

## TEKNIK PENCARIAN HEURISTIK (HEURISTIC SEARCHING)

- Teknik pencarian heuristik (*heuristic searching*) merupakan suatu strategi untuk melakukan proses pencarian ruang keadaan (*state space*) suatu problema secara selektif, yang memandu proses pencarian yang kita lakukan di sepanjang jalur yang memiliki kemungkinan sukses paling besar, dan mengesampingkan usaha yang bodoh dan memboroskan waktu.
- Heuristik adalah sebuah teknik yang mengem-bangkan efisiensi dalam proses pencarian, namun dengan kemungkinan mengorbankan kelengkapan (*completeness*).
- Untuk dapat menerapkan heuristik tersebut dengan baik dalam suatu domain tertentu, diperlukan suatu Fungsi Heuristik.
- Fungsi heuristik ini digunakan untuk mengevaluasi keadaan-keadaan problema individual dan menentukan seberapa jauh hal tersebut dapat digunakan untuk mendapatkan solusi yang diinginkan.
- Jenis-jenis *Heuristic Searching*:
  - ◆ *Generate and Test.*
  - ◆ *Hill Climbing.*
  - ◆ *Best First Search.*
  - ◆ *Alpha Beta Prunning.*
  - ◆ *Means-End-Anlysis.*
  - ◆ *Constraint Satisfaction*

## **GENERATE AND TEST**

- Strategi bangkitkan dan uji (*generate and test*) merupakan pendekatan yang paling sederhana dari semua pendekatan yang akan dibicarakan.
- Pendekatan ini meliputi langkah–langkah sebagai berikut :
  1. **Buatlah/bangkitkan sebuah solusi yang memungkinkan. Untuk sebuah problema hal ini dapat berarti pembuatan sebuah titik khusus dalam ruang problema.**
  2. **Lakukan pengujian untuk melihat apakah solusi yang dibuat benar–benar merupakan sebuah solusi, dengan cara membandingkan titik khusus tersebut dengan *goal*-nya (solusi).**
  3. **Jika telah diperoleh sebuah solusi, langkah–langkah tersebut dapat dihentikan. Jika belum, kembalilah ke langkah pertama.**
- Jika pembangkitan atau pembuatan solusi–solusi yang dimungkinkan dapat dilakukan secara sistematis, maka prosedur ini akan dapat segera menemukan solusinya, (bila ada).
- Namun, jika ruang problema sangat besar, maka proses ini akan membutuhkan waktu yang lama.
- Metode *generate and test* ini kurang efisien untuk masalah yang besar atau kompleks.

## HILL CLIMBING

- *Hill climbing (mendaki bukit)* merupakan salah satu variasi metode buat dan uji (*generate and test*) dimana umpan balik yang berasal dari prosedur uji digunakan untuk memutuskan arah gerak dalam ruang pencarian (*search*).
- Dalam prosedur buat dan uji yang murni, respon fungsi uji hanyalah *ya* atau *tidak*.
- Dalam prosedur *Hill Climbing*, fungsi uji dikombinasikan dengan fungsi heuristik yang menyediakan pengukuran kedekatan suatu keadaan yang diberikan dengan tujuan (*goal*).

### Prosedur *Hill Climbing* :

1. Buatlah solusi usulan pertama dengan cara yang sama seperti yang dilakukan dalam prosedur buat dan uji (*generate and test*). Periksa apakah solusi usulan itu merupakan sebuah solusi. Jika *ya*, berhentilah. Jika *tidak*, kita lanjutkan ke langkah berikutnya.

2. Dari solusi ini, terapkan sejumlah aturan yang dapat diterapkan untuk membuat sekumpulan solusi usulan yang baru.
3. Untuk setiap elemen kumpulan solusi tersebut, lakukanlah hal-hal berikut ini :
  1. Kirimkanlah elemen ini ke fungsi uji. Jika elemen ini merupakan sebuah solusi, berhentilah.
  2. Jika tidak, periksalah apakah elemen ini merupakan yang terdekat dengan solusi yang telah diuji sejauh ini. Jika tidak, buanglah.
4. Ambilah elemen terbaik yang ditemukan di atas dan pakailah sebagai solusi usulan berikutnya. Langkah ini bersesuaian dengan langkah dalam ruang problema dengan arah yang muncul sebagai yang tercepat dalam mencapai tujuan.
5. Kembalilah ke langkah 2.

**Masalah-masalah yang mungkin timbul pada prosedur *Hill Climbing* :**

- ***Maksimum lokal*** adalah suatu keadaan yang lebih baik daripada semua tetangganya namun masih belum lebih baik dari suatu keadaan lain yang jauh letaknya darinya.
- ***Daratan (Plateau)*** adalah suatu daerah datar dari ruang pencarian (*search*) dimana semua himpunan keadaan tetangganya memiliki nilai yang sama.
- ***Punggung (Ridge)*** adalah suatu daerah ruang pencarian (*search*) yang lebih tinggi daripada daerah sekitarnya, namun tidak dapat dibalikkan oleh langkah–langkah tunggal ke arah manapun.

### **Solusinya:**

- Melakukan langkah balik (*backtracking*) ke simpul yang lebih awal dan mencoba bergerak ke arah yang lain.
- Melakukan lompatan besar ke suatu arah untuk mencoba bagian ruang pencarian yang baru.

- Menerapkan dua atau lebih aturan sebelum melakukan uji coba. Ini bersesuaian dengan bergerak ke beberapa arah sekaligus.

## BEST FIRST SEARCH

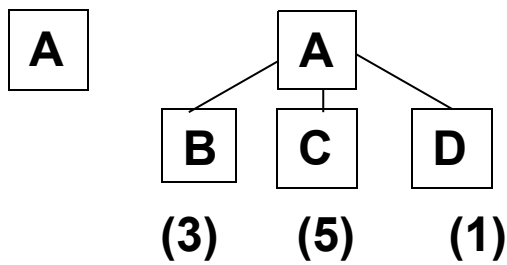
- Pencarian terbaik pertama (*Best First Search*) merupakan suatu cara yang menggabungkan keuntungan atau kelebihan dari pencarian *Breadth-First Search* dan *Depth-First Search*.
- Pada setiap langkah proses pencarian terbaik pertama, kita memilih node-node dengan menerapkan fungsi heuristik yang memadai pada setiap node/simpul yang kita pilih dengan menggunakan aturan-aturan tertentu untuk menghasilkan penggantinya.
- Fungsi Heuristik yang digunakan merupakan prakiraan (estimasi) *cost* dari *initial state* ke *goal state*, yang dinyatakan dengan :

$$f' = g + h'$$

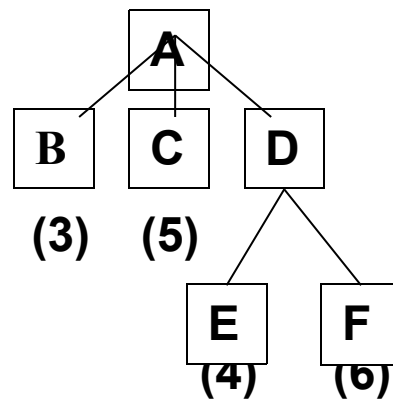
dimana  $f'$  = prakiraan *cost* dari *initial* ke *goal*  
 $g$  = *cost* dari *initial state* ke *current state*  
 $h'$  = prakiraan *cost* dari *current state* ke *goal state*

## Contoh Proses pada *Best First Search*:

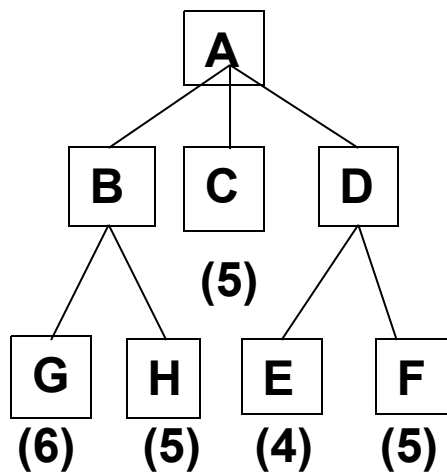
Langkah 1      Langkah 2



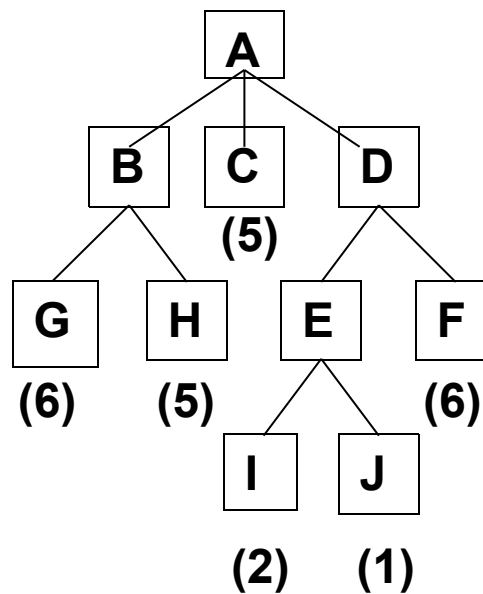
Langkah 3



Langkah 4



Langkah 5



**Gambar 1. Ilustrasi Pencarian Terbaik Pertama  
(*Best-First Search*)**