

EXPERT SYSTEM / ES (Sistem Pakar)

Definisi :

- Secara umum ES adalah system yang berusaha mengadopsi pengetahuan manusia ke computer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan para ahli.
- ES tidak untuk menggantikan kedudukan seorang pakar tetapi untuk memasyarakatkan pengetahuan dan pengalaman pakar tersebut.
- ES dikembangkan pertama kali oleh komunitas AI tahun 1960an. ES yang pertama adalah General Purpose Problem Solver (GPS) yang dikembangkan oleh Newel Simon.

Beberapa ES yang terkenal :

Sistem Pakar	Kegunaan
MYCIN Dirancang oleh Edward Feigenbaum (Universitas Stanford) th '70 an	Diagnosa Penyakit
DENDRAL	Mengidentifikasi struktur molecular campuran yang tidak dikenal
XCON & XSEL Dikembangkan oleh Digital Equipment Corporation (DEC) dan Carnegie Mellon Universitas (CMU), akhir '70 an	Membantu konfigurasi system computer besar
SOPHIE	Analisis sirkuit elektronik
PROSPECTOR Didesign oleh Sheffield Research Institute, akhir '70an	Digunakan di dalam geologi untuk membantu mencari dan menemukan deposit

FOLIO	Membantu memberikan keputusan bagi seorang manajer dalam hal stok broker dan investasi
DELTA	Pemeliharaan lokomotif listrik diesel

Keuntungan ES :

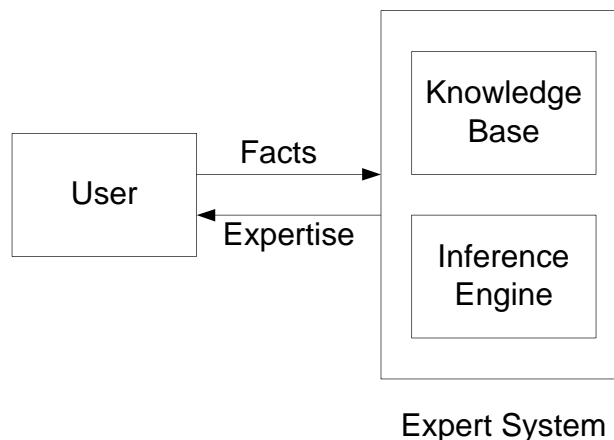
1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli
2. bisa melakukan proses secara berulang secara otomatis
3. menyimpan pengetahuan dan keahlian para pakar
4. meningkatkan output dan produktivitas
5. meningkatkan kualitas
6. mampu mengambil dan melestarikan keahlian para pakar
7. mampu beroperasi dalam lingkungan berbahaya
8. memiliki kemampuan untuk mengakses pengetahuan
9. memiliki realibilitas
10. meningkatkan kapabilitas system computer
11. memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian
12. sebagai media pelengkap dalam pelatihan
13. meningkatkan kapabilitas dalam penyelesaian masalah
14. menghemat waktu dalam pengambilan keputusan

Kelemahan :

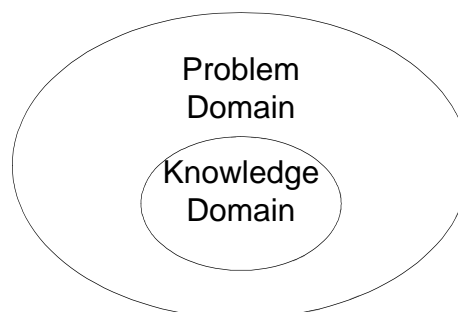
1. biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal
2. sulit dikembangkan. Hal ini erat kaitannya dengan ketersediaan pakar dalam bidangnya
3. system pakar tidak 100% bernilai benar

Konsep Dasar ES

- Menurut Efraim Turban, system pakar harus mengandung : keahlian, ahli, pengalihan keahlian, inferensi, aturan dan kemampuan menjelaskan.
- Keahlian adalah suatu kelebihan penguasaan pengetahuan di bidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman. Bentuk pengetahuan :
 - fakta-fakta pada lingkup permasalahan tertentu
 - teori-teori pada lingkup masalah tertentu
 - prosedur-prosedur berkenaan dengan lingkup masalah tertentu
 - strategi-strategi global untuk menyelesaikan masalah
 - *meta-knowledge* (pengetahuan tentang pengetahuan)
- Blok diagram ES



- Knowledge base (basis pengetahuan) berisi pengetahuan-pengetahuan dalam penyelesaian masalah.
 - Domain pengetahuan seorang pakar pada dasarnya adalah spesifik terhadap domain masalah.



- Inference engine (motor inferensi) bertugas untuk menganalisis pengetahuan dan menarik kesimpulan berdasarkan knowledge base.
- Sistem Konvensional vs. Sistem Pakar (ES)

Sistem Konvensional	Sistem Pakar (ES)
Informasi dan pemrosesan biasanya jadi satu dengan program	Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi
Biasanya tidak bisa menjelaskan mengapa suatu input data itu dibutuhkan atau bagaimana output itu diperoleh	Penjelasan adalah bagian terpenting dari system pakar
Pengubahan program cukup sulit	Pengubahan aturan dapat dilakukan dengan mudah
Sistem hanya akan beroperasi jika system tersebut sudah lengkap	Sistem dapat beroperasi hanya dengan beberapa aturan
Eksekusi dilakukan langkah demi langkah	Eksekusi dilakukan pada keseluruhan basis pengetahuan
Menggunakan data	Menggunakan pengetahuan
Tujuan utamanya adalah efisiensi	Tujuan utamanya adalah efektivitas

- Ciri-ciri ES :
 - Memiliki fasilitas informasi yang handal
 - Mudah dimodifikasi
 - Dapat digunakan dalam berbagai jenis komputer
 - Memiliki kemampuan untuk belajar beradaptasi.

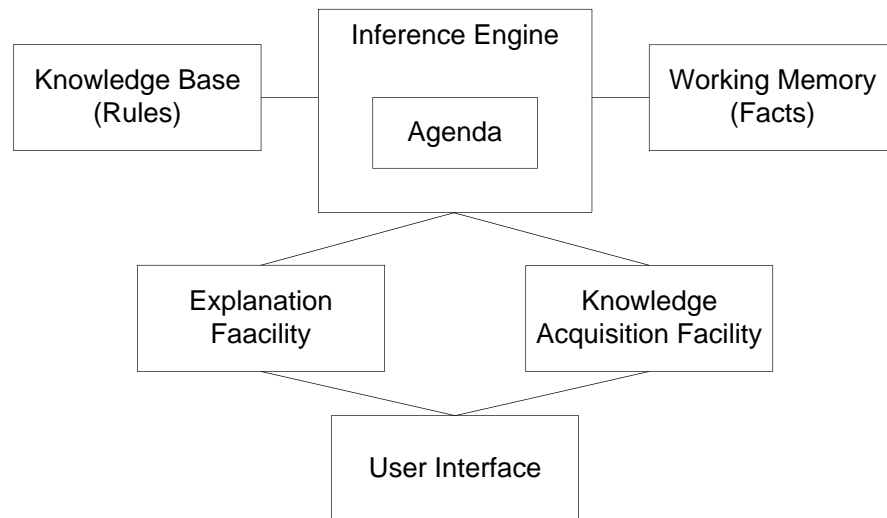
- Permasalahan yang Disentuh oleh ES (Domain ES) :
 - Interpretasi. Pengambilan keputusan dari hasil observasi, termasuk pengenalan ucapan, analisis citra, interpretasi sinyal, dll
 - Prediksi : prediksi demografi, prediksi ekonomi, dll.
 - Diagnosis : diagnosis medis, elektronis, mekanis, dll.
 - Perancangan : perancangan layout sirkuit , bangunan.
 - Perencanaan : perencanaan keuangan, militer, dll
 - Monitoring : *computer aided monitoring system*
 - Debugging : memberikan resep obat terhadap kegagalan
 - Instruksi : melakukan instruksi untuk diagnosis, debugging dan perbaikan kinerja
 - Kontrol : melakukan kontrol terhadap interpretasi, prediksi, perbaikan dan monitoring kelakukan sistem.

- Bentuk ES :
 1. Berdiri sendiri. Sistem jenis ini merupakan s/w yang berdiri sendiri tidak tergabung dengan s/w lain.
 2. Tergabung. Sistem ini merupakan bagian program yang terkandung di dalam suatu algoritma (konvensional) .
 3. Menghubungkan ke s/w lain. Bentuk ini biasanya merupakan ES yang menghubungkan ke suatu paket program tertentu, misalnya DBMS.
 4. Sistem mengabdikan. Sistem ini merupakan bagian dari computer khusus yang dihubungkan dengan suatu fungsi tertentu.

- Elemen ES
 - User interface (antarmuka) : mekanisme komunikasi antara user dan ES
 - Explanation facility (subsistem Penjelasan) : digunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif

- Working memory : database global dari fakta yang digunakan dalam prosedur
- Agenda : daftar prioritas prosedur yang dibuat oleh motor inferensi dan direkam dalam working memory
- Inference engine (motor inferensi) : program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan untuk memformulasikan konklusi.
- Knowledge acquisition facility : berisi pengetahuan-pengetahuan yang dibutuhkan untuk memahami, memformulasikan dan menyelesaikan masalah.

- Struktur ES



- Bahasa, Shell dan Tools ES
 - Bahasa ES difokuskan pada fleksibilitas dan robust dalam merepresentasikan pengetahuan
 - Bahasa ES merupakan bahasa tingkat tinggi yang dirancang secara khusus untuk representasi pengetahuan dan alasan (reasoning).
 - Contoh Bahasa ES : SAIL, KRL, KQML, DAML
 - ES Shell : tools khusus yang dirancang untuk mendukung aplikasi ES, pada saat user memasukkan basis pengetahuan.
 - Contoh ES Shell : EMYCIN (untuk MYCIN), CLIPS

Basis Aturan (*Rule Based*) ES

- Pengetahuan dalam ES direpresentasikan dalam bentuk **IF-THEN** atau dalam bentuk **Production Rules**.
- Motor inferensi menentukan aturan awal (rule antecedents) yang sesuai.
 - o Sisi kiri harus cocok dengan fakta yang ada di memori kerja
- Aturan yang sesuai ditempatkan di agenda dan dapat diaktivasi
- Aturan yang terdapat di agenda dapat diaktivasi
 - o Aktivasi aturan akan membangkitkan fakta baru di sisi kanan
 - o Aktivasi dari satu aturan adalah bagian dari aktivasi aturan yang lain.
- Contoh Basis Aturan :

IF THEN Rules

Rule : Red_Light

IF the light is red
THEN stop

Antecedent
(left hand side)

Rule : Green_Light

IF the light is green
THEN go

Consequent
(right hand side)

Production Rules

The light is red ⇒ stop

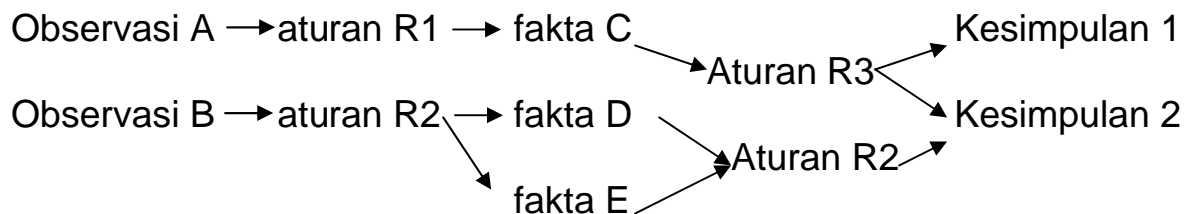
The light is green ⇒ go

Antecedent
(left hand side)

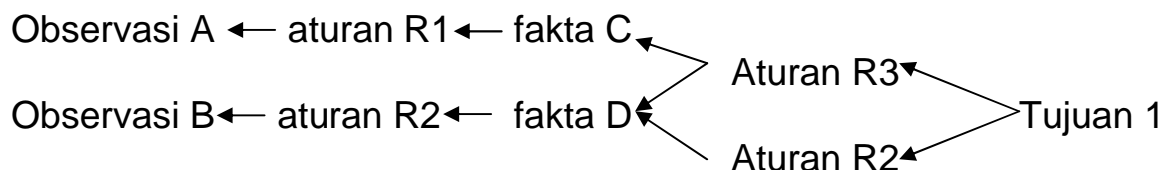
Consequent
(right hand side)

Siklus Motor Inferensi (*Inference Engine Cycle*)

- Menggambarkan eksekusi dari aturan pada motor inferensi :
 - Conflict resolution : pemilihan aturan dengan prioritas tertinggi dari agenda
 - Execution : aksi *consequent* dari aturan yang terpilih
 - Match : pengkinian (update) agenda
- Siklus berakhir ketika tidak ada lagi aturan di agenda atau ketika ditemui perintah stop.
- Ada 2 cara yang dapat dilakukan dalam melakukan inferensi :
 1. Forward Chaining. Pencocokkan fakta atau pernyataan dimulai dari bagian sebelah kiri (IF dulu). Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis.



2. Backward Chaining. Pencocokkan fakta atau pernyataan dimulai dari bagian sebelah kanan (THEN dulu). Dengan kata lain, penalarana dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut dicari fakta-fakta yang ada dalam basis pengetahuan.

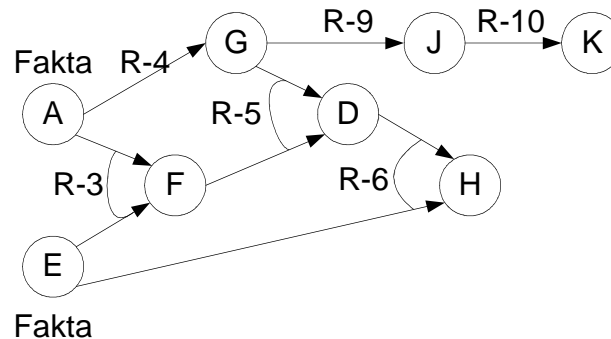


Contoh :

1. Pada tabel di bawah ini terlihat 10 aturan yang tersimpan dalam basis pengetahuan. Fakta awal yang diberikan hanya A & F (artinya A dan F bernilai benar). Ingin dibuktikan apakah K bernilai benar (hipotesis : K) ?

No	Aturan
R-1	IF A & B THEN C
R-2	IF C THEN D
R-3	IF A & E THEN F
R-4	IF A THEN G
R-5	IF F & G THEN D
R-6	IF G & E THEN H
R-7	IF C & H THEN I
R-8	IF I & A THEN J
R-9	IF G THEN J
R-10	IF J THEN K

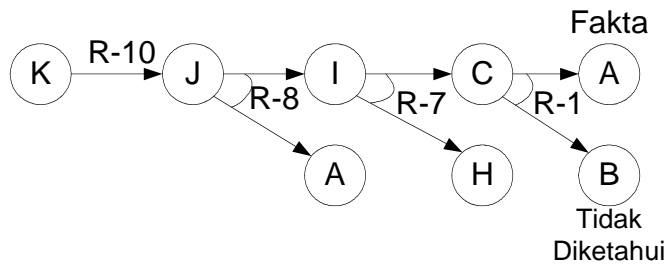
Penyelesaian dengan Forward Chaining :



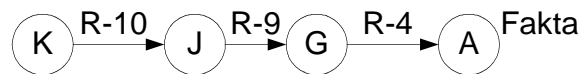
Fakta Baru

Aturan	Fakta Baru
R-3	F
R-4	G
R-5	D
R-6	H
R-9	J
R-10	K (terbukti)

Penyelesaian dengan Backward Chaining :



Pertama : Gagal



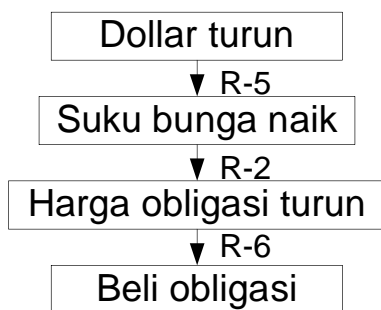
Kedua : Sukses

2. Diketahui sistem pakar dengan aturan-aturan sebagai berikut:

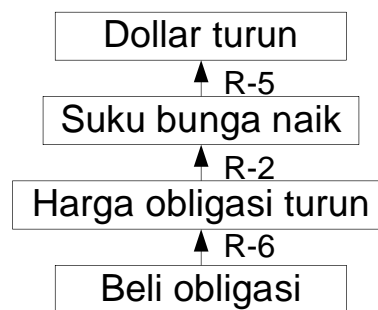
- R1 : IF suku bunga turun THEN harga obligasi naik
- R2 : IF suku bunga naik THEN harga obligasi turun
- R3 : IF suku bunga tidak berubah THEN harga obligasi tidak berubah
- R4 : IF dollar naik THEN suku bunga turun
- R5 : IF dollar turun THEN suku bunga naik
- R6 : IF harga obligasi turun THEN beli obligasi

Apabila diketahui bahwa dollar turun, maka buatlah keputusan apakah akan membeli obligasi atau tidak ?

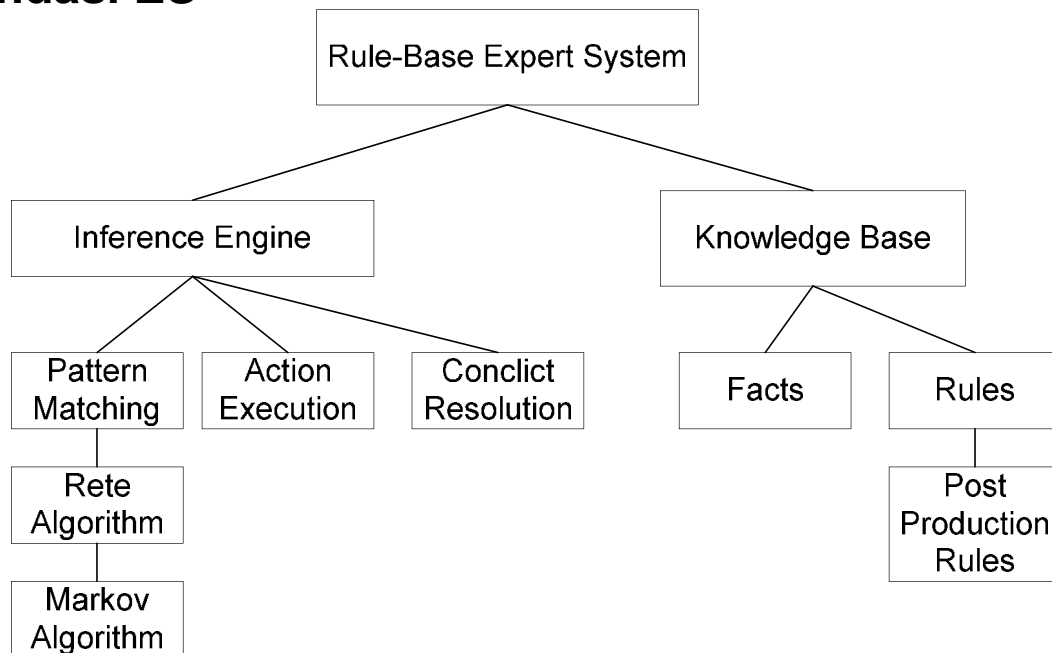
Solusi Forward Chaining



Solusi Backward Chaining



Fondasi ES



Sistem Produksi

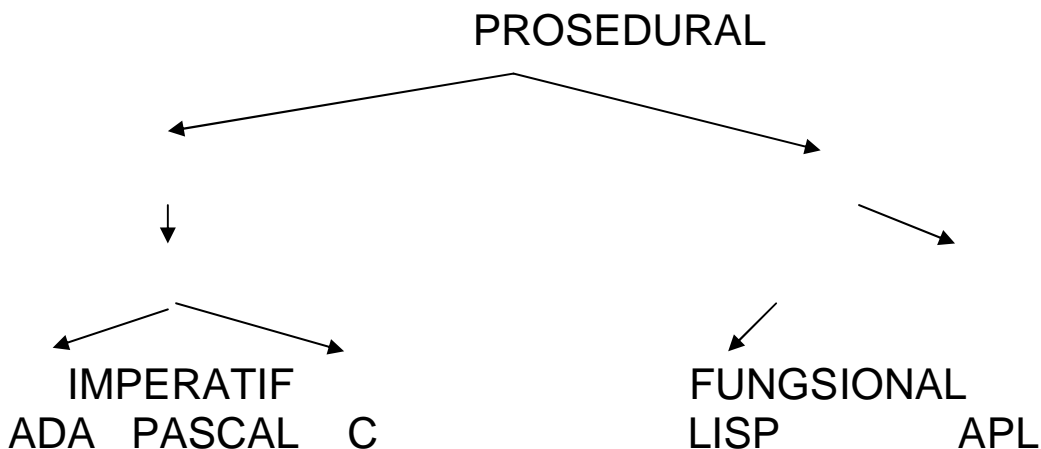
- Salah satu tipe ES yang terkenal adalah sistem yang berdasarkan pada aturan
- Alasannya :
 - modular nature
 - memiliki fasilitas penjelasan
 - kesamaan dalam pemrosesan cognitive
- Sistem Produksi Post
 - aturan produksi yang digunakan oleh Emil L.Post (sekitar awal 1940an) dalam simbol logika
 - Teori Post :
 - Setiap sistem matematik atau logika dapat ditulis dalam bentuk sistem produksi
 - Prinsip aturan produksi
 - Suatu aturan yang mengatur konversi sebuah set string ke dalam set string yang lain.
 - Manipulasi string dalam sintaks sederhana
 - Contoh : BNF grammar pada bahasa pemrograman

- Algoritma Markov
 - Sekitar 1950an A.A.Markov memperkenalkan prioritas sebagaii strktur kendali pada sistem produksi.
 - aturan dengan prioritas tertinggi dijalankan terlebih dahulu
 - efisiensi eksekusi dalam sistem produksi
 - namun... untuk set aturan yang besar, tidak efisien

- Algoritma Rete
 - Dibangun oleh Charles L.Forgy (1970an) untuk Official Production System CMU sebagai perbaikan Algoritma Markov
 - Algoritma yg mengetahui tentang seluruh aturan/baris sistem dan dapat menerapkan suatu baris tanpa harus mencoba setiap baris tanpa berangkai (mencari perubahan dalam gabungan setiap cycle)
 - Merupakan gabungan pola yang sangat cepat, yang mendapatkan kecepatannya dengan menyimpan informasi tentang baris dalam jaringan

Klasifikasi Paradigma Pemrograman

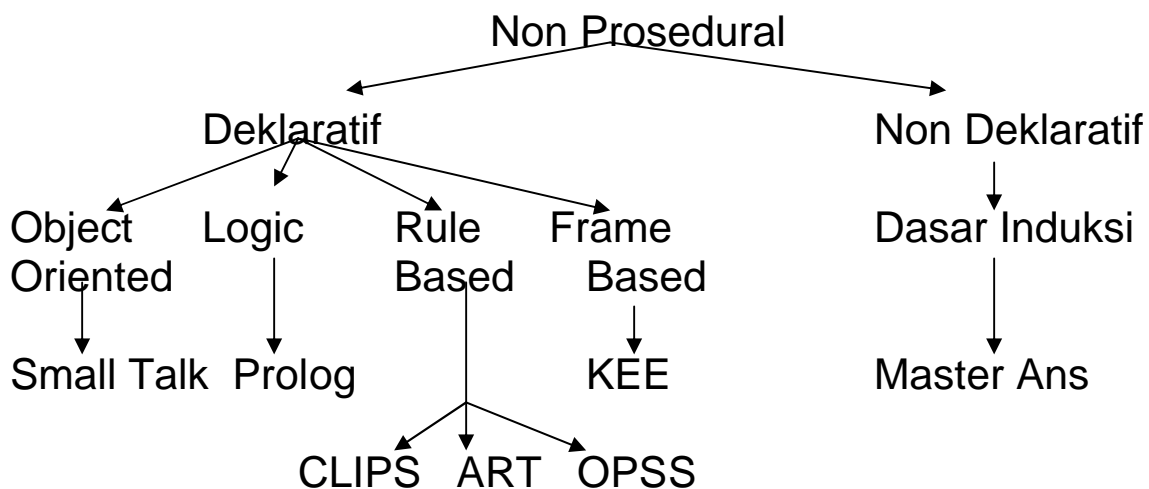
1. Paradigma PROSEDURAL



ALGORITMA adalah metode untuk pemecahan masalah dalam sejumlah tahap/langkah tertentu.

- ✓ Implementasi algoritma dalam suatu program disebut *program prosedural*.
- ✓ Pemrograman algoritma (prosedural) dan konvensional untuk program type non-AI.
- ✓ Sinonim untuk pemrograman prosedural adalah prog. Sequential.
- ✓ Pada pemrograman prosedural programmer harus menentukan sesungguhnya bagaimana pemecahan masalah harus di-code-kan.
- ✓ Pembuat code adalah pemrograman non prosedural.

2. Paradigma NON PROSEDURAL



- ✓ Penekanan pemrograman Non prosedural adalah penentuan apa yg akan diselesaikan dan membiarkan system menentukan bagaimana menyusunnya.

➤ PEMROGRAMAN DEKLARATIF

Memisahkan tujuan dari metode yg digunakan untuk mencapai tujuan.

➤ PEMROGRAMAN OBJECT ORIENTED

Ide : membuat design program dg mempertimbangkan data yg digunakan dalam program sebagai objek dan mengimplemnetasikan operasi pada objek tersebut.

➤ PEMROGRAMAN LOGIKA

Pembuktian teori logika dg Logic Theorirt Program (Newell & Simon) pada Darmouth Conference A.I (1956)

Rangkaian backward dapat digunakan untuk mengekspresikan pengetahuan dalam representasi deklaratif maupun kontrol proses pemberian alasan.

Keuntungannya : pembuatannya dapat diproses secara paralel yaitu jika ada beberapa processor dapat bekerja secara simultan

EXPERT SYSTEM

- ✓ Disebut *pemrograman deklaratif* krn programmer tdk menentukan bagaimana prog. hrs mendapatkan tujuannya pada level algoritma