



# MASALAH & RUANG KEADAAN DALAM KECERDASAN BUATAN

# OUTLINE

- Representasi Masalah
- Representasi Ruang Keadaan
- Karakteristik Masalah



# MEMBANGUN SISTEM MENYELESAIKAN MASALAH KECERDASAN BUATAN

- Mendefinisikan masalah dengan tepat (keadaan awal dan solusi yang diharapkan)
- Menganalisis dan mencari teknik penyelesaian masalah
- Merepresentasikan pengetahuan
- Memilih teknik penyelesaian masalah yang terbaik

## I.4 MENDEFISIKAN SUATU MASALAH DENGAN BAIK

- Definisikan/buat '*state space*' atau *ruang masalah*
- Tentukan keadaan awal (*initial state*)
- Tentukan keadaan akhir/tujuan (*goal state*)
- Tentukan operatornya/aturannya

# Contoh

Mendefinisikan masalah sebagai suatu ruang keadaan dalam “**Permainan Catur**”, maka harus ditentukan :

- Posisi awal pada papan catur
- Aturan-aturan untuk melakukan gerakan secara legal
- Tujuan (*goal*)

# Penyelesaian Permainan Catur:

- posisi awal pada papan catur posisi awal setiap permainan catur selalu sama, yaitu semua bidak diletakkan di atas papan catur dalam 2 sisi, yaitu kubu putih dan kubu hitam.
- Aturan – aturan untuk melakukan gerakan, aturan – aturan ini sangat berguna untuk menentukan gerakan suatu bidak, yaitu melangkah dari satu keadaan ke keadaan lain. Misalkan untuk mempermudah menunjukkan posisi bidak, setiap kotak ditunjukkan dalam huruf (a,b,c,d,e,f,g,h) pada arah horisontal dan angka (1,2,3,4,5,6,7,8) pada arah vertikal. Suatu aturan untuk menggerakkan bidak dari posisi (e,2) ke (e,4) dapat ditunjukkan dengan aturan :
 

IF	bidak putih pada kotak(e,2), and kotak(e,3) kosong, and kotak(e,4) kosong
Then	gerakkan bidak dari (e,2) ke (e,4)
- Tujuan (goal)  
Tujuan yang ingin dicapai adalah posisi pada papan catur yang menunjukkan kemenangan seseorang terhadap lawannya. Kemenangan ini ditandai dengan posisi raja yang sudah tidak dapat bergerak lagi.

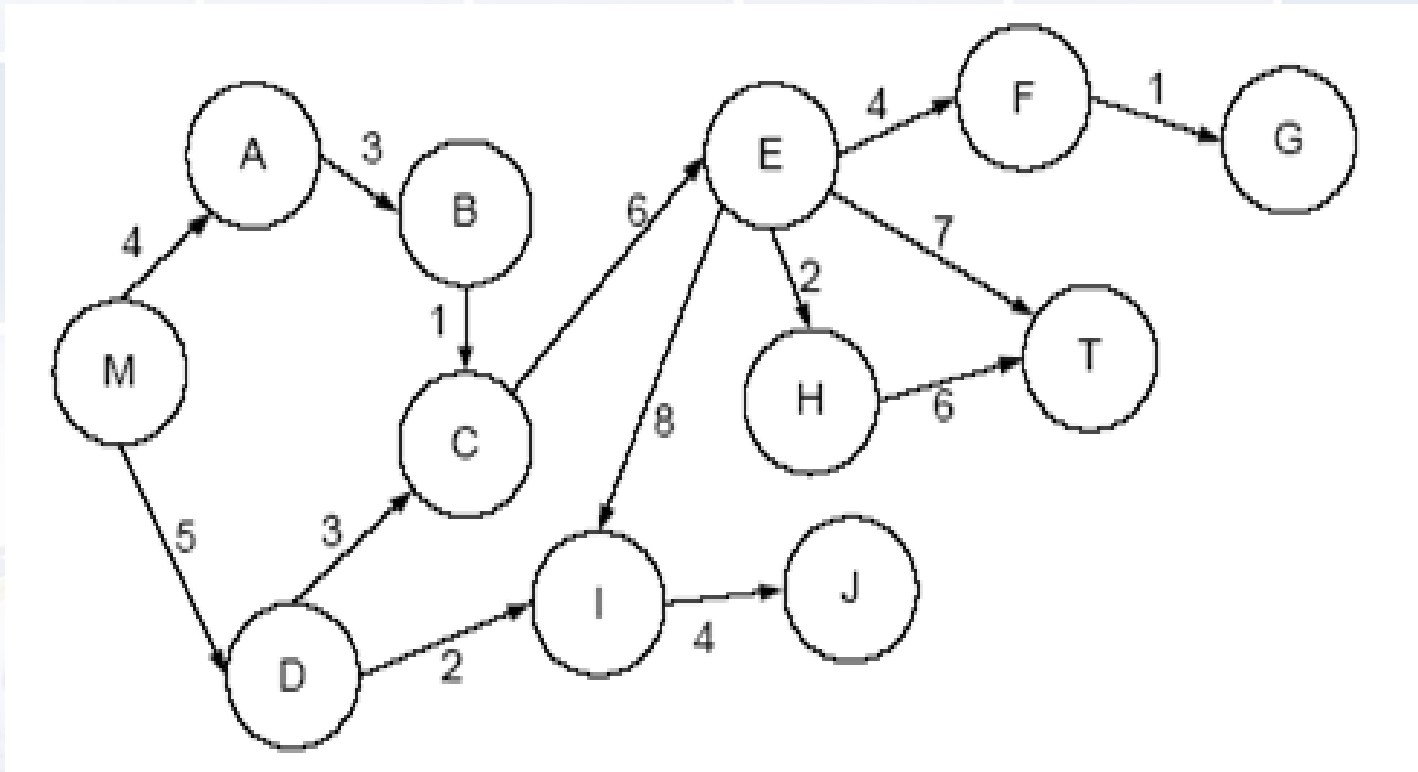
## Penyelesaian Permainan Catur:

Contoh tersebut menunjukkan representasi masalah dalam Ruang Keadaan (State Space), yaitu suatu ruang yang berisi semua keadaan yang mungkin. Kita dapat memulai bermain catur dengan menempatkan diri pada keadaan awal, kemudian bergerak dari satu keadaan ke keadaan yang lain sesuai dengan aturan yang ada, dan mengakhiri permainan jika salah satu telah mencapai tujuan.

# BEBERAPA CARA MEREPRESENTASIKAN RUANG MASALAH

- Graph keadaan, terdiri dari node-node yang menunjukkan keadaan yaitu keadaan awal dan keadaan baru yang akan dicapai dengan menggunakan operator.
- Pohon Pelacakan, pelacakan dengan struktur pohon yang menggambarkan keadaan secara hirarkis.
- Pohon AND/OR

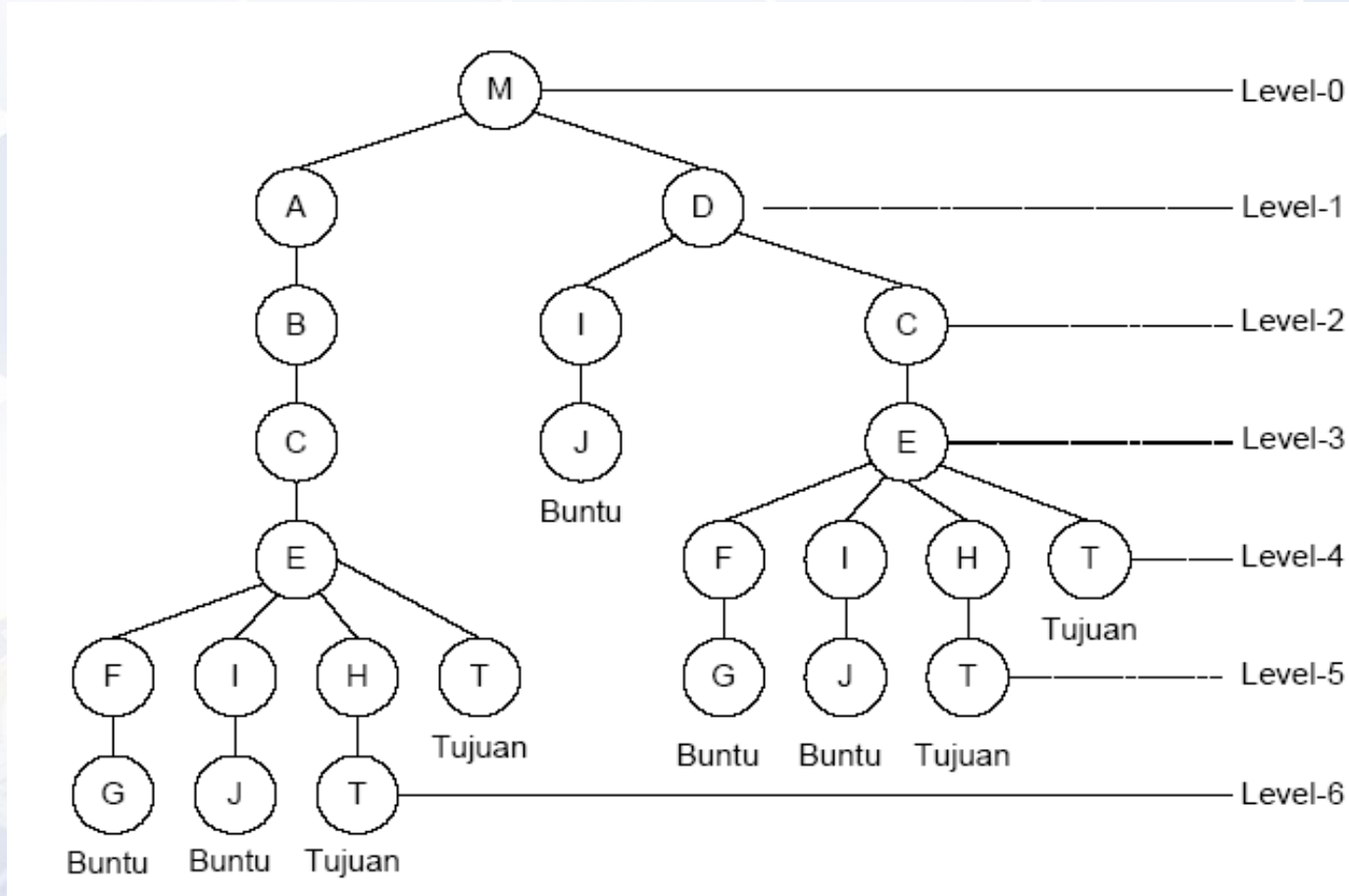
# Contoh Graph Keadaan



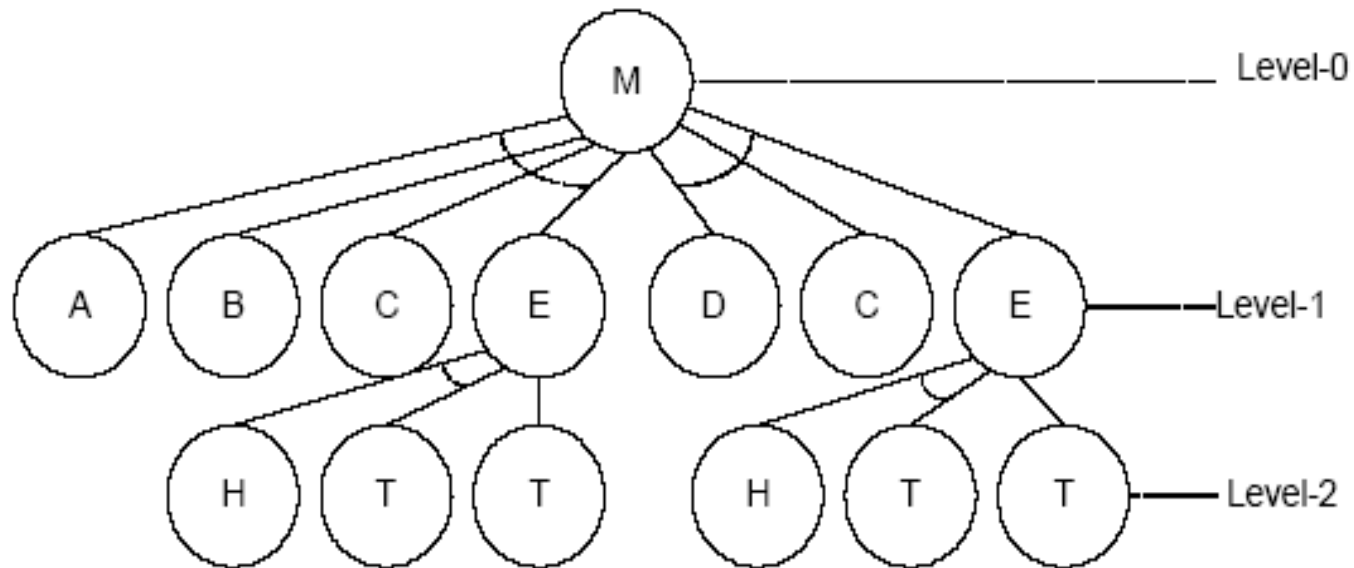
# Contoh Graph Keadaan

Graph berarah dengan satu tujuan (T)	Graph berarah yang menemui jalan buntu	Graph dengan siklus
<ul style="list-style-type: none"> <li>• M-A-B-C-E-T</li> <li>• M-A-B-C-E-H-T</li> <li>• M-D-C-E-T</li> <li>• M-D-C-E-H-T</li> </ul>	<ul style="list-style-type: none"> <li>• M-A-B-C-E-F-G</li> <li>• M-A-B-C-E-I-J</li> <li>• M-D-C-E-F-G</li> <li>• M-D-C-E-I-J</li> <li>• M-D-I-J</li> </ul>	D-E-C-E-I-D

# Contoh Pohon Pelacakan



# Contoh Pohon AND/OR



# KARAKTERISTIK MASALAH/PROBLEMA

1. Apakah masalah dapat dipilah-pilah (*decompose-able*) menjadi sejumlah sub-masalah independent yang lebih kecil atau lebih mudah ?
2. Dapatkah langkah-langkah penyelesaian yang terbukti tidak tepat diabaikan ?
3. Apakah ruang lingkup atau semesta pembicaraan masalah dapat diprakirakan ?

# Karakteristik Masalah/Problema

4. Apakah solusi masalah yang baik telah dibandingkan dengan semua solusi yang dimungkinkan ?
5. Apakah basis pengetahuan yang digunakan untuk memecahkan masalah bersifat konsisten ?



# Karakteristik Masalah/Problema

6. Apakah benar-benar dibutuhkan sejumlah besar informasi untuk memecahkan masalah yang sedang dihadapi, atau pengetahuan hanya penting untuk membatasi proses pencarian (*searching*) ?
7. Apakah sebuah komputer sendirian dapat diberi masalah dan kemudian menyajikan solusi secara sederhana, atau akankah solusi dari suatu masalah membutuhkan interaksi antara komputer dan manusia ?

☺ END ☺

