

## PERTEMUAN I

# PENGANTAR TEORI BAHASA OTOMATA

---

Mahasiswa mengetahui tujuan mata kuliah, alur perkuliahan selama 1 semester, referensi yang digunakan, bentuk & bobot evaluasi

# MATERI PERTEMUAN

---

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

# Apa itu Otomata dan Kompilasi ?

Otomata adalah MODEL.

Model dari sistem apapun yang akan kita komputasikan.

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 'digdaya' ini.

Sementara, Kompilasi adalah ilmu yang mempelajari bagaimana kita dapat merancang & membangun bahasa pemrograman.

Kompilasi merupakan SALAH SATU bidang yang memanfaatkan teori 'sakti' 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**

Beberapa disiplin ilmu yang diadopsi :

■ 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. Teknik rekayasa untuk mengimplementasikan model ke dalam sebuah bentuk sistem yang terkomputasi (programming/coding)

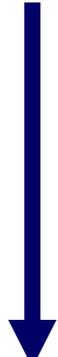
# PENDAHULUAN (3)

---

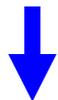
- Noam Chomsky**, ▶ membuat **model matematis** untuk mendeskripsikan bahasa sekaligus menjawab pertanyaan ttg psycho-linguistic
- ▶ membuat perangkat formal untuk memodelkan properti bahasa (disebut **Grammar**)

Open Question :

Perbedaan antara bahasa manusia dan bahasa komputer adalah kita sampai sekarang belum mengetahui bagaimana cara kita mengartikan bahasa? (sementara kita dapat mengetahui secara pasti cara komputer mengartikan bahasa)



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



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

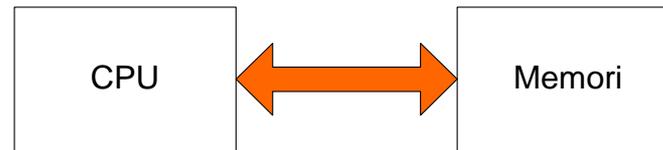


**Alan Turing**, menemukan model untuk mengidentifikasi apakah sebuah permasalahan dapat dikomputasi → **Mesin Turing**

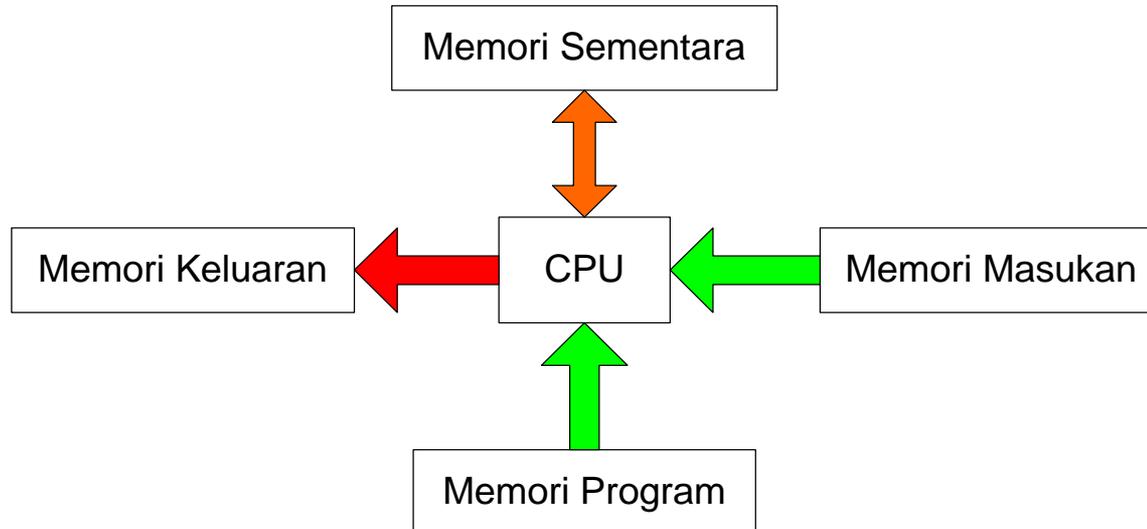
# PENDAHULUAN (4)

---

**Model Komputasi Awal :**



**Model Komputasi Sekarang :**



# PENDAHULUAN (5)

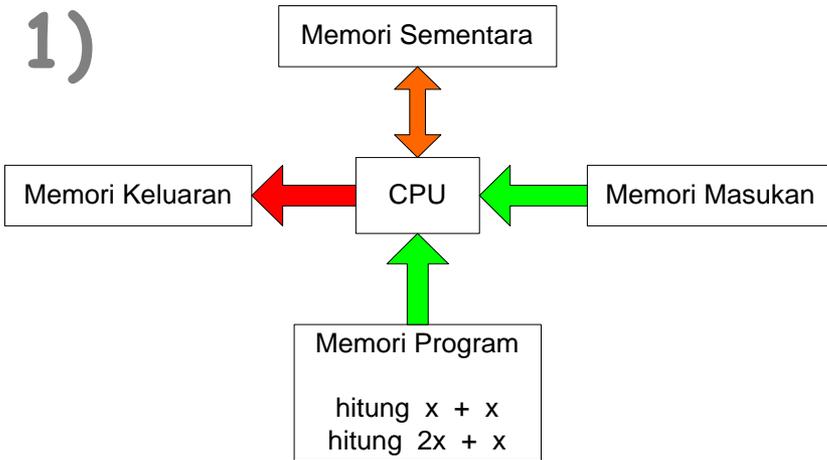
---

Bagaimana proses komputasi untuk :

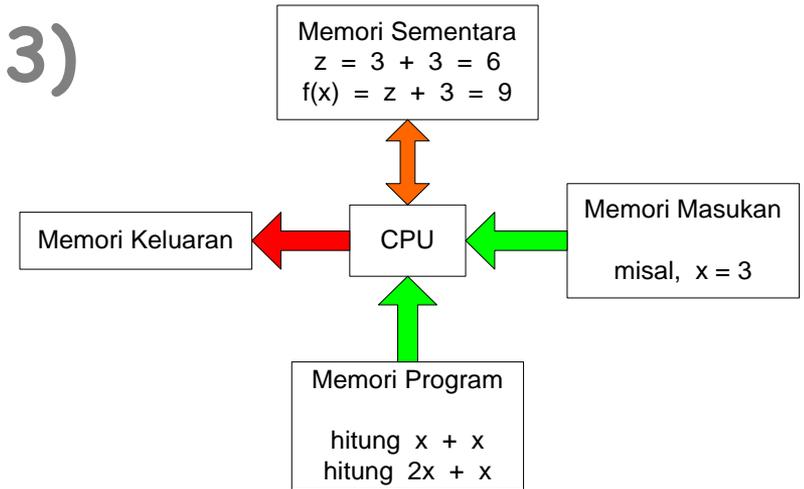
$$f(x) = x + x + x$$

# PENDAHULUAN (6)

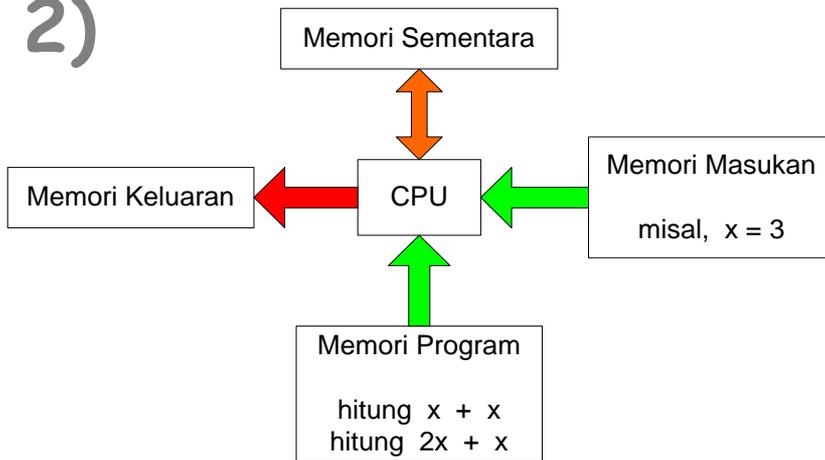
1)



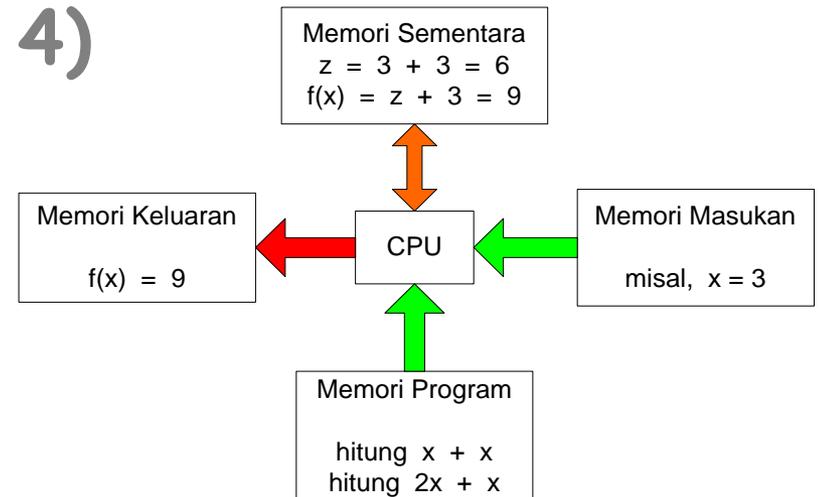
3)



2)



4)





3 model mesin komputasi yang akan kita pelajari dalam otomata :

## 1. Finite Automata (FA)

(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 Kompilasi	<ul style="list-style-type: none"> <li>• Sekilas ttg Otomata &amp; Pengantar Kompilasi</li> <li>• Organisasi Materi</li> <li>• Referensi Perkuliahan</li> <li>• Bobot Evaluasi</li> <li>• Pendahuluan</li> <li>• Template Penulisan Tugas</li> <li>• Penugasan I</li> </ul>	Paparan		Pengetahuan
2	Teori Bahasa & Operasi Matematis (1)	<ul style="list-style-type: none"> <li>• Terminologi Bahasa</li> <li>• Operasi pada Bahasa</li> <li>• Metode Pendefinisian Bahasa</li> <li>• Penugasan II</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
3	Teori Bahasa & Operasi Matematis (2)	<ul style="list-style-type: none"> <li>• Konsep Dasar Grammar</li> <li>• Derivasi &amp; Parse Tree</li> <li>• Klasifikasi Grammar</li> <li>• Penugasan III</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
4	Responsi #1	Pembahasan Tugas I, II, III	Diskusi		Pemahaman
5	Regular Expression & Finite Automata (1)	<ul style="list-style-type: none"> <li>• Regular Expression</li> <li>• Finite Automata</li> <li>• Finite State Diagram</li> <li>• Penugasan IV</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
6	Regular Expression & Finite Automata (2)	<ul style="list-style-type: none"> <li>• Deterministic Finite Automata</li> <li>• Transition Graph</li> <li>• Automata with Output</li> <li>• Penugasan V</li> </ul>	Paparan & Pengerjaan Tugas		Pemahaman
7	Kleene's Theorem	<ul style="list-style-type: none"> <li>• Apa Itu Teorema Kleene ?</li> <li>• Metode Pembuktian</li> <li>• Penugasan VI</li> </ul>	Paparan & Pengerjaan Tugas		Pengetahuan
8	Responsi #2	Pembahasan Tugas IV, V, VI	Diskusi		Pemahaman
9	<b>UJIAN TENGAH SEMESTER (UTS)</b>				

# ORGANISASI MATERI

(2)



10	Non Deterministic Finite Automata (N DFA)	<ul style="list-style-type: none"><li>• Pengertian Non Determinism</li><li>• Non Deterministic Finite Automaton (N DFA)</li><li>• Konversi N DFA ke DFA</li><li>• Konversi RE ke N DFA</li><li>• Penugasan VII</li></ul>	Paparan & Pengerjaan Tugas		Pengetahuan
11	Masalah Regularitas	<ul style="list-style-type: none"><li>• Bahasa Regular</li><li>• Observasi pada DFA</li><li>• Bahasa Non Regular</li><li>• Penugasan VIII</li></ul>	Paparan & Pengerjaan Tugas		Pengetahuan
12	Grammar & Normalisasinya	<ul style="list-style-type: none"><li>• Ambiguitas</li><li>• Penyederhanaan Context-Free Language</li><li>• Transformasi Context-Free Grammar</li><li>• Chomsky Normal Form</li><li>• Greibach Normal Form</li><li>• Penugasan IX</li></ul>	Paparan		Pemahaman
13	Pushdown Automata (PDA) & Turing Machine (TM)	<ul style="list-style-type: none"><li>• Komponen PDA</li><li>• Membentuk PDA dari CFG</li><li>• Komponen Mesin Turing</li><li>• Penugasan X</li></ul>	Paparan & Pengerjaan Tugas		Pemahaman
14	Responsi #3	Pembahasan Tugas VII, VIII, IX, X	Diskusi		Pemahaman
15	Compiling Phases Overview	<ul style="list-style-type: none"><li>• Apa Itu Compiler ?</li><li>• Language Processing System</li><li>• Compiling Phases</li><li>• Compiling Process Overview</li></ul>	Paparan		Pemahaman
16	Lexical Analysis & Syntax Analysis	<ul style="list-style-type: none"><li>• Lexical Analysis Overview</li><li>• Lexical Analysis Generator</li><li>• Studi Kasus : Lex</li><li>• The Role of Parsing</li><li>• General Parsing Methods</li><li>• Syntax Errors and Handling</li><li>• Brute-Force Approach</li><li>• Recursive-Descent Parsing</li><li>• Top-Down Parsing with Limited Backup</li><li>• Studi Kasus : LL(1) Grammar</li></ul>	Paparan & Diskusi		Pengetahuan
17	UJIAN AKHIR SEMESTER (UAS)				



## 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 : 50%
- UAS : 50%