Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung

Nama :…………………………
NIM :…………………………
T.tangan:…………………………

Solusi Kuis ke-4 IF2120 Matematika Diskrit (3 SKS) – Pohon dan Kompleksitas Aalgoritma
Dosen: Rinaldi Munir, Harlili
Senin, 3 Desember 2014
Waktu: 50 menit

1. Diketahui pohon ternary (n-ary dimana n = 3) mempunyai jumlah node 131 pada ketinggian k. Tentukan nilai k dan jumlah simpul daun.

Jawaban:

$3^0 + 3^1 + 3^2 + \ldots + 3^k = 3^0(3^{k+1} - 1) / (3-1)$

$= 131$

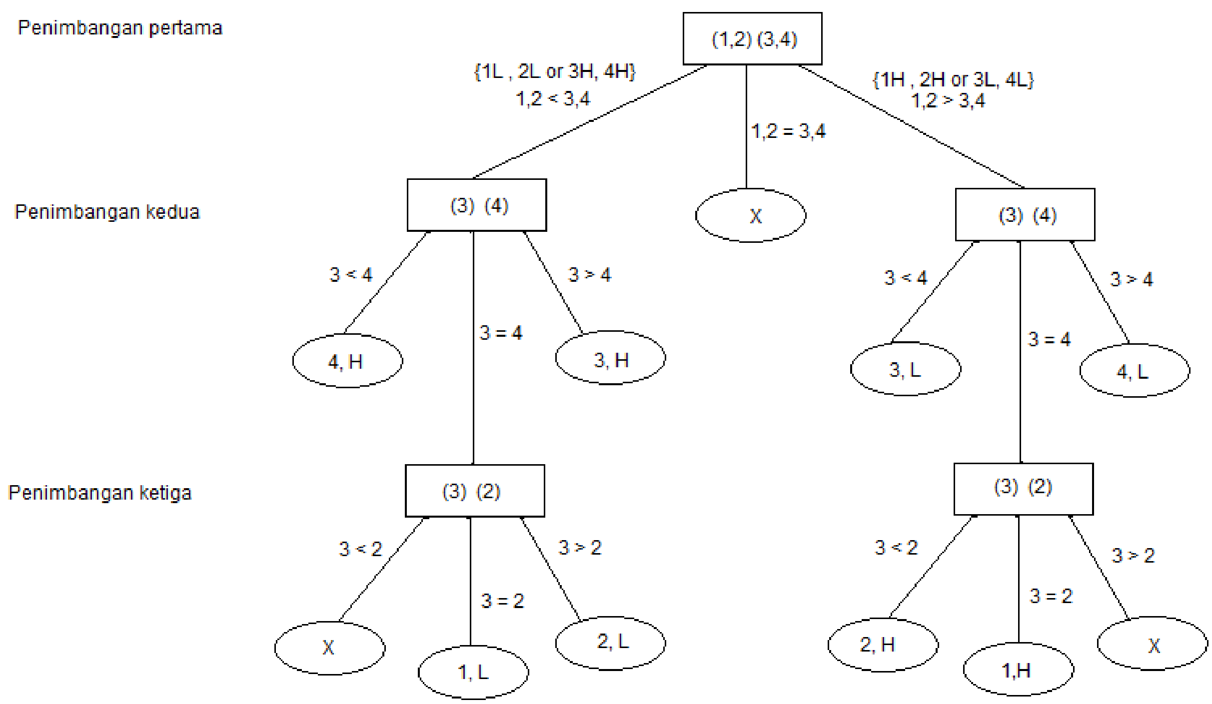
Dengan menyelesaikan persmaan diatas diperoleh k = 4.

Jumlah simpul daun yaitu $3^4 = 81$.

2. Diberikan 4 buah koin yang identik antara satu dengan yang lainnya, namun ternyata satu di antaranya adalah koin yang palsu. Koin yang palsu memiliki berat yang berbeda dengan koin yang asli, namun tidak diketahui apakah koin palsu tersebut lebih berat / lebih ringan daripada yang asli. Untuk menentukan mana yang palsu, diberikan sebuah timbangan, namun hanya dapat digunakan sebanyak 3 kali penimbangan. Dengan menggunakan decision tree, tentukan semua kemungkinan koin yang palsu berdasarkan penimbangan, dan apakah koin palsu tersebut lebih berat / lebih ringan dari yang asli.

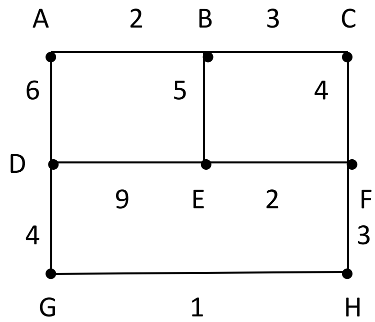
Jawaban:

Semua kemungkinan yang ada, dapat direpresentasikan menggunakan pohon keputusan di bawah ini :

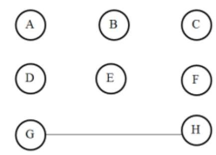
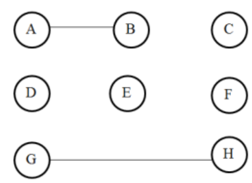
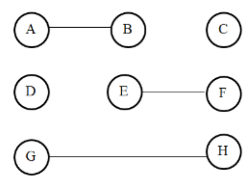
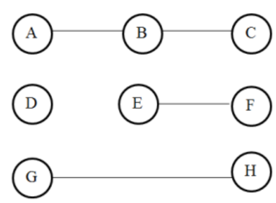
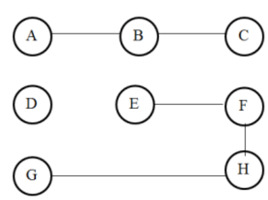
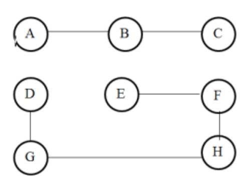


Dengan L artinya lighter (koin palsunya lebih ringan), dan H adalah heavier (koin palsunya lebih berat). X menyatakan kondisi yang tidak mungkin terjadi.

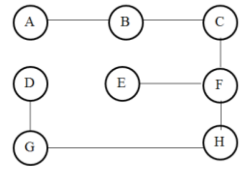
3. Buatlah pohon merentang minimum dari Graf dibawah ini dengan menggunakan algoritma Kruskal (sertakan & sketsakan tahapannya) dan tentukan nilai total cost dari pohon merentang minimum tersebut !



Jawaban:

Langkah	Sisi	Bobot	Pohon Rentang
1	(G,H)	1	
2	(A,B)	2	
3	(E,F)	2	
4	(B,C)	3	
5	(F,H)	3	
6	(D,G)	4	

7	(C,F)	4



Total cost = 19.

4. Nyatakan T(n) dalam notasi *O-besar* dan tentukan pula nilai *C, f(n), dan* n_o! (fungsi *f(n)* harus yang paling sederhana)
   a. $T(n) = 5n^3 + (\log n)^4$
   b. $T(n) = \frac{(n^4+3\log n)}{(n^4+2)}$
   c. $T(n) = n\log(n^2+1) + n^2\log n$
   "

   Jawab.
   a. T(n) = O(n^3)
   $5n^3 + (\log n)^4 \leq 6.\, n^3$ untuk setiap $n_0 \geq 1$ dan C=6, sehingga(*f(n)*= n^3, $n_0 \geq 1$, C=6)
   b. T(n) = O(1)
   $\frac{(n^4+3\log n)}{(n^4+2)} = \frac{(n^4+2)+(3\log n-2)}{(n^4+2)} = 1 + \frac{(3\log n-2)}{(n^4+2)} \leq 2.1$ untuk setiap $n_0 \geq 1$ dan C=2 sehingga
   (*f(n)*= 1, $n_0 \geq 1$, C=2)
   c. T(n) = O($n^2\log n$)
   $n\log(n^2+1) + n^2\log n \leq 2.\, n^2\log n$ untuk setiap $n_0 \geq 2$ dan C=2 sehingga (*f(n)*=$n^2\log n$, $n_0 \geq 2$, C=2)

5. Diberikan sebuah potongan algoritma sebagai berikut :

```
j←n
while j≥1 do
    for i←1 to j do
         x←x+1
    endfor
    j←j div 2
endwhile
```

Jika *T(n)* dihitung dari operasi penjumlahan pada pernyataan x⟵x+1, maka nyatakan *T(n)* dalam notasi *O*-besar, *Ω*-besar, dan *ϑ*-besar dan beri penjelasan!