(sejauh ini) telah dimanfaatkan untuk merancang lexical analyzer, aplikasi editor teks, pengenalan pola, fault tolerant system, dll

## 2. Pushdown Automata (PDA)

(sejauh ini) telah dimanfaatkan untuk mengenali bahasa yang berstruktur context- free grammar, kamus data, query, script, parsing, dll

## 3. Turing Machine (TM)

mesin turing dapat dimanfaatkan untuk mengidentifikasi ketidakmungkinan penulisan sebuah program komputer. Sejauh ini kita dapat meyakini bahwa jika suatu persoalan tidak dapat dimodelkan oleh mesin turing, maka persoalan tersebut tidak akan mungkin dapat diselesaikan secara komputatif oleh mesin komputasi apapun!

# ORGANISASI MATERI (1)

Minggu ke	Topik Bahasan	Materi Bahasan	Bentuk Perkuliahan	Referensi	Target Perkuliahan
1	Pengantar Otomata dan Teori Bahasa	<ul style="list-style-type: none"> <li>▪ Penjelasan singkat tentang Otomata serta contoh aplikasi automata pada kehidupan sehari-hari</li> <li>▪ Bahasa dan Tatabahasa Formal</li> </ul>	Paparan		Pengetahuan
2	Teori Bahasa & Operasi Matematis	<ul style="list-style-type: none"> <li>• Terminologi Bahasa</li> <li>• Operasi pada Bahasa</li> <li>• Metode Pendefinisian Bahasa</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
3	Hirarki Chomsky dan Finite State Automata	<ul style="list-style-type: none"> <li>• Hirarki Chomsky dan Contoh</li> <li>• Finite State Automata (FSA)</li> <li>• Jenis-jenis FSA dan Contoh</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
4	Deterministic Finite State Automata (DFA) dan Non Deterministic Finite State Automata (NDFA)	<ul style="list-style-type: none"> <li>• Pengertian DFA dan NDFA</li> <li>• Konversi NDFA ke DFA</li> <li>• Konversi RE ke NDFA Contoh dan Latihan</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
5	Regular Expression	<ul style="list-style-type: none"> <li>• Regular Expression</li> <li>• DFA dan NFA</li> <li>• Transition Graph</li> <li>• Automata with Output</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
6	Kleene's Theorem	<ul style="list-style-type: none"> <li>• Apa Itu Teorema Kleene ?</li> <li>• Metode Pembuktian</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
7	Aturan Produksi dari FSA	<ul style="list-style-type: none"> <li>• Apa Itu Aturan Produksi?</li> <li>• Equivalensi Grammar Regular (GR) ke FSA</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
8	UJIAN TENGAH SEMESTER (UTS)				

# ORGANISASI MATERI (2)

9	FSA dengan Output (Mesin Mealy dan Mesin Moore)	<ul style="list-style-type: none"> <li>• Mesin Mealy</li> <li>• Mesin Moore</li> <li>• Contoh-contoh</li> <li>• Ekuivalensi Meisn Mealy ke Mesin Moore</li> <li>• Ekuivalensi Meisn Moore ke Mesin Mealy</li> </ul>	Paparan & Pengerjaan Tugas		Pengetahuan
10	Tata bahasa bebas konteks /Context Free Grammar (CFG)	<ul style="list-style-type: none"> <li>• Pengertian CFG</li> <li>• Contoh-contoh CFG</li> <li>• Pohon sintaks</li> <li>• Parsing Tree</li> <li>• Ambigu</li> </ul>	Paparan & Pengerjaan Tugas		Pengetahuan
11	Penyederhanaan Context Free Grammar (CFG)	<ul style="list-style-type: none"> <li>• Tujuan Penyederhanaan</li> <li>• Penghilangan Produksi Usesless</li> <li>• Penghilangan Produksi Unit</li> <li>• Penghilangan Produksi <math>\epsilon</math></li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pengetahuan
12	Grammar & Normalisasinya	<ul style="list-style-type: none"> <li>• Transformasi Context-Free Grammar</li> <li>• Chomsky Normal Form</li> <li>• Greibach Normal Form</li> <li>• Penugasan</li> </ul>	Paparan		Pemahaman
13	Pushdown Automata (PDA)	<ul style="list-style-type: none"> <li>• Komponen PDA</li> <li>• Membentuk PDA dari CFG</li> <li>• Latihan</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
14	Turing Machine (TM)	<ul style="list-style-type: none"> <li>• Komponen Mesin Turing</li> <li>• Penugasan</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
15	Responsi	Pembahasan Tugas	Diskusi		Pemahaman
16	UJIAN AKHIR SEMESTER (UAS)				

# REFERENSI

## REFERENSI – UTAMA

- Aho, Alfred V., Sethi, R., Ulman, J.D., *Compilers : Principles, Techniques, and Tools*, Addison-Wesley Publ. Company, Reading Massachusetts, 1986
- Cohen, Daniel I.A., *Introduction to Computer Theory*, John Wiley & Sons, 1990

## REFERENSI – PENDUKUNG

- Hariyanto, Bambang, *Teori Bahasa, Otomata, dan Komputasi serta Terapannya*, Informatika, Bandung, 2004
- Kelly, Dean, *Otomata Dan Bahasa-Bahasa Formal : Sebuah Pengantar*, PT Prenhallindo, Jakarta, 1999
- Tremblay, Jean P., Sorenson, Paul G., *The Theory and Practice of Compiler Writing*, McGrawHill Book Company, New York, 1982
- Utdirartatmo, Firrar, *Teori Bahasa Dan Otomata*, J & J Learning, Yogyakarta, 2001
- Utdirartatmo Firrar, *Teknik Kompilasi*, J & J Learning, Yogyakarta, 2001

# EVALUASI

- UTS : 40%
- UAS : 40%
- TUGAS : 20%