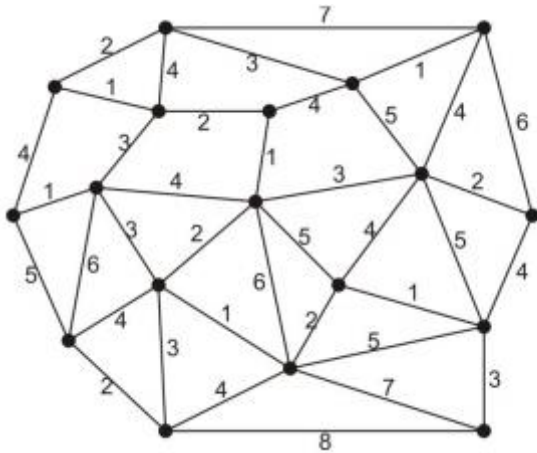


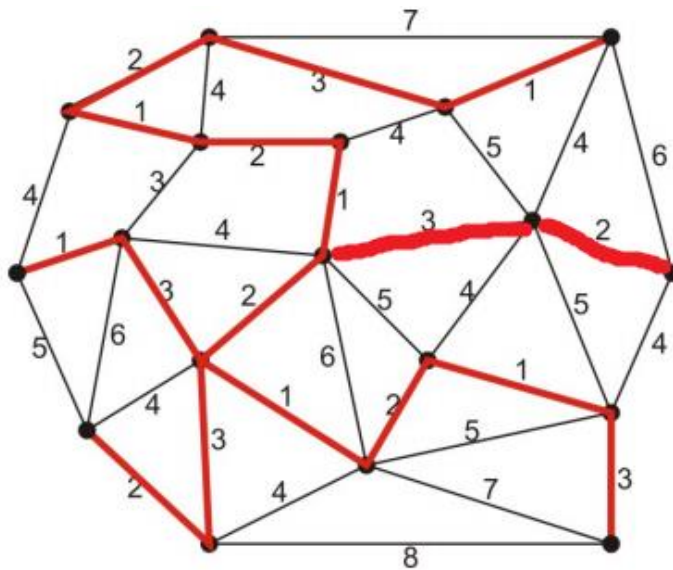
Solus Kuis ke-4 IF2120 Matematika Diskrit (3 SKS) – Rekursif dan Relasi Rekurens, Pohon, dan Kompleksitas Algoritma
 Dosen: Rinaldi Munir, Harlili
 Rabu, 2 Desember 2015
 Waktu: 75 menit

1. Graf di bawah ini merepresentasikan jarak antar 18 kota dalam suatu pulau. Tentukan total jarak minimum yang menghubungkan semua kota tersebut menggunakan pohon merentang minimum. Gunakan algoritma Prim! Gambarkan pohon merentang minimum yang dimaksudkan!



Jawaban:

Pohon merentang minimum yang didapat adalah sebagai berikut.



Total jarak minimum : $1 + 2 + 3 + 2 + 1 + 2 + 3 + 1 + 2 + 1 + 3 + 1 + 2 + 3 + 1 + 2 + 3 + 3 = 36$

2. Misalkan terdapat sebuah pohon P yang merupakan pohon 3-ary teratur dengan tinggi 4.
 a. Tentukanlah jumlah daun P dan jumlah simpul P !
 b. Misalkan dilakukan penghapusan semua anak (beserta keturunannya) dari salah satu simpul pada aras 2. Tentukanlah kembali jumlah daun P dan jumlah simpul P !

Jawaban:

a. Jumlah daun = $3^4 = 81$

Jumlah simpul = $3^0 + 3^1 + 3^2 + 3^3 + 3^4 = 121$

b. Jumlah daun = jumlah simpul dengan aras 4 ditambah satu simpul yang telah dihapus anaknya = $(3^4 - 3^2) + 1 = 73$

Jumlah simpul = $3^0 + 3^1 + 3^2 + (3^3 - 3) + (3^4 - 3^2) = 109$

3. Misalkan terdapat string: "RAJA PADJAJARAN"

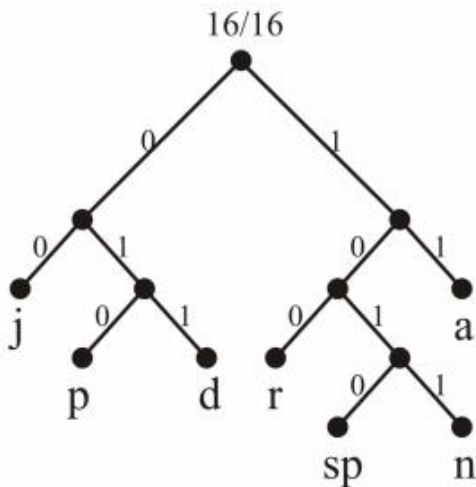
- Gambarkan pohon Huffman dengan terlebih dulu menghitung frekuensi kemunculan tiap karakternya (termasuk spasi)
- Tentukan kode huffman untuk masing-masing karakter dalam bentuk tabel lalu hitung panjang rangkaian bit yang dihasilkan jika string diatas diubah menjadi kode huffman yang telah dibuat
- Tentukan kata yang terbentuk dari rangkaian bit 10010001 dengan proses decoding menggunakan kode huffman diatas (jika tidak ada cukup tulis "tidak ada").

Jawaban:

a. Frekuensi kemunculan tiap karakter :

1/16	1/16	1/16	2/16	2/16	3/16	6/16
●	●	●	●	●	●	●
sp	n	p	d	r	j	a

Pohon huffman yang terbentuk



b. Kode Huffman

Karakter	Frekuensi	Peluang	Kode Huffman
spasi	1	1/16	1010
r	2	2/16	100
a	6	6/16	11
j	3	3/16	00
P	1	1/16	010
d	2	2/16	011
n	1	1/16	1011

c. Kata yang terbentuk : "rdj"

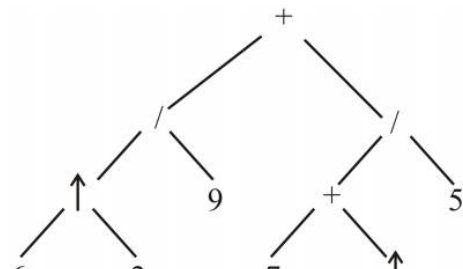
4. Misalkan terdapat operasi biner: penjumlahan (+), pengurangan (-), perkalian (*), pembagian (/), dan perpangkatan (\uparrow). Diketahui suatu ekspresi dalam bentuk postfix

6 2 \uparrow 9 / 7 2 3 \uparrow + 5 / +

- Tentukan pohon ekspresi dan nilai dari ekspresi postfix tersebut!
- Tentukan bentuk prefiks dan infiks dari ekspresi postfix tersebut!

Jawaban:

a. Pohon ekspresi yang terbentuk :



Nilai dari ekspresi tersebut : $6^2/9 + 7 + 2^3/5 = 4 + 3 = 7$

- Bentuk Prefiks : $+ / \uparrow 6 2 9 / + 7 \uparrow 2 3 5$
Bentuk Infiks : $6 \uparrow 2 / 9 + 7 + 2 \uparrow 3 / 5^3$

5. Tentukan notasi O , Ω dan Θ untuk $T(n) = 2n^2 + 7n \log n + 1!$

Jawaban:

- Karena $2n^2 + 7n \log n + 1 \leq 2n^2 + 7n^2 + n^2 = 10n^2$ untuk $n \geq 1$, maka dengan $C = 10$, diperoleh $2n^2 + 7n \log n + 1 = O(n^2)$
- Karena $2n^2 + 7n \log n + 1 \geq n^2$ untuk $n \geq 1$, maka dengan $C = 1$ diperoleh $2n^2 + 7n \log n + 1 = \Omega(n^2)$
- Karena $2n^2 + 7n \log n + 1 = O(n^2)$ dan $2n^2 + 7n \log n + 1 = \Omega(n^2)$, maka diperoleh $2n^2 + 7n \log n + 1 = \Theta(n^2)$

6. Tentukan solusi relasi rekurens berikut:

$$a_n = \frac{a_{n-2}}{4} \text{ untuk } n \geq 2, a_0 = 1, a_1 = 0$$

Jawaban:

Persamaan karakteristik: $r^2 - r - 6 = 0$.

Akar-akarnya: $(r - 3)(r + 2) = 0 \rightarrow r_1 = 3$ dan $r_2 = -2$

$a_n = \alpha_1 r_1^n + \alpha_2 r_2^n \rightarrow a_n = \alpha_1 3^n + \alpha_2 (-2)^n$

$a_0 = 3 \rightarrow a_0 = 3 = \alpha_1 3^0 + \alpha_2 (-2)^0 = \alpha_1 + \alpha_2$

$a_1 = 6 \rightarrow a_1 = 6 = \alpha_1 3^1 + \alpha_2 (-2)^1 = 3\alpha_1 - 2\alpha_2$

Diperoleh dua persamaan: $\alpha_1 + \alpha_2 = 3$ dan $3\alpha_1 - 2\alpha_2 = 6$,

solusinya adalah $\alpha_1 = 12/5 = 2.4$ dan $\alpha_2 = 3/5 = 0.6$

Jadi, solusi relasi rekurens adalah: $a_n = (2.4) \cdot 3^n + (0.6)(-2)^n$

7. Diketahui prosedur `foo` dalam notasi Pascal-like seperti dibawah:

```
function foo(n: integer) : integer
var i, j, k: integer

Algoritma
  i:=n
  while (i>0) do
```

```
j:=1
while(j<n) do
  k:=0
  while(k<n) do
    sum:=sum + (i + j + k)
    k:=k +2
  end while
  j:= j * 2,
end while
i:=i/2
end while
return sum
```

Tentukan kompleksitas algoritmanya jika ditinjau berdasarkan *assignment* nilai pada instruksi
sum:=sum + (i + j + k)!

Jawaban:

Di iterasi while paling luar, variabel i terus terbagi dua, sehingga akan terulang sekitar $\log_2 n$ kali.

Untuk setiap i, iterasi selanjutnya juga terulang sebanyak $\log_2 n$ kali

Pada iterasi while paling dalam, iterasi terulang sebanyak $\frac{n}{2}$ kali.

Karena ini adalah algoritma nested-loops, maka kalikan seluruh perkaliannya, didapat kompleksitas algoritmanya adalah $O((n(\log n))^2)$