



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

Mata Kuliah  
**Dasar Pemrograman**



# Dasar Pemrograman

TIM DASAR PEMROGRAMAN  
TEKNIK INFORMATIKA S1  
UNIVERSITAS DIAN NUSWANTORO

## Capaian Pembelajaran

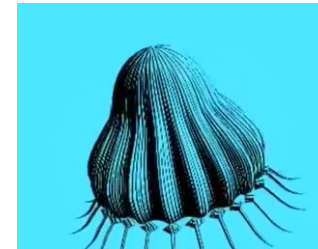
- Setelah mengikuti matakuliah ini mahasiswa dapat menjelaskan tentang teknologi informasi, berpikir komputasional, proses rekayasa perangkat lunak sederhana dengan menerapkan suatu paradigma pemrograman procedural dengan menerapkan aksi sekuensial dan
- Mempraktekkan melakukan instalasi software yang digunakan sebagai persiapan memprogram

# Apa yang Anda ketahui tentang Teknologi?

- Sosial media
- Super computer
- Game yang berkembang
- Kecerdasan buatan
- Robotika
- Roket yang bisa mendarat sendiri
- Komputer kuantum
- Augmented reality
- **Apa lagi?**



Sumber: <https://youtu.be/eBV14-3LT-g>



<https://www.cnbcindonesia.com/tech/20190222161910-37-57153/peluncuran-roket-pakai-spacex-lebih-ekonomis-semurah-apa>



Sumber: <https://youtu.be/dFoEcn3t9hc>



## Pemahaman seperti apa yang dibutuhkan?

- Teknologi? Ilmu yang berkaitan dengan pengetahuan sains / seni dengan mengaplikasikan pengetahuan saintifik ke praktis
- Aplikasi teknologi? Industri/bisnis dan sains itu sendiri
- Data
- Informasi
- sistem

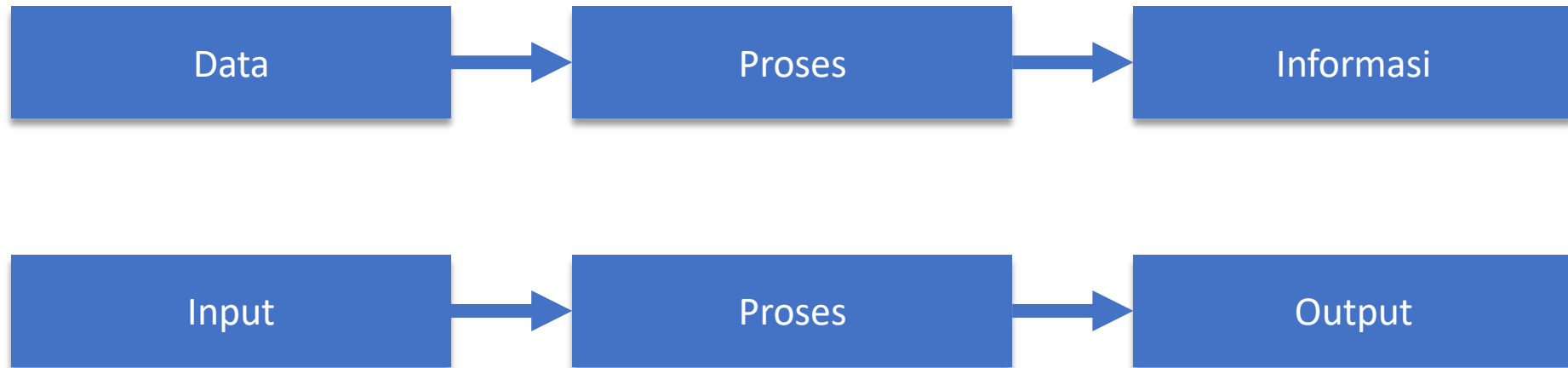
# Data

- Terdiri dari fakta mentah
- Bisa berbentuk tulisan, gambar, suara, sinyal
- Yang akan diproses menjadi suatu **informasi**

# Informasi

- Merupakan data atau kumpulan data yang sudah diolah atau di manipulasi
- Digunakan untuk menentukan suatu keputusan (decision making)
- Data vs informasi?
  - Data: imam bonjol, udinus, 50131
  - Informasi: Universitas Dian Nuswantoro (udinus) berada di jalan imam bonjol

## Struktur Sistem secara umum



- Apa isi dari proses dan kaitannya dengan komputer?



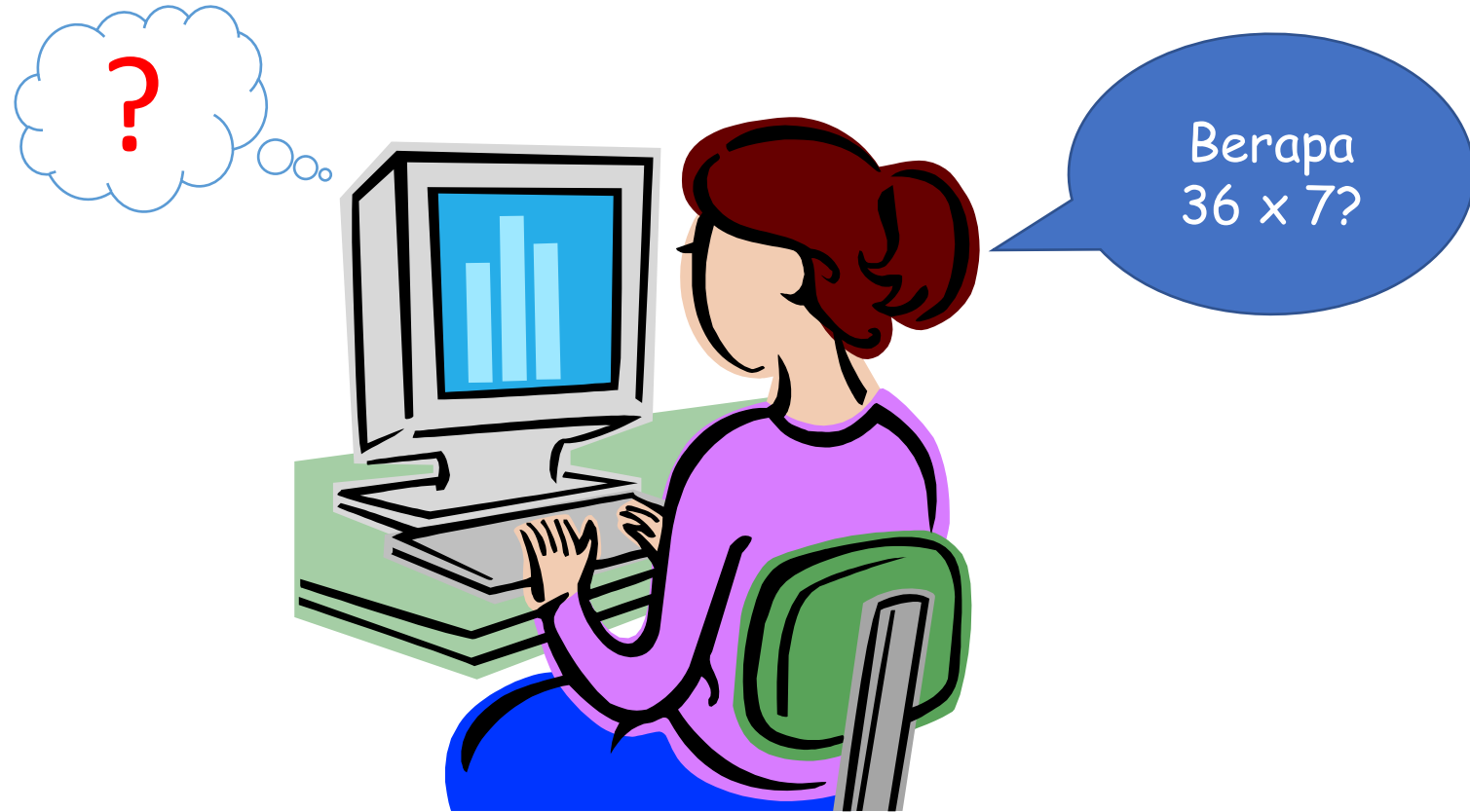
**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# Komputer



## Fungsi Komputer

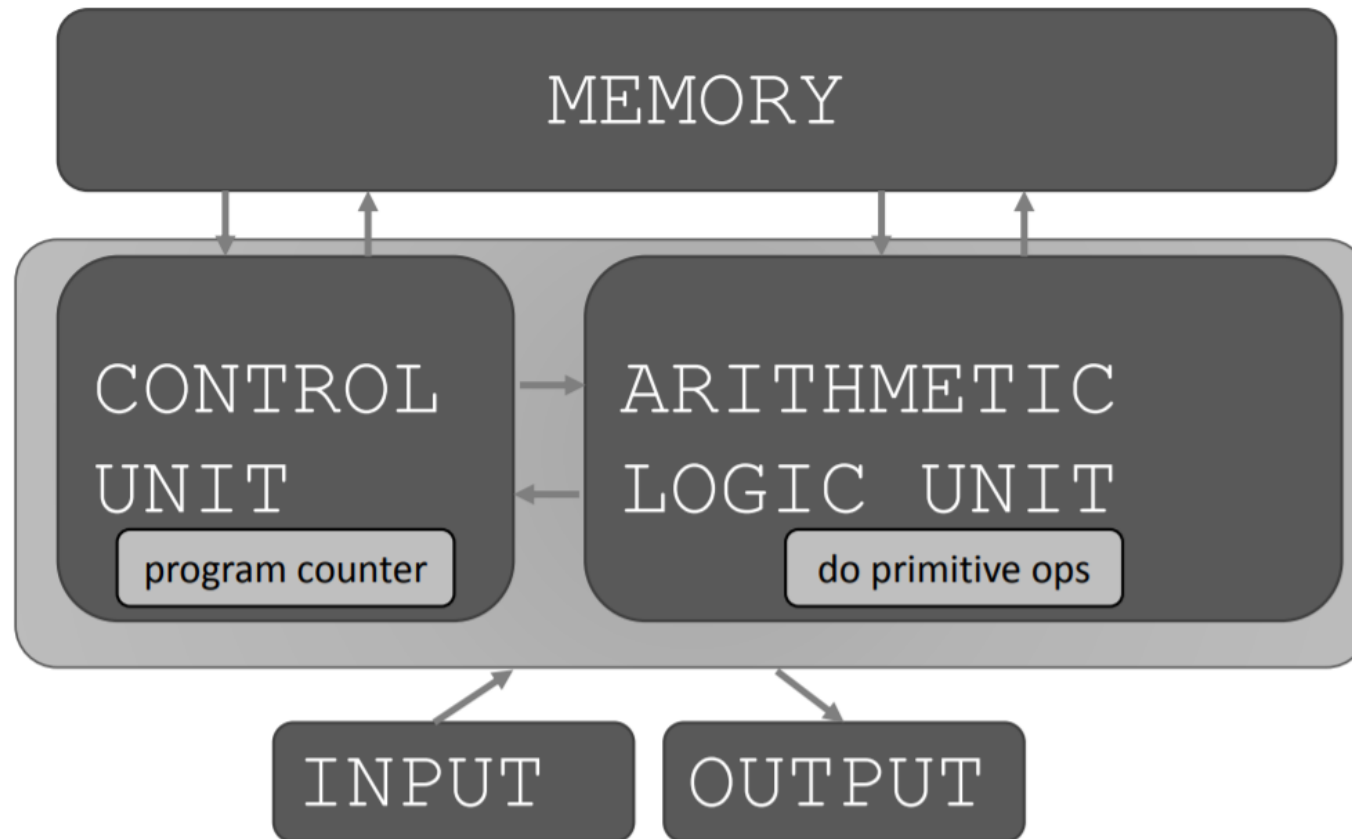
- Sebagai alat bantu menyelesaikan persoalan
- Alat elektronik yang dapat melakukan perhitungan



# Pahami Bahwa KOMPUTER = MESIN

- Bagaimana cara untuk menangkap “resep memprogram” dalam proses mekanis suatu mesin?
- Membuat program yang fixed?
  - Contoh sederhana: kalkulator
- Membuat program untuk penyimpanan (stored)
  - Mesin akan menyimpan dan mengeksekusi instruksi

# Arsitektur Dasar Mesin

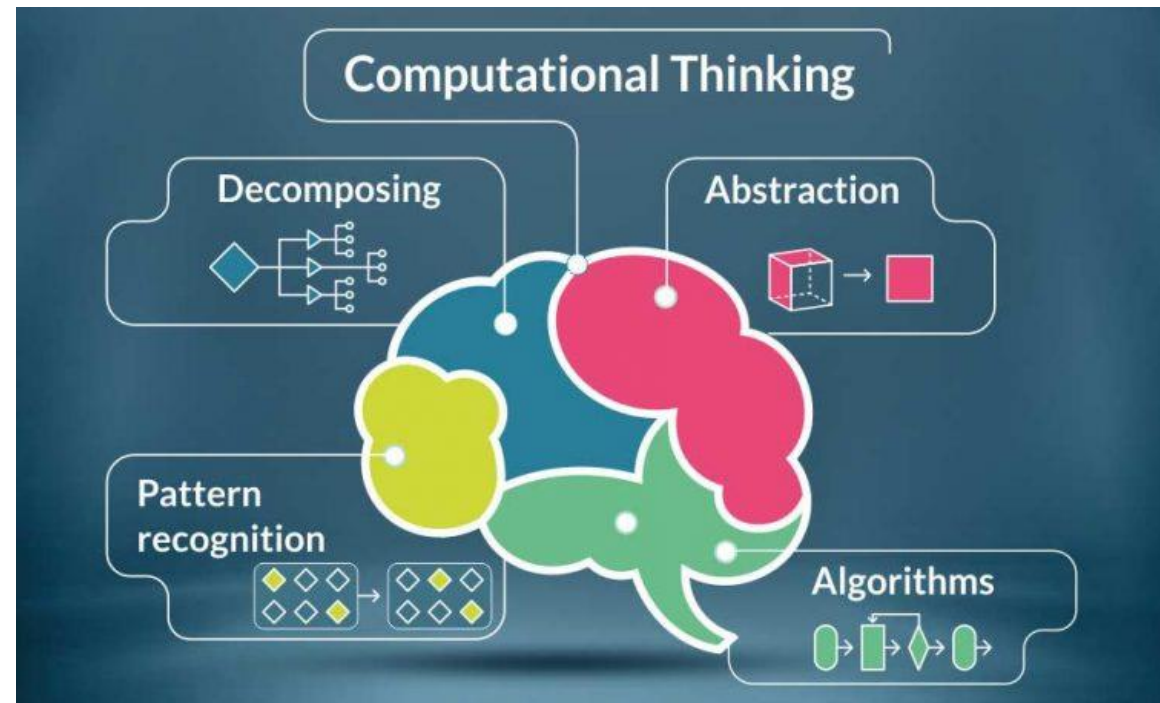


# Apa yang dilakukan Komputer

- Secara mendasar:
  - Melakukan **perhitungan**
    - 1 miliar kalkulasi per detik
  - **Mengingat** hasil
    - Penyimpanan sampai 100 GB
- **Perhitungan** macam apa?
  - Dibentuk dengan suatu **Bahasa**
  - Yang dilakukan oleh orang yang disebut sebagai **Programmer**
- **KOMPUTER HANYA TAHU APA YANG DIPERINTAHKAN KEPADANYA!**

# Berpikir Komputasional

- Suatu cara berpikir yang melibatkan sekumpulan keahlian dan Teknik pemecahan masalah yang biasanya digunakan oleh developer untuk menulis program aplikasi



# Teknik Berpikir Komputasional

- **Decomposition:** memecah – mecah masalah menjadi bagian – bagian lebih kecil dan sampai ke pokok sebuah masalah hingga kita menyelesaikan suatu masalah tersebut
- **Pattern Recognition:** mencari pola, biasanya didalam sebuah masalah terdapat pola pola tertentu untuk memecahkannya disitu kita dituntut mengetahui sendiri bagaimana pola tersebut
- **Abstraction:** melakukan generalisasi dan mengidentifikasi prinsip – prinsip umum yang menghasilkan pola, tren dan keteraturan tersebut
- **Algorithm:** mengembangkan pemecahan masalah step by step, tahapan demi tahapan untuk menyelesaikan permasalahan



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# Pemrograman

# Program Penyimpanan Komputer

- Urutan dari instruksi atau perintah tersimpan di dalam computer
  - Terdiri atas perintah – perintah dasar yang sudah didefinisikan sebelumnya
    1. Aritmatika dan logika
    2. Tes sederhana
    3. Perpindahan data
- Program special yang mengeksekusi instruksi sesuai urutan hingga selesai
  - **Interpreter:** menerjemahkan dan mengeksekusi setiap perintah dalam suatu aksi urutan (perintah demi perintah)
  - **Compiler:** menghasilkan kode objek yang kemudian di-link oleh linker menjadi kode yang dapat dieksekusi



# Cara Kerja Komputer

- Manusia memberikan perintah pada computer
- Komputer beberapa kali menerjemahkan perintah
- Terdapat beberapa tingkat Bahasa Pemrograman:
  - **High level language:** Java, python, PHP
  - **Mid level language:** C
  - **Low level language:** Assembly
  - **Machine language:** 01010101

# Pemrograman

- Tidak sekedar membuat kode – kode perintah
- Belajar pemrograman BUKAN belajar Bahasa pemrograman
- Pemrograman = science atau art
- Kegiatan memprogram:
  - Utak atik, mencari solusi permasalahan, menganalisis, membuat spesifikasi, menulis kode, mengeksekusi kode, membaca program, menganalisis kompleksitas
- Kompleksitas = ke-kompleks-an dari suatu program berkaitan dengan pembuatan kode program yang efektif dan efisien

# Apa resep memprogram

- Urutan dari Langkah – Langkah sederhana
- Aliran proses control yang menentukan kapan masing - masing Langkah dieksekusi
- Tersedia sarana untuk menentukan kapan harus berhenti
- $1 + 2 + 3 = \text{algoritma!}$

**Secara formal, ALGORTIMA adalah daftar instruksi yang terbatas yang mendeskripsikan suatu KOMPUTASI yang ketika dijalankan pada set input akan di proses melalui set kejadian yang terdefinisi dengan baik dan akhirnya menghasilkan output**

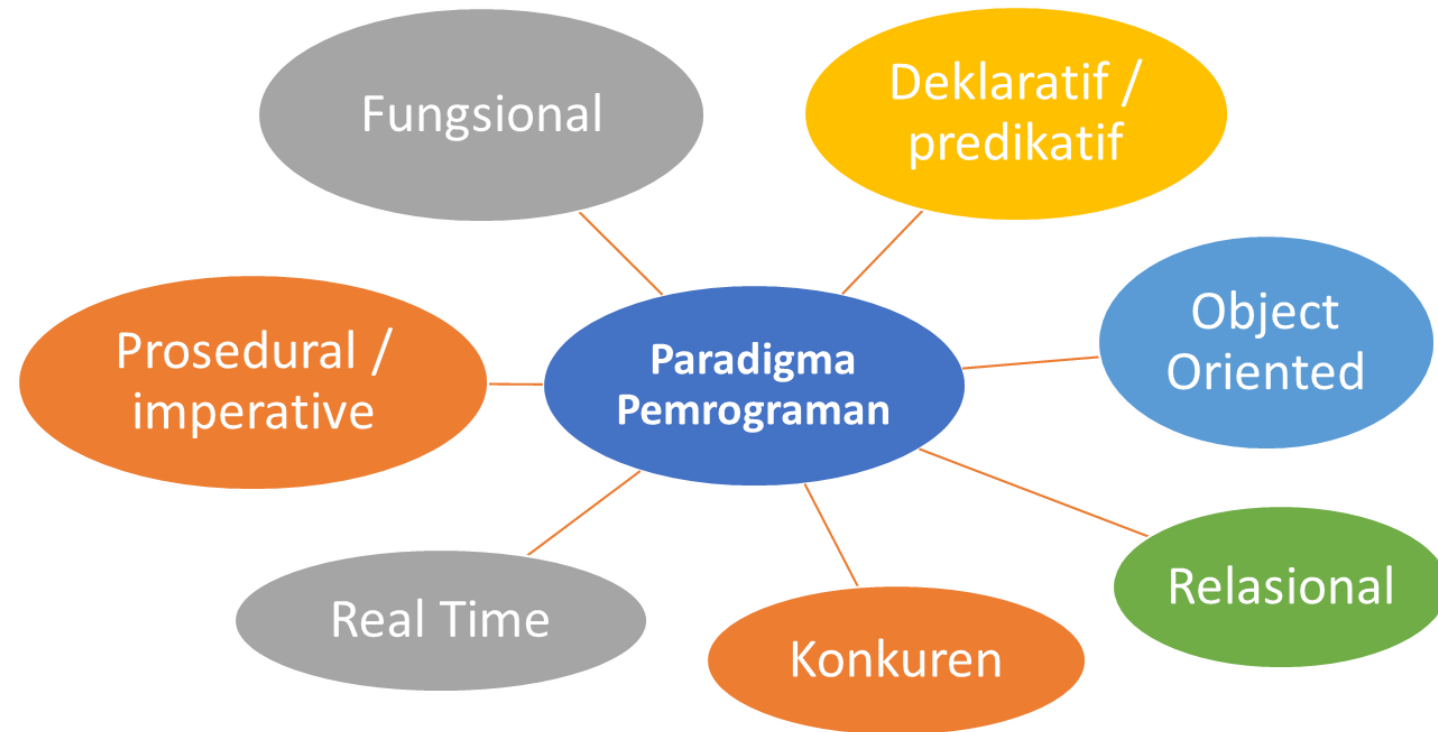


# Tipe Pengetahuan

- Pengetahuan **deklaratif** adalah suatu pernyataan fakta
  - Seseorang kenyang setelah makan mie goreng instan
- Pengetahuan **imperative/procedural** adalah resep atau bagaimana caranya
  1. Beli mi di warung
  2. Buka kemasan
  3. Siapkan air di panci sampai mendidih
  4. Masak mie
  5. Siapkan bumbu di piring
  6. Tiriskan mie
  7. Campur bumbu dengan mie
  8. Adi makan mie, sampai akhirnya kenyang

# Paradigma Pemrograman

- Paradigma pemrograman adalah sudut pandang penyelesaian persoalan dengan program



# Area Pemrograman

- Teksstual vs visual
- Dekstop vs internet vs mobile
- Client server vs N-tier
- Online vs batch vs mini-batch
- Lingkungan pemrograman:
  - Textual programming
  - Visual programming
- Realisasi program: source code



**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# Bahasa Pemrograman

# Bagaimana agar mesin paham dengan perintah kita?

- Menggunakan Bahasa mesin
  - Contoh: 10101011101010101110101
- Menggunakan Bahasa pemrograman
  - Contoh: `print("halo")`

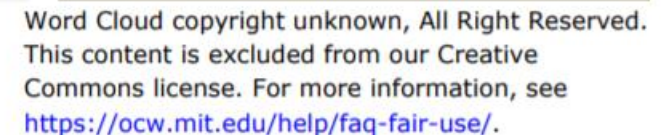


# Macam – macam Bahasa Pemrograman

- Ribuan Bahasa pemrograman di dunia
- Fortran, cobol, LISP, Prolog, Pascal, Basic, C, C++, C#, Java, Javascript, Perl, PHP, visual, c++, Python, R, B, dll



- Bahasa Indonesia: kata, angka
- Bahasa pemrograman: string, number, operator



## Aspek Bahasa (sintaks)

- Bahasa Indonesia:
  - “nasi ibu makan” → secara sintaksis tidak valid
  - “ibu makan nasi” → secara sintaksis valid
- Bahasa pemrograman:
  - “hi”5 → secara sintaksis tidak valid
  - 2+5\*100 → secara sintaksis valid

## Aspek Bahasa (semantic statis)

- Suatu kalimat atau string yang valid secara sintaks dan mempunyai arti
- Bahasa Indonesia:
  - Ibu dimakan nasi → secara sintaksis valid, secara semantic statis tidak valid
- Bahasa pemrograman:
  - $3+5$  → semantic statis valid
  - $3+"hi"$  → semantic statis tidak valid
- Semantik valid artinya makna yang terkait dengan rangkaian symbol/kata benar secara sintaksis tanpa kesalahan semantic statis

# Kesalahan Memprogram

- Error sintaksis
  - Hal biasa bagi pemula, mudah dicari dan diperbaiki
- Error semantic statis
  - Bahasa pemrograman secara otomatis akan memeriksa ini
  - Dapat menimbulkan perilaku tidak terduga (hasil yang sulit diartikan atau tidak pasti)
- Tidak ada kesalahan sintaksis maupun semantic statis tetapi **memiliki arti yang berbeda dengan yang diharapkan oleh programmer**
  - Crash/lag/infinite loop/memory leak
  - Output tidak seperti yang diharapkan
  - Istilah keren “bug” → solusi debugging/metani (proses untuk mencari bug/kutu)





**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS DIAN NUSWANTORO**

# Pemrograman Prosedural

# Pemrograman Prosedural

- Istilah lain adalah pemrograman imperative
- Dihasilkan berdasarkan suatu dekomposisi (pemecahan masalah) aksional, menjadi aksi – aksi yang dijalankan secara berurutan.
- Aksi-aksi:
  - Jelas ada suatu keadaan awal (*initial state*) dan keadaan akhir (*final state*) dengan waktu yang terbatas
  - dapat dipecah dalam sub aksi
- Aksi diterjemahkan sebagai instruksi – instruksi yang dapat dijalankan oleh mesin

## Aksi sekuensial

- Merupakan struktur control algoritma yang paling sederhana
- Terdiri dari deretan instruksi/aksi/perintah dari aksi ke 1,2,3,...,sampai n
- Aksi tersebut diawali dengan keadaan awal/initial state yang harus dipenuhi dan keadaan akhir/final state
- Initial state disingkat I.S
- Final state disingkat F.S



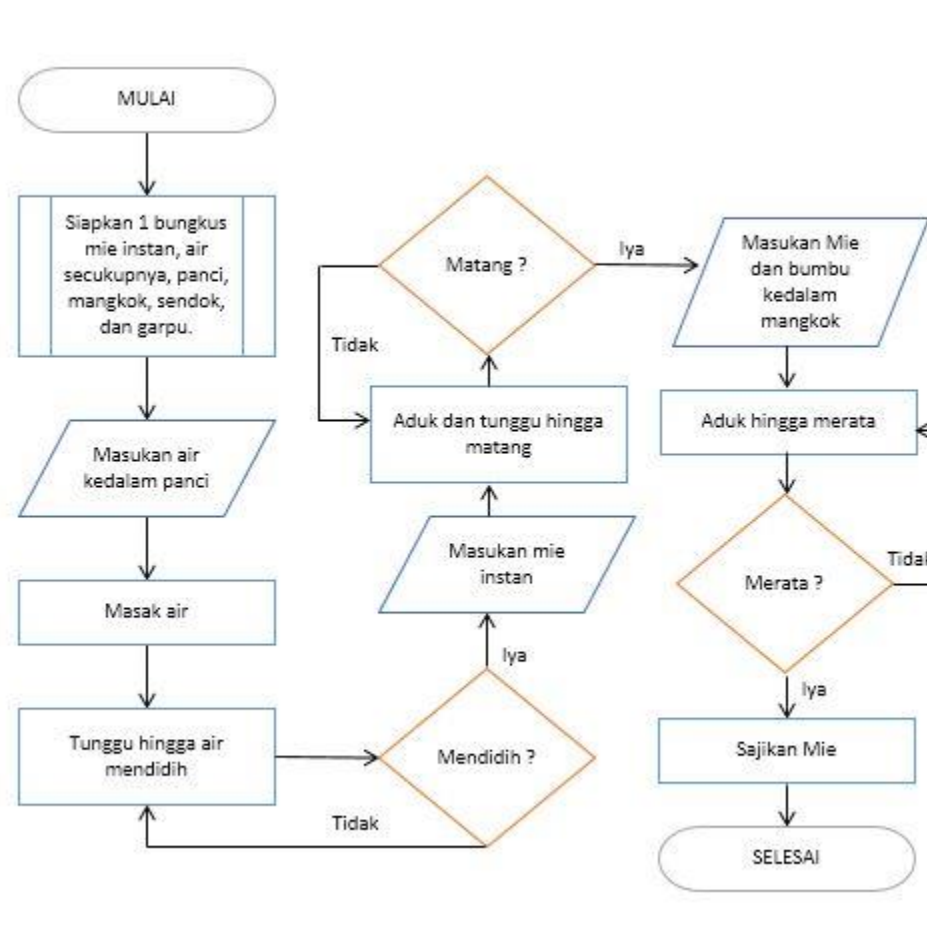
## Aksi masak mie instan(versi singkat)

- I.S: niat
- F.S: mie disajikan di piring
- Sub-aksi:
  1. Ambil uang untuk beli mie
  2. Beli mie instan di warung
  3. Buka bungkus mie instan
  4. Masak mie instan
  5. Siapkan bumbu
  6. Tiriskan mie

# Notasi Algoritmik

- Aksi sekuensial bisa dibuat dengan notasi algoritma
- Bentuk penulisan algoritma/teks algoritma
- Biasanya terdiri dari 3 bagian utama:
  1. **Judul:** spesifikasi deskripsi algoritma secara umum
  2. **Kamus:** definisi konstanta, tipe data, variable, spesifikasi prosedur / fungsi
  3. **Algoritma:** isi penyelesaian masalah dengan menggunakan elemen – elemen pada kamus, intinya program yang berisi instruksi – instruksi atau pemanggilan aksi

# Diagram Alir/Flow Chart



# Template Notasi Algoritmik

## **JUDUL**

*{ judul sebagai spesifikasi teks algoritmik secara umum }*

Algoritma memasak air

## **KAMUS**

*{pada bagian ini, dilakukan pendefinisian konstanta, nama variable, spesifikasi prosedur, spesifikasi fungsi yang akan digunakan dalam algoritma}*

*1 panci*

*1 liter air*

## **DESKRIPSI ALGORITMA**

*{spesifikasi teks algoritmik secara detail}*

Masukkan 1 liter air dalam sebuah panci

Panaskan air didalam panic hingga mendidih

## Keterangan

- Judul diisi dengan judul program untuk aksi sekuensial yang didefinisikan
- Kamus digunakan untuk mendeklarasikan suatu variable atau konstanta dengan suatu “tipe”
  - Contoh deklarasi a: integer
- Algoritma merupakan bagian utama program dalam bentuk aksi – aksi sekuensial
  - Bisa berisi bermacam – macam: instruksi perintah input/output, assignment, statemen sekuensial, analisis kasus, dan pengulangan

# komentar

- Komentar dapat memudahkan dalam membaca alur dalam sebuah program
- Fungsi komentar:
  - Menerangkan apa yang ada dalam sejumlah baris kode dibawahnya
  - Mendokumentasikan siapa yang menulis kode tersebut atau informasi tambahan lainnya
  - Menonaktifkan sejumlah baris kode – mungkin hanya temporer
- Contoh komentar

Notasi Algoritmik	C++
{Ini komentar}	//ini komentar /* ini komentar */

# Referensi

## Utama:

1. Bjarne Stroustrup, 2014, Programming: Principles and Practice Using C++ (Second Edition), Addison-Wesley Professional

## Pendukung:

1. Introduction to Computer Science and Programming in Python, MIT  
<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016>
2. Introduction to Computer Science and Programming, MIT  
<https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-00sc-introduction-to-computer-science-and-programming-spring-2011/index.htm>



# TERIMA KASIH

ANY QUESTIONS?