

## BAB I

### PENDAHULUAN

#### 1.1. Pengertian dan Sejarah Dari Artificial Intelligence(AI)

AI mempelajari bagaimana membuat komputer melakukan sesuatu pada suatu kejadian/peristiwa yang mana orang melakukannya dengan baik. <sup>1)</sup>.

##### a. Pengertian AI

Definisi AI : *merupakan proses di mana peralatan mekanik dapat melaksanakan kejadian-kejadian dengan menggunakan pemikiran atau kecerdasan seperti manusia.*

Pengertian AI dapat ditinjau dari dua pendekatan : <sup>2)</sup>

##### 1). Pendekatan Ilmiah( *A Scientific Approach*)

Pendekatan dasar ilmiah timbul sebelum invansi ke komputer, ini tidak sama dengan kasus mesin uap. Pendekatan ilmiah melihat batas sementara dari komputer, dan dapat diatasi dengan perkembangan teknologi lanjutan. Mereka tidak mengakibatkan tingkatan pada konsep.

##### 2). Pendekatan Teknik( *An Engineering Approach*)

Usaha untuk menghindari definisi AI, tetapi ingin mengatasi atau memecahkan persoalan-persoalan dunia nyata(*real world problem*).

Dalam kuliah ini, kita menggunakan kedua pendekatan ini.

*Mengapa kita mempelajari AI ? karena*

- *AI merepresentasikan bagian tengah atau inti dari ilmu komputer(Computer Science).*
- *AI mewujudkan suatu bentuk ketidak tepatan dari komputasi (karakteristik dalam matematika).*
- *AI mempunyai suatu kekuatan alami antar cabang ilmu, AI adalah bagian ilmu teknik dari Cognitive Science, Cognitive Science adalah suatu perpaduan ilmu filsafat, ilmu linguistik dan ilmu psikologi.*
- *AI memperlakukan representasi pengetahuan dan manipulasinya.*

---

1). Rich, Elaine, and Knight, Kevin, "Artificial Intelligence", Second Edition, page 3, McGraw-Hill Inc., 1991

2). Charniack, Eugene and McDermott, Drew, "Introduction To Artificial Intelligence", page 1, McGraw-Hill Inc., 1985.

- *Pengetahuan (knowledge) adalah pusat dari semua ilmu teknik dan AI adalah pusat dari semua ilmu teknik.*
- *Alasan penting lainnya adalah penelitian AI diharapkan me-nemukan atau membongkar bentuk krisis besar dalam waktunya. Krisis dibuat oleh interaksi dari teknologi, ilmiah(science) dan filsafat.*

Program *Intelligent*: program yang mampu menyimpan kenyataan (*facts*) dan proposisi dan hubungannya yang beralasan.

#### **b. Sejarah dari AI**

Awal pekerjaan dipusatkan pada seperti *game playing* (misalnya: audio dengan kecerdasan dan permainan catur(*chess player*), pembuktian teorema (*theorem proving*) pada Tugas-tugas formal (*Formal Tasks*).

Samual(1963) menulis sebuah program yang diberi nama *check-er-playing* program, yang tidak hanya untuk bermain *game*, tetapi digunakan juga pengalamannya pada permainan untuk mendukung kemampuan sebelumnya.

Catur juga diterima, karena banyak sekali perhatian terhadap permainan catur yang merupakan permainan yang lengkap atau kompleks, program catur di sini situasinya harus jelas dan rule atau ketentuannya harus seperti dunia nyata. Kandidat AI harus mampu menangani masalah-masalah yang sulit.

*Logic theorist* diawal percobaan untuk membuktikan teorema matematika. Ia mampu membuktikan beberapa teorema dari bab 1 Prinsip Matematika Whitehead dan Russell.

*Theorema Gelernter* (1963) membuktikan pencarian area yang lain dari matematika yaitu geometri.

Pada tahun 1963, pemecahan masalah umum menggunakan *object*, pembuktian dengan atraksi(eksternal).

Dari awal pekerjaan AI ini memindahkan lebih khusus tugas yang sering berguna antara lain: <sup>3)</sup>

#### **a. Tugas biasa/keduniaan(*Mundane Tasks*)**

- Persepsi : - *vision*  
- *speech*

---

3) opcit <sup>1)</sup>, page 5.

- *Natural Language* :
  - *understanding*
  - *generation*
  - *translation*
- *Commonsense Reasoning* ( pertimbangan berdasarkan pikiran sehat):
  - *robot control*

b. Tugas Formil(*Formal Tasks*)

- *Games* :
  - *chess*
  - *checkers*
- *Matematics:*
  - *geometri*
  - *logic*
  - *proving properties of programs*

c. Tugas Ahli(*Expert Tasks*)

- *Teknik* :
  - *Design*
  - *Fault Diagnosis*
  - *Planning*
- *Scientific Analysis*
- *Medical* : *Diagnosis & Theraphy.*

## 1.2. Aplikasi-aplikasi Dari AI

Kecerdasan tiruan (AI) telah dipelajari selama kira-kira 40 tahun. Hingga saat ini telah dihasilkan beberapa produk aplikasi AI secara komersial. Produk-produk tersebut umumnya dapat dijalankan pada perangkat keras komputer mulai dari komputer pribadi(PC) se-harga USA\$5000 sampai dengan komputer besar(*mainframe*) seharga USA\$50,000. Secara khas masukkan untuk produk-produk tersebut berbentuk data simbolis. Aplikasi-aplikasi AI antara lain:

- ☐ *Game Playing*
- ☐ *Sistem Bahasa Alami*
- ☐ *Sistem Perancangan dan Pembuatan CAD/CAM*
- ☐ *Sistem Pakar VLSI*
- ☐ *Sistem Pakar Reparasi Perangkat Keras*

- ❑ *Manajemen Data Cerdas*
- ❑ *Sistem Otomatisasi Kantor*
- ❑ *Analisa Kecerdasan Militer*
- ❑ *Kendali dan Pemanggilan informasi disk video*
- ❑ *Kendali Robot*
- ❑ *Analisis Program Komputer*
- ❑ *Diagnosis Penyakit*
- ❑ *Konfigurasi komputer*
- ❑ *Ramalan senyawa kimia*
- ❑ *Sintesis ucapan*
- ❑ *Sistem Pakar Operator Komputer*
- ❑ *Manajemen Kendali Senjata*

Apa AI hari ini adalah sesuatu yang baru sekarang. Bagian-bagian dari AI antara lain :

- ❑ *Parallel Distributed Processing (Neural Network)*
- ❑ *Machine Vision*
- ❑ *Automatic Programming Tools*
- ❑ *Memory Management*
- ❑ *Pattern Recognition*
- ❑ *Natural Language Processing*
- ❑ *Development Of Knowledge Base*

Kecerdasan tiruan(*Artificial Intelligence*) adalah sub bagian dari ilmu komputer yang merupakan suatu teknik perangkat lunak yang pemrogramannya dengan cara menyatakan data, pemrosesan data dan penyelesaian masalah secara simbolik, dari pada secara numerik.

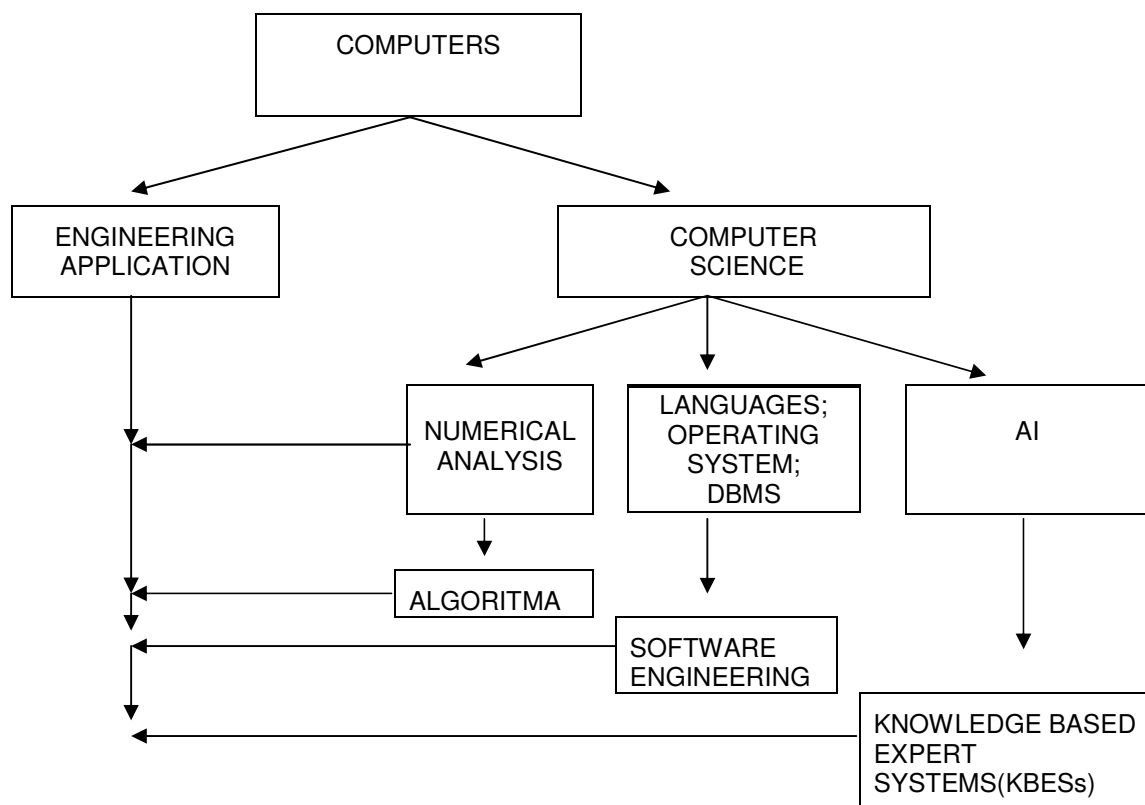
Masalah-masalah dalam bentuk simbolik ini adalah masalah-masalah yang sering kita jumpai dalam kehidupan sehari-hari. Masalah-masalah ini lebih berhubungan dengan simbol dan konsep simbol dari pada dengan angka-angka.

Di sini dengan kecerdasan tiruan diusahakan untuk membuat komputer seakan dapat berpikir secara cerdas.

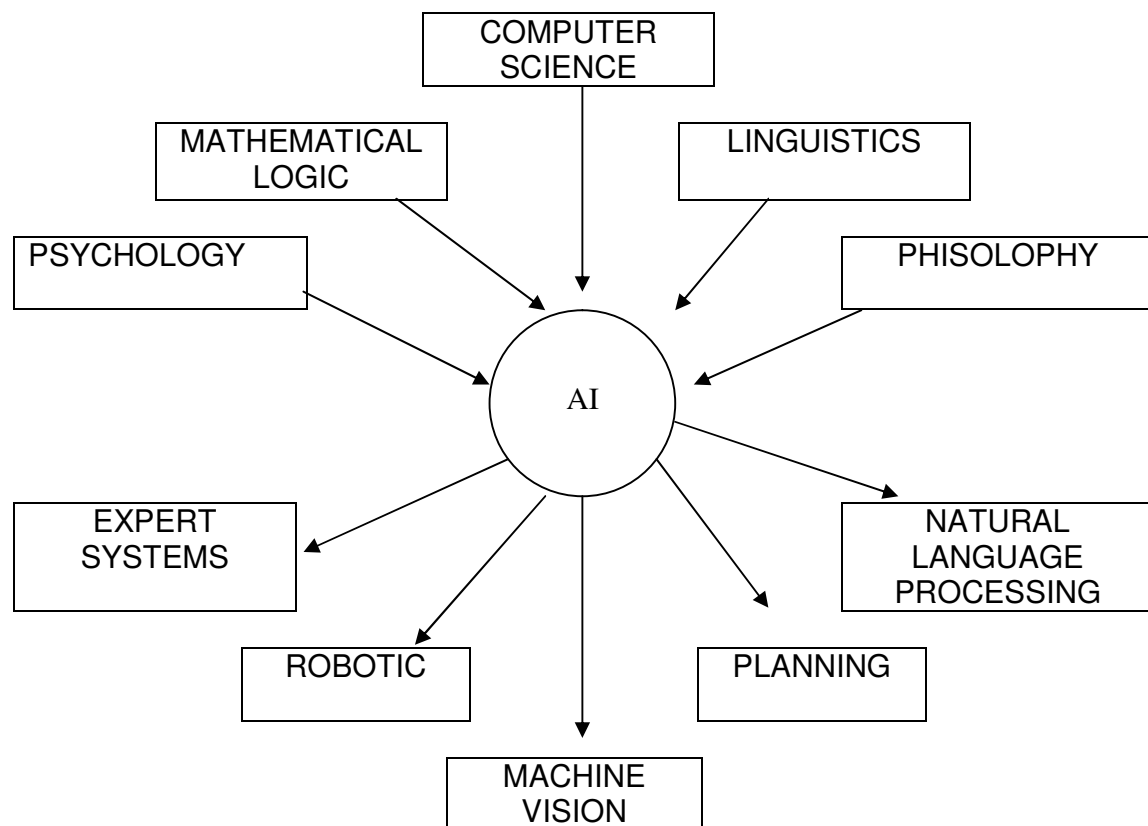
Untuk mudah dapat dimengerti dengan apa yang dimaksud dengan kecerdasan tiruan atau proses simbolik adalah dengan cara membandingkannya dengan program konvensional yang melakukan operasi numerik.

Program konvensional dapat menjawab “ $X + Y$ ” jika harga  $X$  dan  $Y$  diberikan, tetapi program ini tidak dapat menjawab bahwa “ $X + X = 2X$ ”, atau tidak dapat menjawab mengapa mobil tidak dapat distart.

Program kecerdasan tiruan berbeda dari program konvensional tidak saja dalam hal mengerti akan simbol atau informasi, tetapi juga program simbolik, karakter digit, kata, data dan lainnya saling berhubungan. Informasi dan bentuk hubungannya yang bervariasi digunakan untuk merepresentasikan hubungan antar-informasi. Hubungan antara simbol dan informasi secara tidak langsung menyatakan apa yang diinterpretasikan oleh manusia sebagai arti atau pengetahuan.



Gambar 1.1. Hubungan Antara Pengembangan di dalam Computer Science dan Engineering Applications.  
 Sumber: Clive L. Dym & Raymond E. Levitt, "Knowledge Based Systems in Engineering", page 9, Mc Graw-Hill, 1991



Gambar 1.2. Suatu Input-Output Model Untuk Artificial Intelligence  
 . (Source : Kamran Parsaye & Mark Chignell, "Expert Systems For Experts", page 11, John Wiley & Sons Inc, 1988)

Apa saja yang dikerjakan dalam bagian-bagian AI berikut ini:

- ❑ *Machine Vision* : Bertujuan pada pengenalan pola dalam beberapa jalan yang sama sebagai kegiatan sistem visual/indera manusia.
- ❑ *Robotics* : Difokuskan pada produksi alat-alat mekanik yang dapat mengendalikan gerak. Sebagai contoh: sebuah robot sederhana mampu atau dapat bergerak/pindah ke depan, belakang, kanan atau kiri atau pindah tempat ke ruangan berbeda. Sebuah robot sebenarnya buta akan bentuk urutan dari aksi bila tanpa usaha untuk mengganti komponennya atau bisa mendeteksi dan memperbaiki kesalahan dalam rencananya akan menjadi sulit bila tanpa kecerdasan. Sering sebuah robot akan diformulasikan pada sebuah rencana dasar pada informasi yang tidak lengkap dan benar dalam menjalankan sebuah rencana

- ❑ *Speech Processing* : Bertujuan pada pengenalan dan sintesa pembicaraan manusia.
- ❑ *Theorem Proving* : Usaha untuk membuktikan secara otomatis masalah-masalah dalam matematika dan logika.
- ❑ *General Problem Solving* : Bertujuan pada pemecahan kelas-kelas dari masalah-masalah yang ditekankan dalam sebuah bahasa formal.
- ❑ *Pattern Recognition*: Difokuskan pada pengenalan dan klasifikasi dari pola-pola.
- ❑ *Game Playing*: Pembuatan program-program bermain permainan.
- ❑ *Machine Learning* : Bertujuan pada produksi mesin-mesin yang mengakumulasi pengetahuan dengan contoh-contoh observasi.
- ❑ *Learning* merupakan sebuah persoalan sulit untuk program AI, dalam mencapai kesuksesan diperlukan dalam pemecahan persoalan. Bagian kemampuan untuk mempelajari komponen terpenting dari tindak tanduk/jalan kecerdasan. Sebuah sistem pakar harus berkemampuan ekstensif dan dapat menghitung kerugian dalam memecahkan sebuah persoalan. Tidak seperti manusia, bilamana jika ia diberikan persoalan yang serupa pada waktu berikutnya, dia tidak akan ingat solusinya. Dia membentuk urutan yang sama untuk menghitung lagi. Learning merupakan sebuah area yang sulit diteliti, beberapa program telah ditulis dengan tujuan bahwa ini bukan merupakan hasil(goal) yang diinginkan.
- ❑ *Planning* adalah aspek terpenting pendukung untuk mendesain atau merancang robot-robot dengan kemampuan menyelesaikan tugas mereka dengan tingkat fleksibilitas dan tanggap terhadap dunia luar. Planning merupakan masalah sulit dari sejumlah alasan yang tidak lebih dari ukuran tempat kosong(space) yang mungkin diurutkan dan dipindahkan.
- ❑ *Neural Network atau Parallel Distributed* : teknik-teknik terbaik untuk merepresentasikan pengetahuan dan merancang algoritma pencarian yang hati-hati untuk implementasi kecerdasan.

### 1.3. Kerja Kecerdasan Manusia

Karena kecerdasan tiruan adalah ilmu yang berdasarkan proses manusia berpikir, maka penelitian bagaimana proses manusia berpikir adalah hal yang pokok.

Pada saat ini para peneliti hanya mulai mengerti sedikit dari proses berpikir tersebut, tetapi sudah cukup diketahui untuk membuat asumsi-asumsi yang pasti tentang bagaimana cara berpikir dan menggunakan asumsi-asumsi tersebut untuk mendesain suatu pro-gram komputer yang mempunyai kecerdasan secara tiruan.

Semua proses berpikir menolong manusia untuk menyelesaikan sesuatu masalah. Pada saat otak manusia mendapat informasi dari luar, maka suatu proses berpikir memberikan petunjuk tindakan atau respon apa yang dilakukan. Hal ini merupakan suatu reaksi otomatis dan respon yang spesifik dicari untuk menyelesaikan masalah tertentu. Hasil akhir dari semua proses berpikir tersebut disebut tujuan (goal).

Pada saat tujuan telah dicapai, pikiran akan segera berha-dapan dengan tujuan-tujuan lainnya yang akan dicapai. Di mana se-mua tujuan-tujuan ini bila terselesaikan akan mengantar ke suatu tujuan utama. Dalam proses ini tidak ada satupun cara berpikir yang mengarah ke tujuan akhir dilakukan secara acak dan sembarangan.

Kecerdasan manusia dapat dipecah-pecah menjadi kumpulan fakta-fakta (*facts*) dan fakta-fakta ini yang digunakan untuk mencapai tujuan. Hal ini dilakukan dengan memformulasikan sekelompok aturan-aturan(*rules*) yang berhubungan dengan fakta-fakta yang disimpan dalam otak.

Contoh jenis fakta dan aturan yang berhubungan, yang digunakan sehari-hari, adalah:

Fakta 1 : *Air sangat mendidih*

Aturan 1 : **IF** *saya menaruh tangan ke air panas* **THEN** *sakit*

Di sini aturan ditulis dalam bentuk IF-THEN yang berdasarkan fakta, dimana IF adalah kondisi tertentu yang ada, dan THEN adalah respon atau aksi yang akan dihasilkan.



Dalam proses berpikir, proses ini berhubungan dengan fakta-fakta yang sangat banyak sebelum memberikan suatu tindakan atau respon. Selama proses ada suatu sistem yang mengarahkan pemilihan respon yang tepat. Proses ini disebut dengan pemotongan(*prunning*). Proses ini mengeliminasi lintasan dari berpikir yang tidak relevan dalam usaha mencapai tujuan. Jadi proses ini akan memotong setiap fakta-fakta atau aturan-aturan yang tidak akan mengarah ke tujuan.

Teknik pemrograman dengan kecerdasan tiruan melakukan prosesnya sama dengan apa yang dilakukan oleh otak manusia. Kecerdasan tiruan juga meniru proses belajar manusia di mana informasi yang baru diserap dan dimungkinkan untuk digunakan sebagai referensi pada waktu yang akan datang. Di sini informasi yang baru dapat disimpan tanpa harus mengubah cara kerja pikiran atau mengganggu seluruh fakta-fakta yang sudah ada. Sehingga dengan kecerdasan tiruan dimungkinkan untuk membuat program di mana se-tiap bagian dari program benar-benar independen. Di sini setiap bagian dari program seperti potongan-potongan informasi dalam otak manusia.

Secara umum kecerdasan tiruan dibagi menjadi tiga kategori dasar, yaitu:

1. Sistem Berbasis Pengetahuan atau sistem pakar(*Expert System/Knowledge Based System*), yaitu program komputer yang berisi pengetahuan manusia yang digunakan untuk menyelesaikan masalah dalam domain tertentu.
2. Sistem bahasa alami(*Natural Language System*), yaitu pemrograman yang mengerti bahasa manusia.
3. Sistem dengan kemampuan memahami(*Perception System*), yaitu sistem untuk penglihatan, pembicaraan atau sentuhan.

Dari ketiga jenis kecerdasan tiruan itu, sistem pakar adalah yang paling banyak aplikasinya dalam membantu menyelesaikan masalah-masalah dalam dunia nyata.

*Contoh aplikasi dari program ini antara lain yaitu : <sup>4)</sup>*

- Delta dari General Electric untuk konsultasi kerusakan lokomotif.
- Prospector :
  - Merupakan sistem pakar klasik yang lainnya

- *Dikembangkan oleh Stanford Research Institute*
- *Digunakan untuk penaksiran prospek mineral, sehingga bisa membedakan kemungkinan informasi lokasi dan tipe dari dasar lubang endapan geologi di suatu tempat.*
- *Xycon*
  - *Dikembangkan oleh Digital Equipment Corp's .*
  - *Digunakan untuk mengkonfigurasi bagian-bagian komputer VAX.*
  - *Telah digunakan sejak 1981.*
- *Dendral:*
  - *Dikembangkan di Stanford pada akhir tahun 1960.*
  - *Dirancang untuk menduga informasi struktur dari formula-formula molukel-molekul organik dan banyak informasi spekto-grafik tentang kimia yang ditampilkan dalam molekul. Sebab molekul-molekul organik cenderung menjadi besar, jumlah ke-mungkinan struktur untuk molekul-molekul ini cenderung menjadi sangat besar.*
- *Mycin*
  - *Dikembangkan di Stanford pada pertengahan 1970.*
  - *Salah satu program pertama yang dialamatkan pada masalah pemikiran dengan ketidak pastian dan tidak lengkapnya informasi.*
- *Internist*
  - *Program untuk mendiagnosa penyakit dalam.*
- *Dipmeter Advisor*
  - *Digunakan untuk menafsirkan hasil pengeboran minyak.*
  - *Dibuat oleh Smith & Backer, tahun 1983.*

---

4). F. Luger, George, and A. Stubblefield William, "Artificial Intelligence : Structure and Strategies For Complex Problem Solving", Second Edition, page 16, Benyamin/Cummings Publishing Company, Inc., 1993.

Program kecerdasan tiruan ini dapat dilakukan dengan menggunakan suatu program paket, yaitu alat pengembangan sistem aplikasi pengetahuan (*knowledge system application development tool*) seperti:

- *VP-Expert*
- *PC PLUS*

- GURU
- JESS(*Java Expert System Shell*) Version 5.0  
<http://herzberg.ca.sandia.gov/jess/FAO.html>
- EXSYS, dan lain-lain.

Atau dengan menggunakan bahasa untuk pemrograman kecerdasan tiruan seperti :

- PROLOG (*Programming Logic*)
- WIN-PROLOG 4.040 (*Windows-Programming Logic*)  
<http://www.lpa.co.uk/web386/8f922bf1.zip>
- LISP(*Lisp Processing*)
- CLIPS( *C Language Integrated Production System*)  
<http://www.ghgcorp.com/clips/download/source/>

Lapangan dari *Artificial Intelligence* adalah gabungan beberapa area study, yaitu:<sup>5)</sup>

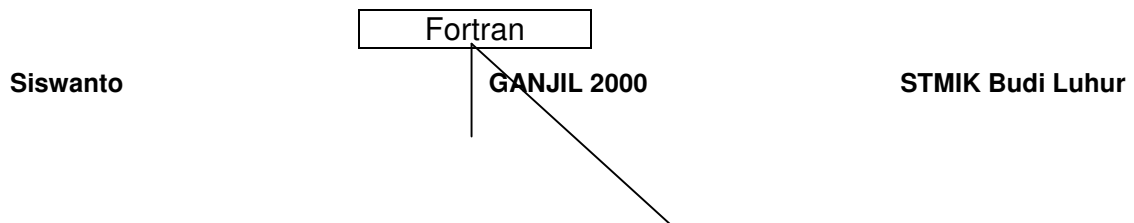
- ❖ *logic*
- ❖ *Searching*
- ❖ *Vision, Recognition dan Pattern Matching*
- ❖ *Natural Language Processing*
- ❖ *Expert System*
- ❖ *Robotik*
- ❖ *Learning*
- ❖ *Uncertainty dan Fuzzy Logic*

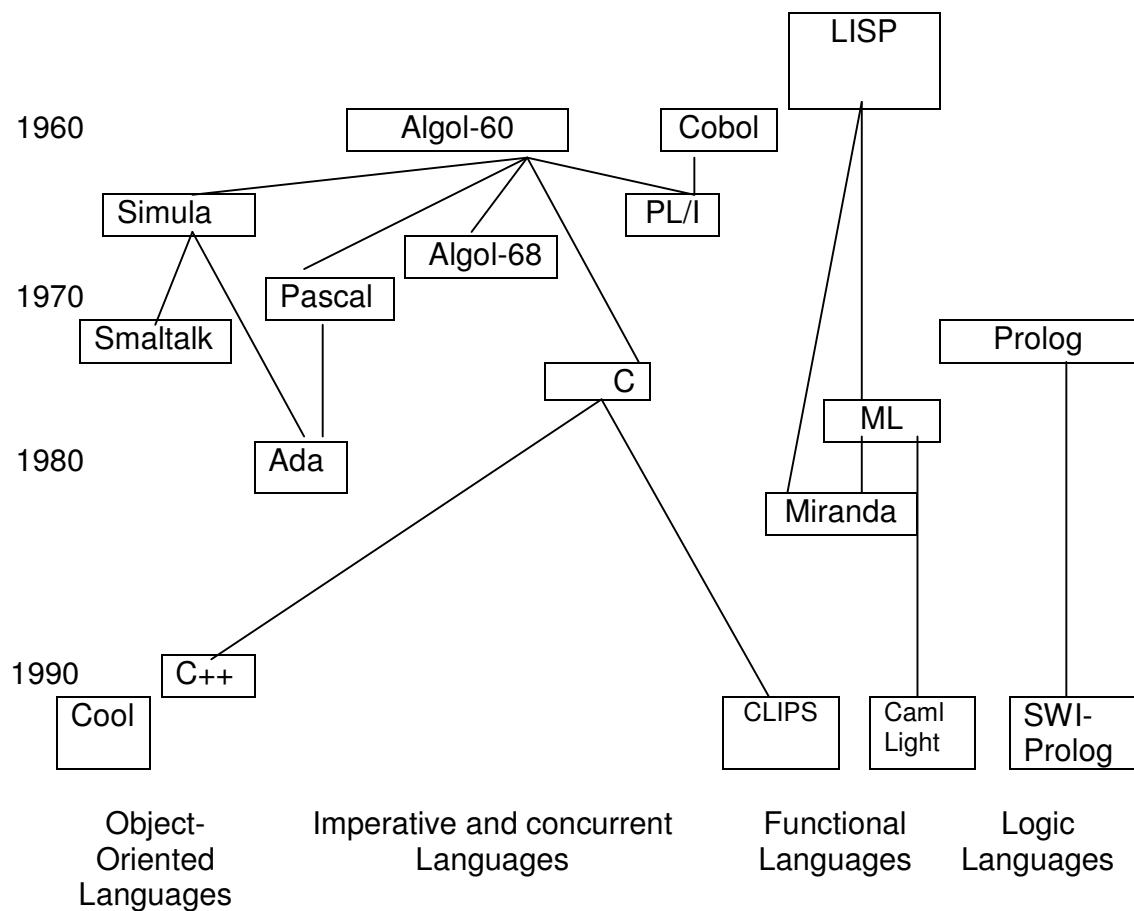
---

5) Schildt, Herbert, "Artificial Intelligence Using C", page 11, McGraw-Hill, 1987

Perangkat-perangkat lunak ini dapat dijalankan dengan komputer pribadi(PC), sehingga pengembangan untuk aplikasi kecerdasan tiruan dapat dilakukan dengan mudah dan dengan biaya yang murah.

1950





Gambar 1.3. Tahun dan Awal mulanya dari Bahasa Pemrogramman Utama  
(Sumber : David A. Watt, "Programming Language Concepts & Paradigms", page 4, Prentice-Hall, 1990.)

Penjelasan mengenai Lapangan dari *Artificial Intelligence*, sebagai berikut:

❖ **LOGIC**

Program dapat digunakan untuk mempelajari perbaikan logika dari sebuah argumen dengan menerapkan aturan logika standar.

❖ **SEARCHING**

Diterapkan pada AI mengacu pada pencarian untuk penyelesaian sebuah masalah.

❖ **VISION, RECOGNITION DAN PATTERN MATCHING**

Penting untuk beberapa aplikasi, termasuk robotik dan pengolah citra (*image processing*). Pada hal ini dibutuhkan untuk memperbolehkan komputer berhubungan secara langsung ke dunia dan manusia. Jika komputer

berhubungan secara menyeluruh dengan dunia manusia, maka dibutuhkan beberapa kemampuan bayangan (*vision*).

❖ *NATURAL LANGUAGE PROCESSING(NLP)*

Bagian yang paling sulit dari sasaran AI untuk mendapatkannya karena NLP memperbolehkan komputer untuk mengerti bahasa manusia secara langsung.

Permasalahan:

1. Ukuran kekompleksan bahasa manusia.
2. Mencoba untuk membuat komputer mengerti informasi secara konteks.

Produk komersial pertama dari AI yang memiliki 2 buah atribut :

- a. Diperbolehkan memasukkan informasi tentang subyek ke dalam komputer (*knowledge Base*/dasar pengetahuan)
- b. Menyelidiki *knowledge base* dan berlaku sebagai *expert* atau pakar pada subyek

❖ *ROBOTIK*

Digunakan untuk mempelajari mengontrol gerakan.

❖ *LEARNING*

Bertransaksi dengan pembuatan program yang belajar dari kesalahan dari observasi atau permintaan komputer mempunyai kemampuan untuk mengambil keuntungan dari pengalaman.

❖ *UNCERTAINTY(Ketidak Pastian) DAN FUZZY LOGIC*

Komputer dapat berpikir dengan menggunakan pengetahuan yang tidak lengkap dengan menerapkan penggunaan *Fuzzy Logic*.

#### 1.4. Teknik-teknik AI

Suatu teknik kecerdasan tiruan(AI) adalah sebuah metode yang memanfaatkan pengetahuan yang akan direpresentasikan sedemikian rupa, sehingga: <sup>6)</sup>

- Penyamaraan penangkapan pengetahuan.

*Dengan kata lain, dia tidak diperlukan untuk menampilkan secara terpisah menurut situasi individu. Malahan sifat berbagi situasi yang terpenting adalah kerjasama dalam kelompok atau grup. Jika pengetahuan tidak mempunyai sifat ini, jumlah memori banyak sekali dan memperbaruinya akan diperlukan. Jadi kita biasanya memanggil sesuatu tanpa sifat dari data daripada pengetahuan.*

- Dapat dimengerti oleh orang yang semestinya menyediakannya.

*Meskipun untuk beberapa program, bagian terbesar dari data dapat diperoleh secara otomatis, dalam beberapa daerah AI, pada akhirnya kebanyakan program AI semestinya disediakan oleh orang yang mengerti.*

- Dapat dengan mudah memodifikasi yang salah menjadi benar dan mencerminkan perubahan di dalam dunia dan di luar permukaan dunia.
- Dapat digunakan dalam beberapa situasi kejadian yang besar, jika ini tidak sepenuhnya akurat dan lengkap.
- Dapat digunakan untuk membantu menanggulangi permasalahan yang sangat penting dengan sendirinya, dengan bantuan seksama dari kemungkinan jarak yang semestinya dipertimbangkan.

Meskipun teknik-teknik AI seharusnya dirancang dengan menjaga ketidakleluasaan ditentukan oleh masalah-masalah AI, ada beberapa tingkatan diantara masalah-masalah dan teknik pemecahan masalah/persoalan. Memungkinkan saja dalam memecahkan persoalan AI tanpa menggunakan teknik-teknik AI (meskipun dianjurkan di atas, solusinya tidak baik). Dan mungkin saja digunakan teknik-teknik AI untuk memecahkan bukan persoalan AI. Ini seperti sesuatu yang baik dilakukan untuk persoalan-persoalan yang mempunyai beberapa persamaan karakteristik sebagai persoalan-persoalan AI. Urutan

**6) opcit <sup>1)</sup>, page 8.**

dalam mencoba mencirikan teknik-teknik AI dalam masalah yang berdiri sendiri sebagai kemungkinan jalan .

Tiga Teknik AI yang penting : <sup>7)</sup>

- 1). Pencarian/penelusuran(Search)

Menyediakan sebuah jalan untuk memecahkan persoalan yang lebih dari beberapa pendekatan tidak langsung yang tersedia, sebaik sebuah kerangka ke dalam beberapa teknik langsung yang dapat disimpan.

2). Penggunaan dari Pengetahuan(*Use of Knowledge*)

Memberikan sebuah jalan untuk memecahkan struktur-struktur dari objek yang dilibatkan.

3). Abstraksi(*Abstraction*)

Memberikan sebuah jalan yang mengutamakan pemisahan dan variasi dari beberapa yang tidak penting, kalau tidak meliputi beberapa proses.

Untuk solusi dari masalah yang sulit, program memanfaatkan teknik-teknik ini yang mempunyai beberapa keuntungan di atas atau bisa juga tidak. Mereka lebih sedikit ke pinggir, mereka akan mengeluarkan dengan lengkap oleh suatu gangguan kecil dalam masukan mereka. Orang dapat mudah mengerti, apakah pengetahuan program dan teknik ini dapat bekerja untuk masalah-masalah yang besar, di mana beberapa metoda perincian langsung.

Kita masih tidak memberikan suatu definisi khusus mengenai teknik AI, ini kemungkinan yang tidak mungkin dilaksanakan. Tetapi kita telah memberikan beberapa contoh apa yang bisa dan yang tidak bisa.

### 1.5. Tingkatan Dari Model

Sebelum mengatur untuk melakukan sesuatu, ini merupakan ide bagus untuk memutuskan dengan tepat apakah ini merupakan salah satu yang sedang diuji coba untuk melakukannya. Jadi kita semestinya bertanya kepada diri kita sendiri, apakah goal kita dalam uji coba untuk memproduksi program-

---

7) *opcit*<sup>1)</sup>, page 22.

program yang melakukan sesuatu kecerdasan seperti yang orang lakukan? Apakah kita mencoba untuk memproduksi program-program yang dapat melakukan tugas-tugas yang sama seperti manusia? atau, apakah kita mencoba untuk memproduksi program-program yang dapat melakukan secara sederhana

tugas-tugas apa saja dengan jalan memunculkannya dengan mudah? Ini telah memotivasi proyek-proyek AI oleh setiap goal ini.

Usaha untuk mengembangkan program-program yang melakukan tugas-tugas yang bisa dikerjakan manusia, dapat dibagi dalam 2 kelas:

- 1). Mencoba memecahkan persoalan yang tidak benar-benar patut/pantas sebagai definisi dari sebuah tugas AI.

Mereka merupakan masalah yang dapat dipecahkan dengan mudah oleh sebuah komputer, meskipun solusinya mudah, yang akan memanfaatkan mekanisme-mekanisme yang tidak dapat seperti manusia.

Sebuah contoh klasik untuk program kelas ini adalah:

EPAM (*Elementary Perceiver And Memorizer*), dengan memori yang dipetakan berpasangan dengan suku kata-suku kata omong kosong adalah mudah untuk sebuah komputer. Masukan EPAM sederhana, untuk menerima suatu relasi suku kata memberikan pendorong yang menghubungkan dengan yang satunya. Lalu pengamatan untuk mendorong salah satunya dan sebagai aki-batnya disimpan pada berikutnya, tetapi tugas ini sulit untuk orang. EPAM mensimulasikan sebuah jalan buat seseorang untuk bisa meningkatkan tugasnya.

Program ini dapat, bilamana alat-alat pembantu untuk ahli Psikologi yang ingin mengetes teori kemampuan manusia.

- 2). Mencoba kemampuan model manusia, merupakan sesuatu yang lebih jelas dilakukan tanpa definisi-definisi dari tugas AI, maka melakukan sesuatu yang tidak sepele\sederhana buat kom-puter.

Ada beberapa alasan untuk kemampuan model manusia yang menginginkan urutan-urutan tugas :

- a). Ujicoba teori psikologi dari kemampuan manusia

*Sebuah contoh menarik dari sebuah program yang telah ditulis untuk alasan ini adalah PARRY, yang merupakan sebuah model kelakuan manusia gila ketakutan. Model ini cukup baik, beberapa ahli psikologi diberikan kesempatan untuk mengkonversikan*



*dengan program lewat sebuah terminal, mereka mendiagnosa gila ketakutan.*

- b). Memungkinkan komputer mengerti jalan pikiran manusia

*Sebagai contoh, untuk sebuah komputer yang dapat memba-ca sebuah cerita surat kabar dan lalu menjawab sebuah pertanyaan, seperti: Apakah resiko mengadakan invasi? Program ini semestinya dapat mensimulasikan alasan proses dari orang.*

- c). Memungkinkan orang mengerti alasan komputer

*Dalam beberapa lingkungan, seseorang akan menjadi engan untuk mengandalkan keluaran dari sebuah komputer. Tidak sedikit dari mereka akan mengerti bagaimana mesin tiba pada hasilnya. Jika proses jalan pikiran komputer sama dengan orang, lalu produksi dengan sebuah penjelasan yang dapat diterima, ini akan menjadi mudah.*

- d). Memanfaatkan apakah pengetahuan kita akan dapat dikumpulkan dari sedikit orang

*Sejak orang merupakan pembentuk pengertian terbaik dari beberapa tugas dengan perputaran, membuat sejumlah rasa, untuk melihat mereka untuk menunjukkan bagaimana prosesnya.*

Kita memerlukan metoda-metoda untuk membantu kita memecahkan secara serius dilema-dilema atau persoalan AI :

- a) Sebuah sistem kecerdasan tiruan yang semestinya terdiri dari sejumlah pengetahuan, jika ia untuk menangani sesuatu tetapi masalah-masalah permainan sepele.
- b) Tetapi sebagai perkembangan sejumlah pengetahuan, ia menjadi sulit untuk mengakses sesuatu yang tepat ketika diperlukan. Jadi beberapa pengetahuan semestinya ditambahkan untuk mem-bantu, tetapi sekarang ada beberapa pengetahuan untuk me-nanganinya, jadi beberapa semestinya ditambah dan sejauh mungkin.

Dalam goal AI kita, dikontruksikan/dibangun suatu program yang kerjanya memecahkan masalah yang kita anggap menarik.

Kecerdasan Tiruan(AI) masih merupakan disiplin ilmu yang muda. Kita telah mempelajari banyak hal, tetapi ini masih sulit untuk diketahui secara tepat gambaran dari yang semestinya digambarkan.

### 1.6. Representasi Persoalan/Masalah

Kita akan menerangkan 3(tiga) langkah utama yang termasuk untuk membangun sebuah sistem untuk memecahkan suatu masalah-masalah khusus:

1. Definisikan masalah/persoalan secara jelas Definisi ini termasuk kejelasan spesifikasi dari situasi harga awal(initial) atau start *state* akan lebih baik bahwa situasi akhir atau goal *state* berpedoman pada solusi yang pantas untuk persoalannya
2. Analisa Masalah
3. Sebagian kecil dari ciri-ciri penting dapat berpengaruh yang luas sekali pada kelayakan dari berjenis-jenis teknik yang mungkin untuk memecahkan masalah/persoalan.

Pilih teknik terbaik dan pakai teknik tersebut untuk masalah khusus.

Dasar untuk aplikasi-aplikasi AI pada umumnya adalah penyelesaian masalah (*Problem Solving*).

Dua tipe dari masalah:

- a. tipe pertama dapat diselesaikan dengan menggunakan beberapa tipe prosedur deterministik, yaitu menjamin keberhasilan dan disebut: *Computation*
- b. tipe kedua terdiri dari masalah-masalah yang diselesaikan oleh pencarian(*searching*) untuk penyelesaian.

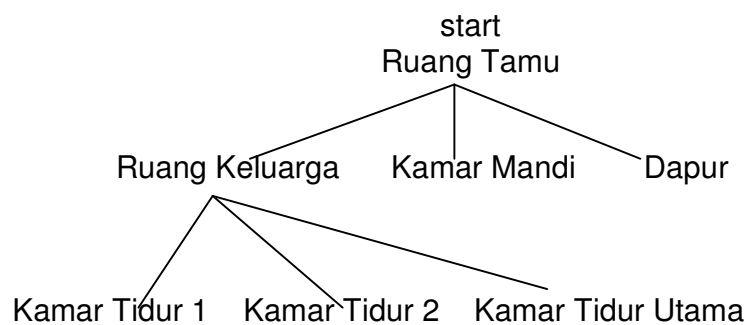
Contoh:

Bila terdapat denah rumah seperti gambar 1.4, Si Dul akan ujian Kalkulus di STMIK Budi Luhur, baru saja ia meninggalkan rumahnya dalam jarak 100 meter, ia teringat kalkulatornya tertinggal. Dan ia kembali ke rumahnya untuk mencari kalkulatornya.

Kamar Tidur 2	Kamar Tidur 1	Kamar Mandi	Dapur
Ruang Keluarga			
Kamar Tidur	Utama	Ruang	Tamu

Untuk memudahkan si Dul mencari kalkulator di rumahnya yang dimulai dari ruang tamu hingga dapur, maka dibuatlah graph sebagai berikut:

GRAPH:



Gambar 1.4. Denah Rumah dan Graph kemungkinan pencarian

#### 1.6.1. Representasi Tempat Kosong (*Representation in state space*)

Dalam *state space representation* dari sebuah masalah, node-node atau titik-titik dari sebuah graph korespondensi ke solusi-solusi tetap persoalan khusus dan tangan-tangan bersentuhan atau korespondensi ke langkah-langkah proses pemecahan sebuah persoalan.

Sebuah *state awal*(*start state*), korespondensi untuk memberikan informasi dalam cakupan sebuah masalah, bentuk akar(*root*) dari graph.

Graph juga mendefinisikan kondisi tujuan(*State Goal*), yang merupakan solusi dari cakupan sebuah persoalan.

*State Space* mencari karakteristik pemecahan persoalan, sebagai proses dari penemuan sebuah path/jalan kecil/garis edar solusi dari *state awal* sampai menghasilkan goal.

Sebuah goal boleh mendeskripsikan sebuah *state* atau ketetapan, seperti papan kemenangan Tic-Tac-Toe atau konfigurasi *Goal* dalam *8-Puzzle*.

Contoh : 8-Puzzle<sup>8)</sup>

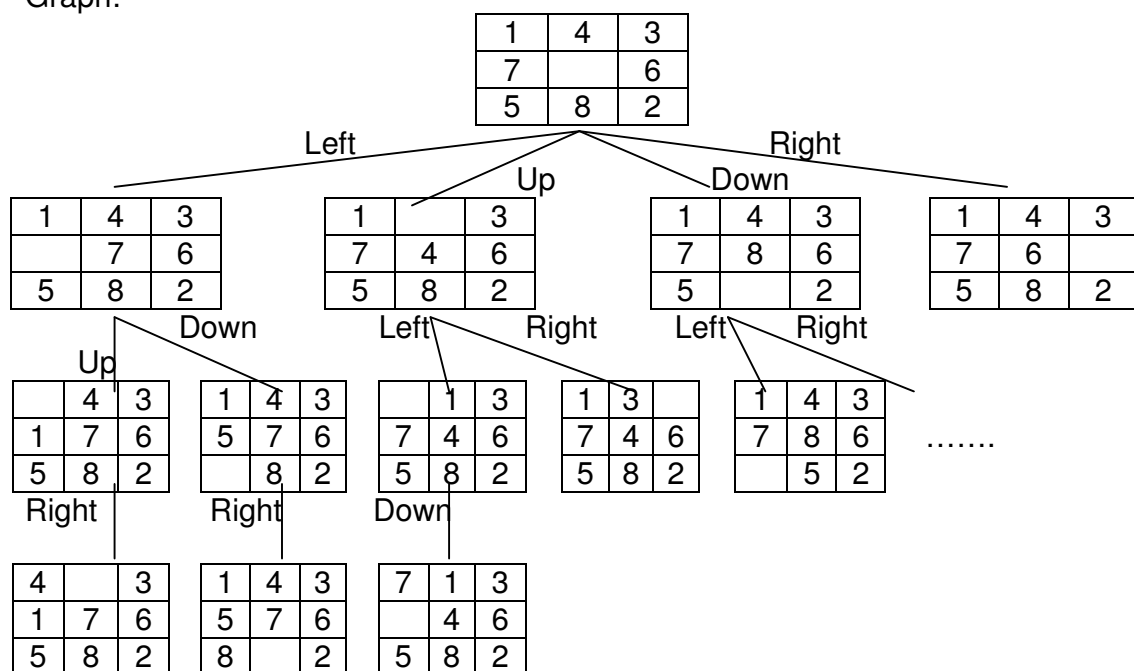
Start State :

1	4	3
7		6
5	8	2

Goal State :

1	2	3
8		4
7	6	5

Graph:



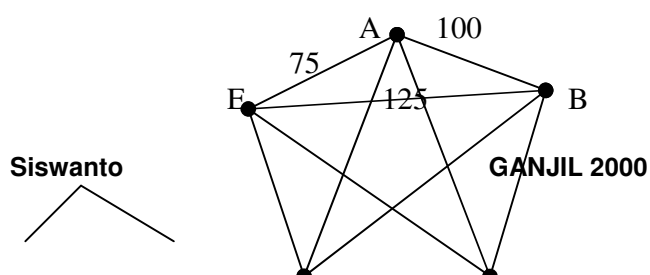
Gambar 1.5. Graph pencarian 8-Puzzle dengan memindahkan tempat kosong

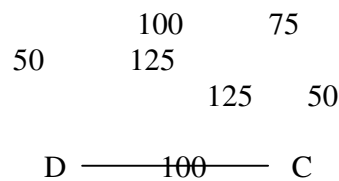
Dengan alternatif, sebuah goal dapat menggambarkan beberapa ketentuan dari solusi path sendiri.

8) OPCIT<sup>1)</sup>, page 47.

Dalam masalah perjalanan seorang pedagang (*Travelling Sales Person Problem*) dapat digambarkan seperti gambar 1.6.

Contoh : Search pada Travelling Salesperson Problem





Gambar 1.6. Jangkauan dari masalah perjalanan seorang pedagang

Pencarian berakhir ketika path terpendek ditemukan, yang mencakup semua titik(node) dari graph. Dalam masalah parsing, path analisis berhasil dari sebuah indikasi berakhir.

Hubungan dari kedudukan tempat kosong(*state Space*) korespondensi ke langkah-langkah proses solusi dan path melalui solusi representasi tempat kosong. Dalam berbagai tingkatan path lengkap dicari, mula-mula pada awal *state* dan dilanjutkan ke graph, sampai deskripsi goal didapatkan atau mereka batalkan.

#### Definisi Pencarian *state space*

Sebuah *state space* direpresentasikan dengan 4 komponen [N,A,S,GD] di mana:

**N**, adalah himpunan dari node atau *state* dari graph yang korespondensi ke *state-state* dalam sebuah proses pemecahan persoalan.

**A**, adalah himpunan dari hubungan/arc/links antara node-node yang korespondensi ke langkah-langkah dalam sebuah proses pemecahan persoalan.

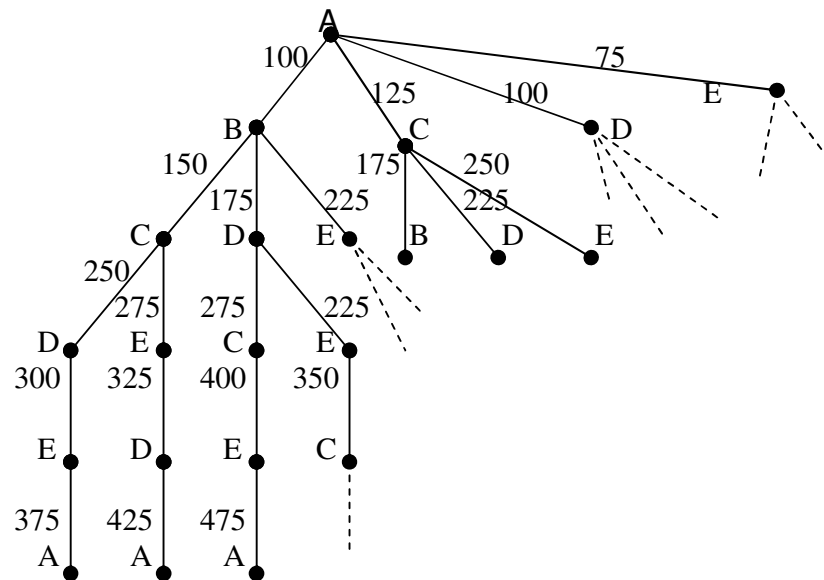
**S**, adalah himpunan bagian yang tidak kosong dari N, terdiri dari *state* awal(*Start State*) dari masalah.

**GD**, adalah himpunan bagian yang tidak kosong dari N, terdiri dari *Goal State* dari masalah. *State* dalam GD dideskripsikan menggunakan: ukuran ketentuan/rule dari *state-state* yang ditemukan dalam pencarian dan ketentuan dari pengembangan path.

*Path Solusi* adalah sebuah path yang melalui graph ini dari sebuah node dalam S ke sebuah Node dalam GD.

Gambar 1.7. menceritakan proses pencarian dari *Travelling Sales Person Problem*, setiap hubungan ditandai dengan total biaya dari semua path yang dilalui dari node awal (A) ke node akhir.

GRAPH:



Gambar 1.7. Pencarian dari *Travelling Sales Person Problem*

Kesimpulan :

Path:  
ABCDEA  
Biaya: 375

Path:  
ABCEDA  
Biaya: 425

Path:  
ABDCEA  
Biaya: 475

Kompleksitas dari pencarian yang mendalam dalam *travelling Sales person problem* adalah  $(n-1)!$ , di mana N adalah jumlah kota dalam graph. Untuk 9 kota, kita bisa mencoba secara mendalam semua path, tetapi untuk beberapa masalah menyangkut ukuran kepentingan, seperti 50 kota, pencarian tidak baik kemampuannya tanpa sebuah kriteria waktu/lamanya perjalanan.

#### 1.6.2. Pengetahuan(*Knowledge*) Untuk Pemecahan Persoalan

Contoh :

***“The bank president ate a dish of pasta salad with the fork”***

Dari kalimat di atas, tentu persepsi dari beberapa orang yang membacanya akan berbeda dari pengetahuan yang dimilikinya. Dan pengetahuan itu diperoleh dari pengalaman orang tersebut terhadap suatu objek. Sehingga ada anggapan sebagai berikut:

- Kata “*bank*” bisa dihubungkan pada suatu institusi keuangan atau pada suatu pingiran/tepi dari sebuah sungai, tetapi hanya satu yang mempunyai seorang presiden.
- Kata “*dish*” adalah objek dari kata kerja “*eat*”. Ini mungkin saja bahwa sebuah piring(*a dish*) dimakan, tetapi lebih disukai yang dimakan adalah *pasta salad* di dalam piring.
- *Pasta salad* adalah suatu *salad* yang mengandung *pasta*, tetapi di sini ada pengertian lain yang dapat dibentuk dari padanan kata benda. Seperti : “*dog food*”, secara normal tidak mengandung anjing/*dog*.
- Ungkapan “*with the fork*” dapat dimodifikasi menjadi beberapa bagian dari kalimat. Dalam kasus ini modifikasi kata kerja “*eat*”. Tetapi, jika ungkapan telah diubah “*with vegetables*”, maka perbedaannya terletak pada modifikasi struktur. Dan jika ungkapan “*with her friend*”, struktur masih berbeda.

#### PERBEDAAN PEMROGRAMAN KONVENSIONAL DENGAN AI

		<b>KONVENSIONAL PROGRAMMING</b>	<b>AI PROGRAMMING</b>
1	Yang Disimpan	Data	Pengetahuan
2	Yang Diolah	Angka, String --> Kuantitatif	Simbol-simbol Pengertian --> Kualitatif
3	Masalah	Menyelesaikan masalah	Menentukan cara solusi

4	Dasar Kerja	Dengan Algoritma	Manipulasi Simbol
5	Cara Kerja	Langkah demi Langkah -- > hasil	Melacak Pengetahuan --> Jawaban
6	Tingkat Kerja	Perhitungan	Pengambilan Konklusi
7	Kemampuan Nalar	Ada Tidak punya kemampuan mengambil kesimpulan	Tidak Ada Punya kemampuan mengambil kesimpulan (Inferensi)
8	Kebutuhan Memori	Cukup	Sangat Besar
			Komputer

Catatan :

- Dalam Implementasi AI Programming, algoritma masih diperlukan tapi bukan dasar kerja.
- Penyelesaian masalah dalam AI Programming --> melacak pengetahuan.

## BAB II LOGIC