

# TREE





## Tree

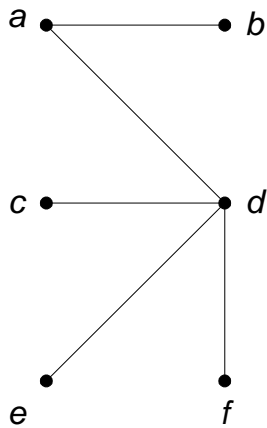
merupakan bentuk khusus dari graf (graf yang mempunyai sifat-sifat khusus) yang banyak digunakan dalam bidang ilmu komputer.

Tree paling sedikit mempunyai 1 vertek, sama seperti graf.

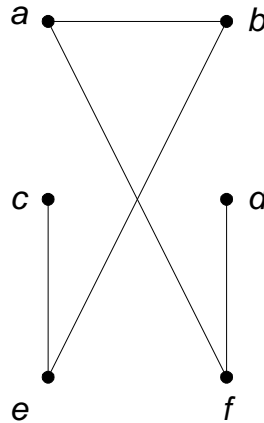


# Definisi Tree

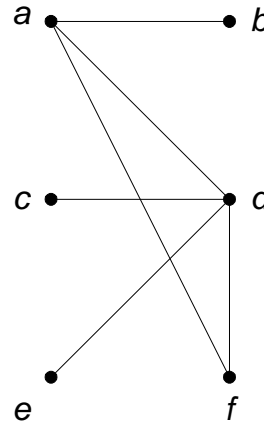
Tree adalah suatu graf tak berarah yang terhubung dan tidak mengandung sirkuit.



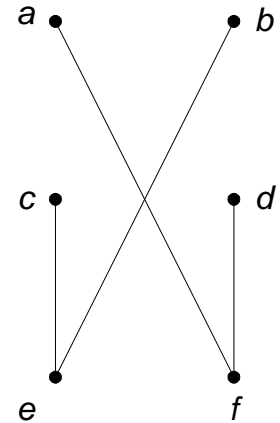
tree



tree



bukan tree



bukan tree



# Definisi Tree

Sifat - sifat tree :

- Ada sebuah jalur yang unik (path elementer) antara dua buah vertek,
- Dalam sebuah tree, jumlah vertek satu lebih banyak dari pada jumlah sisi,
- No cycles



# Istilah

- Leaf = Daun = Simpul Terminal

Leaf adalah vertek berderajat satu dalam sebuah tree

- Branch = Cabang = Simpul Internal

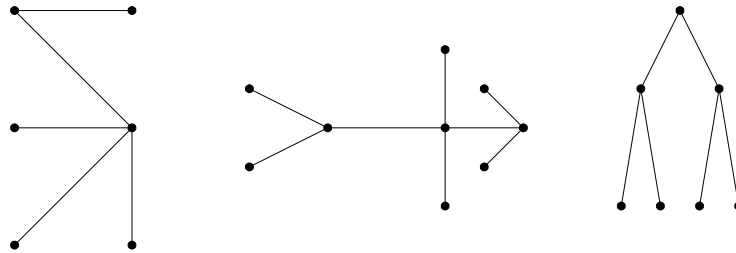
Branch adalah vertek berderajat lebih dari satu dalam sebuah tree.



# Istilah

- Hutan ( forest )

Hutan adalah Himpunan tree yang terpisah satu sama lain.



Contoh hutan yang terdiri dari 3 buah tree

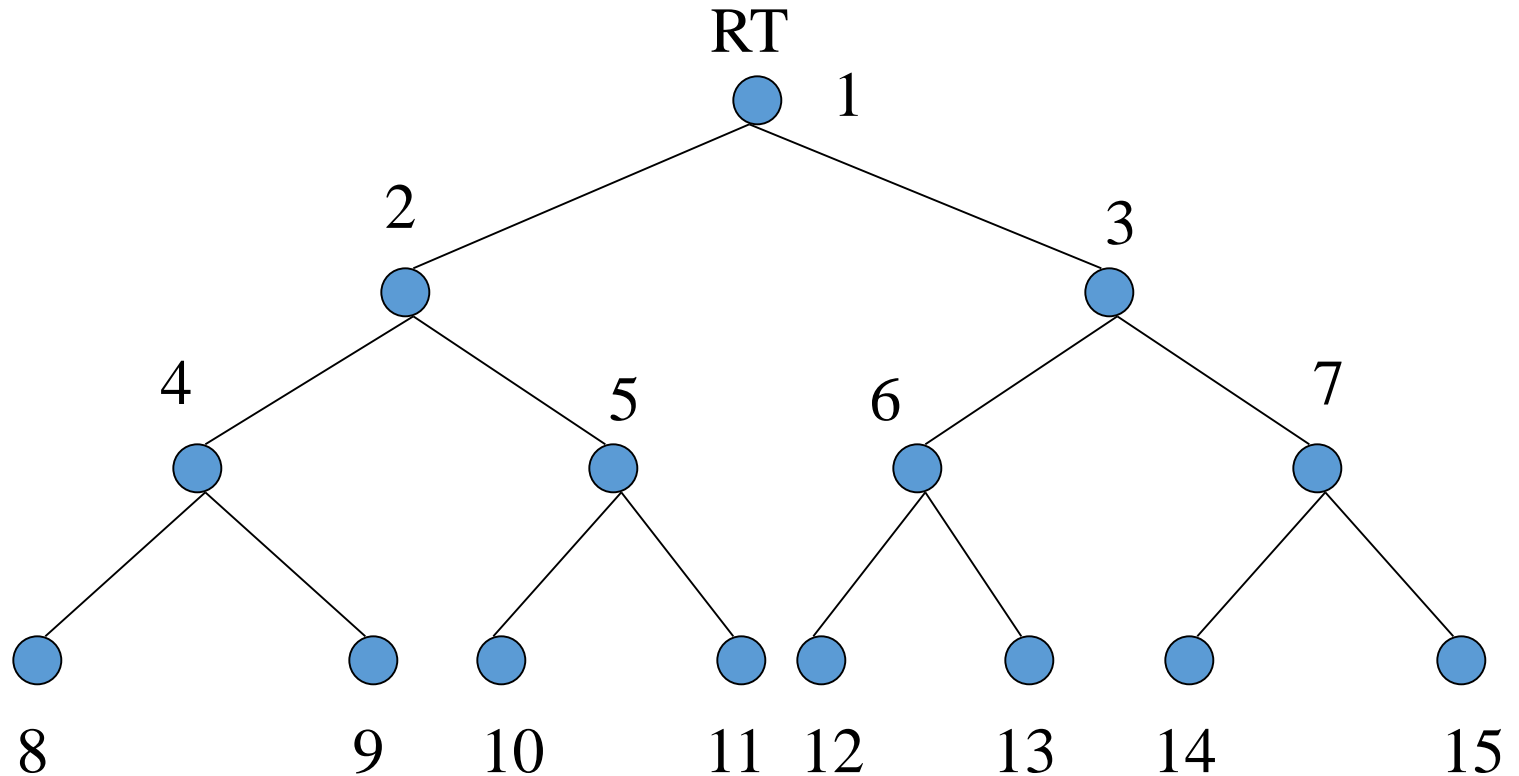
# Contoh Tree

Suatu komplek perumahan terdiri dari 15 keluarga. Suatu saat ketua RT ingin memberitahukan suatu berita pada seluruh warga komplek tersebut. Misal semua keluarga mempunyai telepon dan berita tersebut disebarluaskan dengan cara berikut :

Ketua RT hanya menelepon 2 keluarga, kemudian masing-masing keluarga yang telah ditelepon menelepon 2 keluarga lainnya yang belum ditelepon demikian seterusnya sampai dengan semua keluarga menerima berita tersebut.



## Contoh Tree ( *Lanjutan* )



Node no. 8 sampai no. 15 disebut **daun**.  
Node no. 1 sampai no. 7 disebut **cabang**

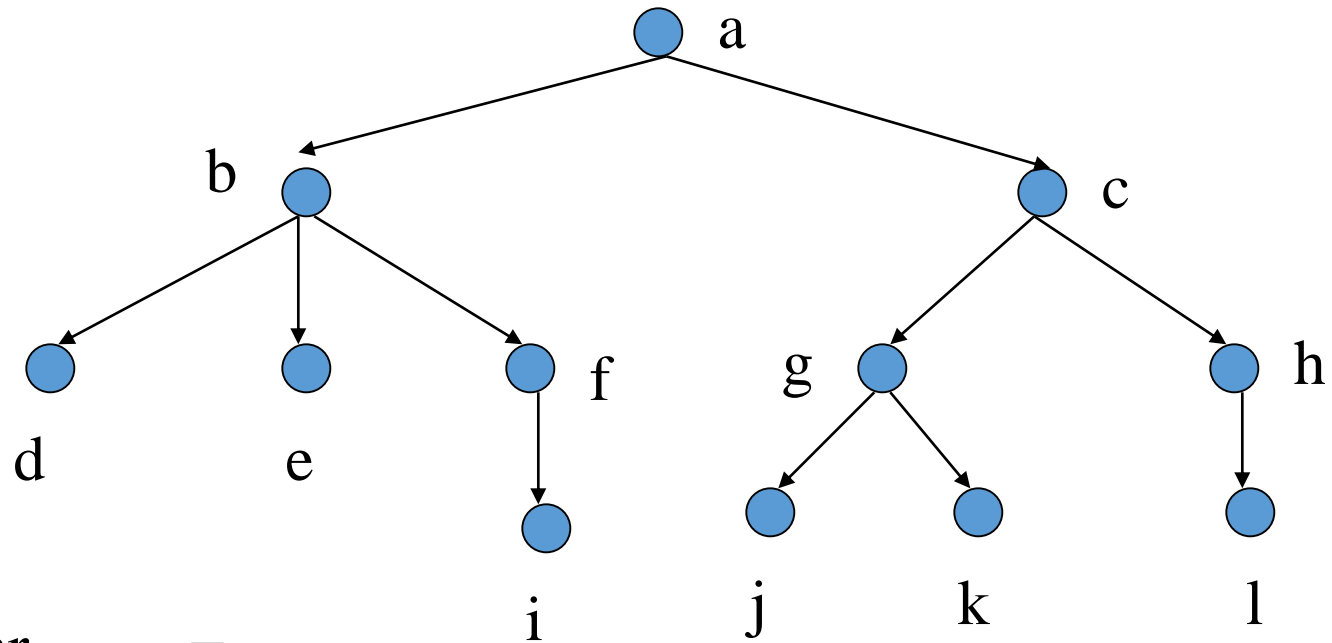




# Tree Berakar

- Daun = vertek yang mempunyai  $dout = 0$
- Cabang = vertek yang mempunyai  $dout \neq 0$ .
- Panjang lintasan suatu vertek dalam pohon berakar adalah banyaknya rusuk yang menyusun lintasan tersebut dari akar ke vertek.
- Tinggi suatu pohon berakar adalah panjang lintasan maksimum di dalam pohon tersebut.

Contoh : Tree berakar



Akar =

Cabang =

Daun =

Panjang lintasan vertek g =

Tinggi pohon =



# Tree Berarah & Tree Berakar ( *Lanjutan* )

- Anak (son)

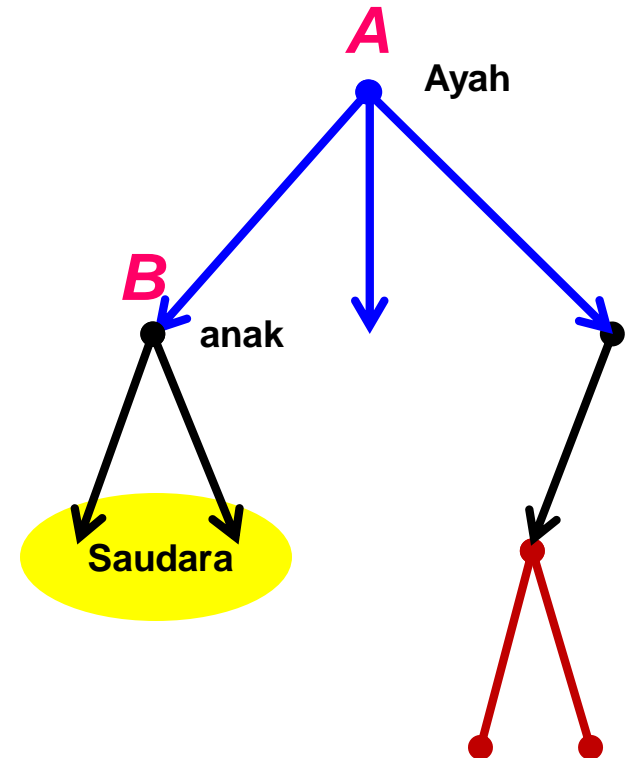
Vertek B disebut anak dari suatu vertek A bila ada rusuk berarah dari A ke B.

- Ayah (father)

Vertek A disebut ayah dari suatu vertek B bila ada rusuk berarah dari A ke B.

- Saudara (brother / sibling)

Dua buah vertek dikatakan bersaudara bila mempunyai vertek ayah yang sama.



# Tree Berarah & Tree Berakar ( *Lanjutan* )

- Keturunan (descendant)

Vertek B disebut keturunan dari vertek A bila ada jalur berarah dari A ke B.

- Leluhur (Ancestor)

Vertek A disebut leluhur dari vertek B bila ada jalur berarah dari A ke B.

# Binary Tree = Pohon Biner

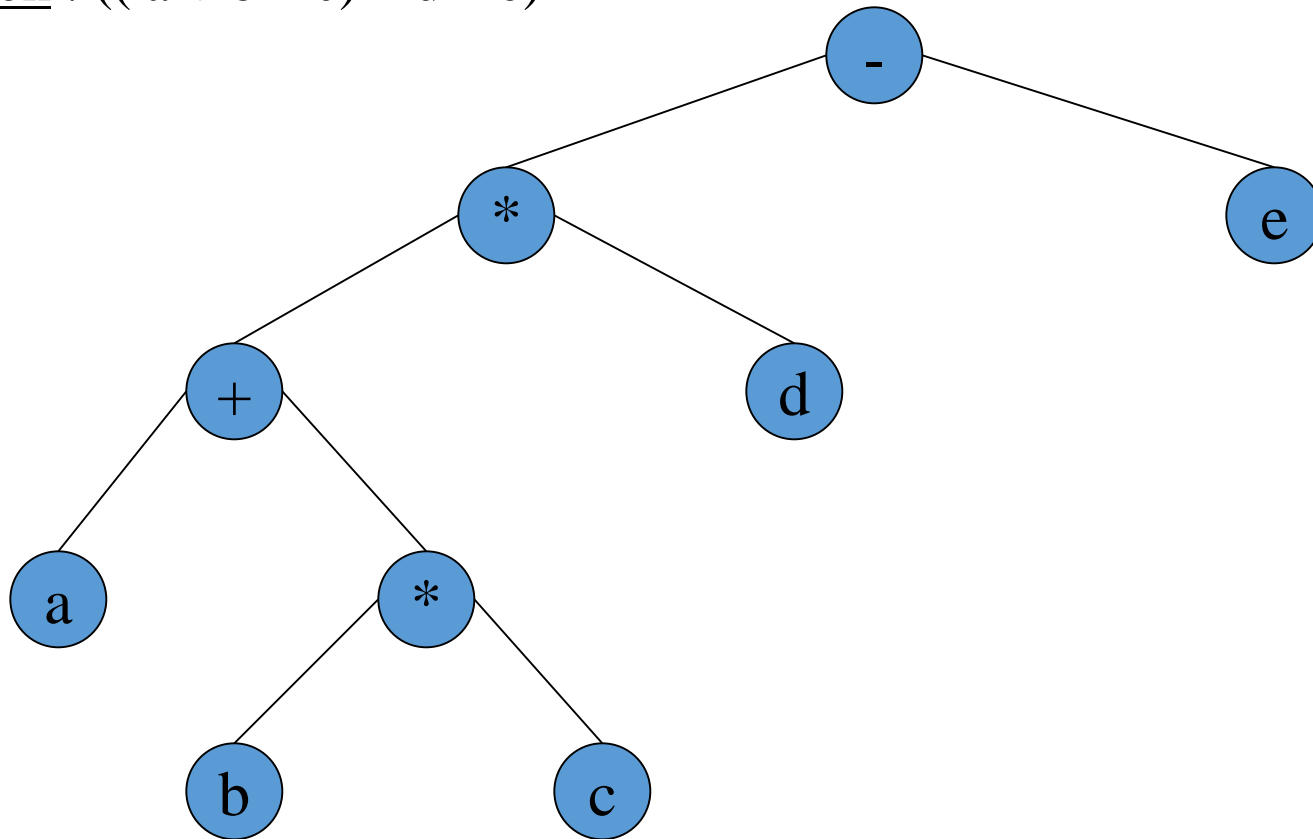
Pohon biner adalah suatu pohon di mana setiap cabangnya mempunyai maksimum 2 anak ( $m=2$ ).

Anak pohon pertama disebut **anak pohon kiri (left subtree)**,

Anak pohon kedua disebut **anak pohon kanan (right subtree)**.

# Binary Tree = Pohon Biner ( *Lanjutan* )

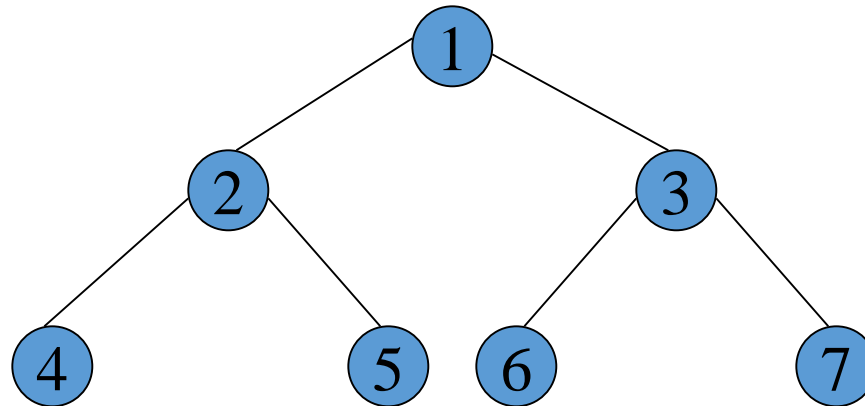
Contoh :  $((a + b * c) * d - e)$



# Binary Tree = Pohon Biner ( *Lanjutan* )

Pohon Biner Teratur Penuh (full regular binary tree) adalah suatu pohon biner beraturan di mana semua daunnya mempunyai panjang lintasan yang sama, yaitu tinggi pohon.

Contoh :



# Binary Tree = Pohon Biner ( *Lanjutan* )

- Teorema 1

Pohon biner beraturan dengan vertek lebih dari 2 mempunyai tepat 1 vertek berderajat 2 dan vertek lainnya berderajat 1 atau 3.

- Teorema 2

Banyaknya vertek dalam suatu pohon biner beraturan selalu ganjil.

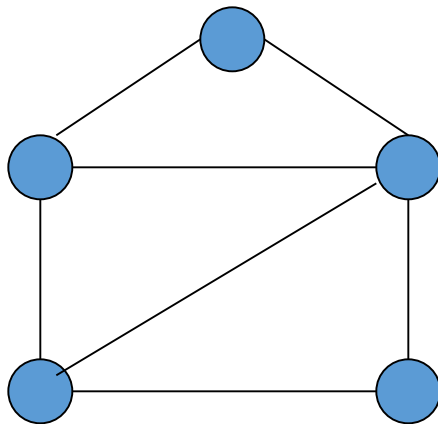




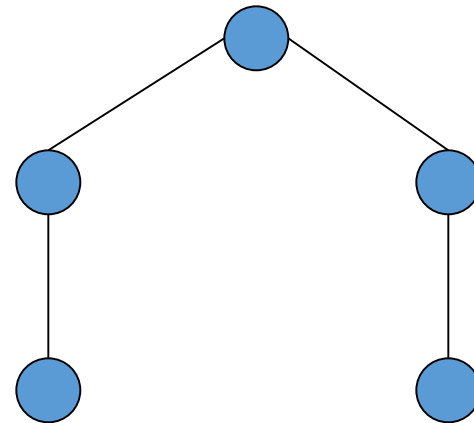
# Spanning Tree (Pohon Pembentang)

T disebut spanning tree dari graf terhubung  $G$ , bila  $T$  adalah suatu spanning subgraf dari graf  $G$  yang merupakan suatu tree.

Contoh :



Graf  $G$



Spanning Tree  $T$



# Minimum Spanning Tree

- Pembobot Spanning Tree adalah jumlah pembobot semua busur yang menyusun pohon tersebut.
- Minimum Spanning Tree adalah spanning tree dengan pembobot minimum.



# Minimum Spanning Tree ( *Lanjutan* )

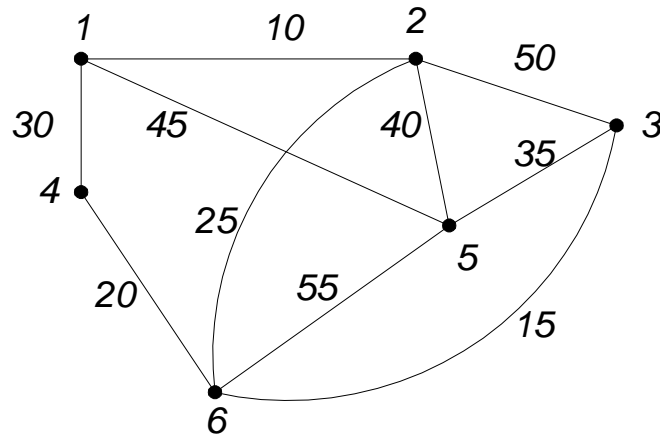
Cara menentukan minimum spanning tree

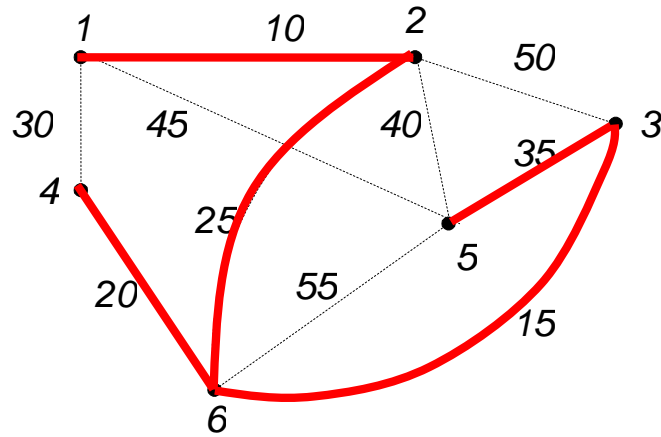
1. Pilih sembarang vertek dari graf  $G$  sebagai vertek awal pembentuk tree  $T$ ,
2. Pilih vertek lain yang belum ada di tree yang terhubung langsung dengan salah satu vertek yang menyusun tree yang mempunyai bobot minimal tetapi tidak membentuk loop,
3. Ulangi langkah 2 sampai semua vertek yang ada pada graf  $G$  ada pada tree  $T$ .
4. Tree yang terbentuk adalah minimum spanning tree.



## Contoh

Tentukan minimum spanning tree dari graf berikut dimulai dari vertex 1

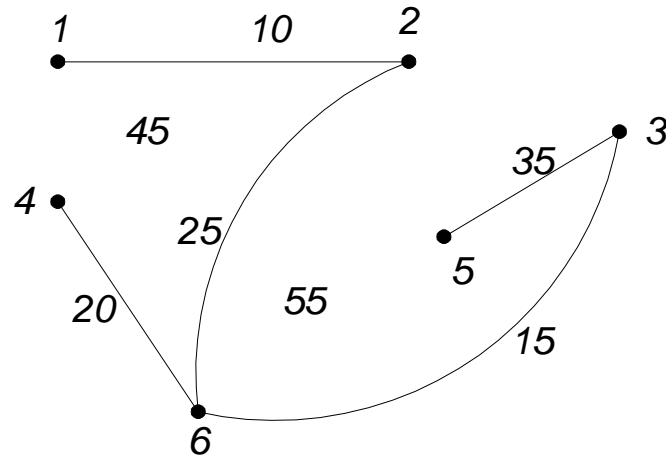




Sisi	Bobot
1. (1,2)	10
2. (2,6)	25
3. (6,3)	15
4. (6,4)	20
5. (3,5)	35



## Hasil minimum spanning tree



$$\text{Bobot} = 10 + 25 + 15 + 20 + 35 = 105$$



- Spanning tree yang dihasilkan tidak selalu unik meskipun bobotnya tetap sama.
- Hal ini terjadi jika ada beberapa sisi yang akan dipilih berbobot sama.

