

Solusi Kuis ke-4 IF1220 Matematika Diskrit (3 SKS) – Graf, Pohon, Kompleksitas Algoritma
Dosen: Rinaldi Munir, Rila Mandala, Arrival Dwi Sentosa
Selasa, 17 Desember 2024
Waktu: 90 menit

1. Paul menggambar 6 buah titik pada papan tulis. Kemudian ia menghubungkan beberapa titik dengan garis sedemikian sehingga tidak ada titik yang terhubung oleh 5 atau lebih garis. Paul memastikan tidak ada garis yang berpotongan. Paul juga ingin gambar yang ia buat mengakibatkan papan tulis terbagi menjadi sebanyak mungkin wilayah. Buktikan bahwa gambar yang dibentuk Paul termasuk graf teratur! (Nilai: 15)

Petunjuk: Soal ini dapat dijawab tanpa menggambar graf terlebih dahulu, cukup gunakan sifat graf.

Jawaban:

Perhatikan bahwa graf yang digambar Paul pasti merupakan graf bidang karena tidak ada garis yang berpotongan. Maka, berlaku rumus euler yaitu

$$n - e + f = 2 \rightarrow f = e - 4$$

Karena paul ingin graf membagi papan tulis menjadi sebanyak mungkin wilayah, maka paul perlu memastikan nilai f maksimum. Hal ini terjadi apabila nilai e juga maksimum. Ingat bahwa pada graf planar berlaku

$$e \leq 3n - 6 \rightarrow e \leq 12$$

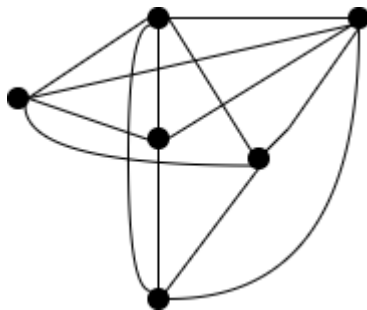
Sehingga diperoleh bahwa nilai maksimal dari e adalah 12. Hal ini juga akan mengakibatkan banyak wilayah pada papan tulis semaksimal mungkin. Dari lemma jabat tangan, akan diperoleh

$$\sum_{v \in V} d(v) = 2e \rightarrow \sum_{v \in V} d(v) = 24$$

Jumlah dari derajat seluruh simpul adalah 24. Pada soal disebutkan bahwa tidak ada simpul yang terhubung oleh 5 atau lebih garis. Akibatnya, derajat maksimal dari setiap simpul haruslah 4 saja. Artinya, setiap simpul pada graf pasti akan memiliki derajat yang sama yaitu 4, apabila tidak jumlah derajat sebanyak 24 tidak akan tercapai. Jadi, bisa disimpulkan bahwa setiap simpul memiliki derajat yang sama, atau dengan kata lain graf ini termasuk graf teratur.

2. Untuk mencapai target bisnis di tahun 2024, perusahaan *Infomin* menerima proyek pengembangan perangkat lunak yang dapat membuat desain jaringan transportasi kereta secara otomatis. Untuk memastikan kualitas perangkat lunak, Infomin meminta Anda untuk memeriksa jaringan transportasi yang didesain oleh perangkat lunak pada Gambar 1. Salah satu syarat yang harus dipenuhi desain jaringan adalah bahwa lokasi setiap stasiun dalam jaringan harus dapat diubah sedemikian rupa sehingga semua jalur kereta **tidak saling bersilangan**. Tentukan apakah desain sudah memenuhi syarat tersebut. Jika ya, gambarkan salah satu susunan stasiun yang memenuhi syarat. Jika tidak, jelaskan alasannya *dengan tegas* menggunakan teorema yang telah Anda pelajari.

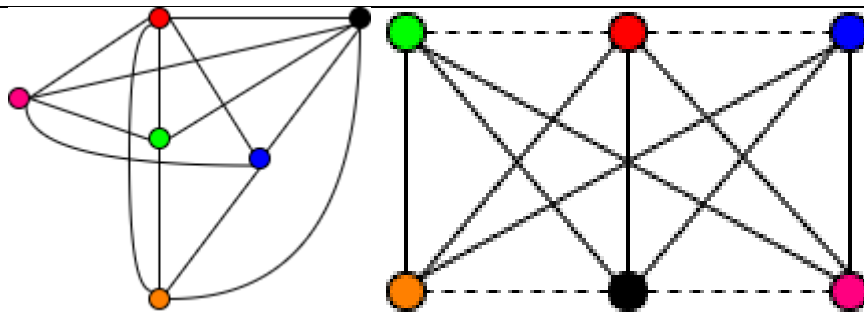
(Nilai: 15)



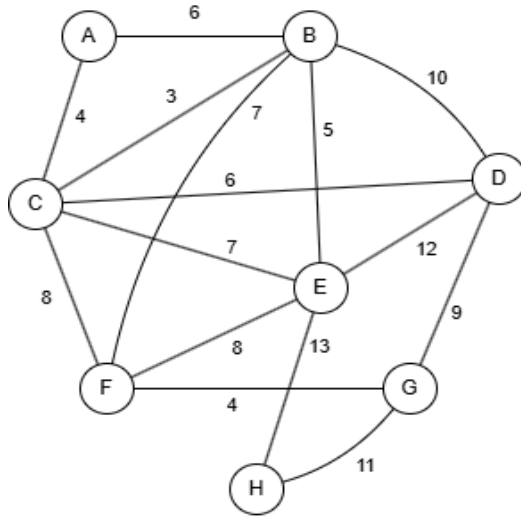
Gambar 1

Jawaban:

Syarat dapat dipenuhi apabila jaringan, yang dapat direpresentasikan dalam graf, adalah planar. Berdasarkan teorema Kuratowski, jaringan mengandung upagraf yang isomorfik dengan $K_{3,3}$ sehingga jaringan tidak planar dan tidak mungkin tidak bersilangan.



3. Tentukanlah pohon merentang minimum serta bobot total akhir dari graf pada Gambar 2 di samping kanan dengan menggunakan Algoritma Kruskal! Untuk setiap langkah, tuliskan sisi yang ditambahkan beserta bobot dari sisi tersebut (seperti pada salinda kuliah). Jika terdapat dua sisi dengan bobot yang sama, pilihlah sisi dengan huruf terurut berdasarkan alfabet (A, B, C ..). **(Nilai: 15)**



Gambar 2

Jawaban:

Sisi	(B,C)	(A,C)	(F,G)	(B,E)	(A,B)	(C,D)	(B,F)	(C,E)	(C,F)	(E,F)	(D,G)	(B,D)	(G,H)	(D,E)	(E,H)
Bobot	3	4	4	5	6	6	7	7	8	8	9	10	11	12	13

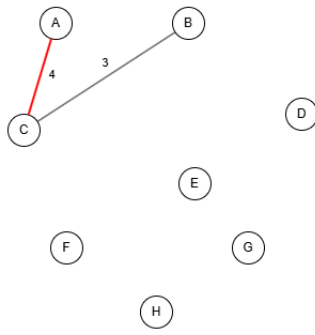
Algoritma Kruskal

Langkah	Sisi	Bobot	Pohon Rentang
1.	(B, C)	3	

2.

(A, C)

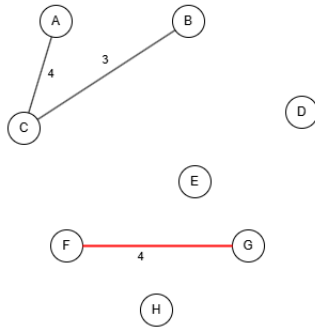
4



3.

(F, G)

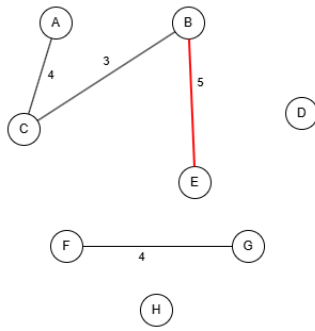
4



4.

(B, E)

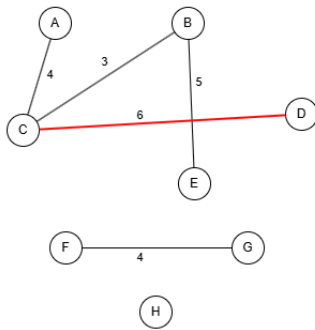
5



5.

(C, D)

6



6.	(B, F)	7	
7.	(G, H)	11	
Total		40	

4. Muthe ingin menggunakan Kode Huffman untuk mengirim pesan secara rahasia kepada temannya. Jika ia mengirim kata 'SEVENTEEN', tentukan kode Huffman yang dihasilkan dan berapa rasio pemampatannya? **(Nilai: 15)**

Jawaban:

Frekuensi

S = 1

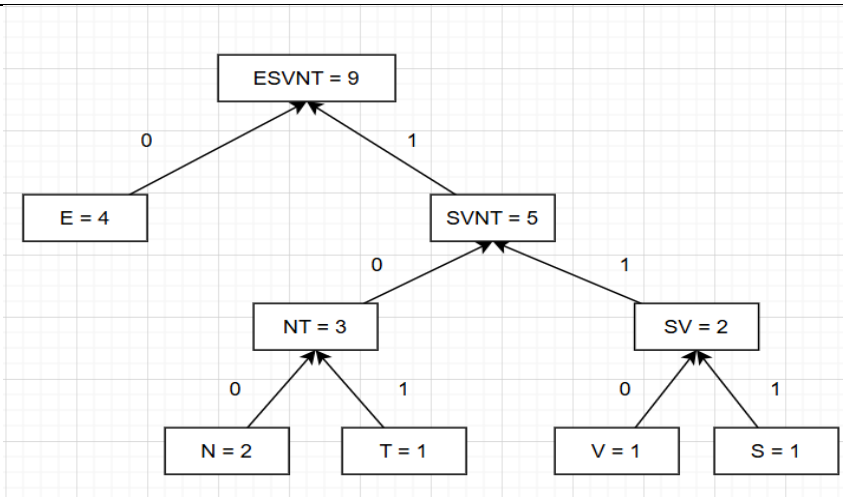
E = 4

V = 1

N = 2

T = 1

Pohon Huffman



Kode Huffman:

E = 0

N = 100

T = 101

V = 110

S = 111

1110110110010100100

Jumlah bit = 19 bit

Rasio pemampatan = $19 / 72 \times 100\% = 26,38\%$

5. Urutkan waktu eksekusi proses dibawah ini dari yang paling lambat menggunakan notasi O-besar!

- (a) $T(n) = n + 5\sqrt{n} + 3n\log(n)$ (b) $T(n) = n! + 12n^2$
 (c) $T(n) = \log(n^6) + 3n^2$ (d) $T(n) = n^2 \log(n) + n(\log(n))^2$
 (e) $T(n) = 1000n + 10000$

(Nilai: 10)

Jawaban:

Cari suku dominan tiap persamaan:

1. $3n(\log(n)) \rightarrow O(n\log(n))$
2. $n! \rightarrow O(n!)$
3. $3n^2 \rightarrow O(n^2)$
4. $n^2(\log(n)) \rightarrow O(n^2\log(n))$
5. $1000n \rightarrow O(n)$

Urutkan kompleksitas waktu tinggi ke rendah

$O(n!) > O(n^2\log(n)) > O(n^2) > O(n\log(n)) > O(n)$

Maka urutan waktu eksekusi proses dari yang paling lambat adalah

2 — 4 — 3 — 1 — 5

6. Tentukan kompleksitas waktu $T(n)$ dari algoritma di bawah ini berdasarkan jumlah operasi penjumlahan dan perkalian. Nyatakan $T(n)$ dalam notasi O-besar, dan tentukan pula nilai C dan $f(n)$! Tentukan juga notasi Omega-besar, dan Theta-besarnya! (Nilai: 10)

```

for i := 1 to n
  for j := 1 to 2i
    x := (c[j] + d[i]) * 3
  next j
next i
  
```

Jawaban:

Hitung Operasi Inner Loop

Dalam inner loop, j berjalan dari 1 hingga $2i$. Jadi, jumlah iterasi inner loop adalah $2i$. Operasi yang dilakukan dalam setiap iterasi adalah:

Penjumlahan $c[j]+d[i] \rightarrow 1$ operasi

Perkalian hasil dengan 3 $\rightarrow 1$ operasi.

Sehingga total operasi dalam inner loop adalah 2 operasi per iterasi.

Hitung Total Operasi Inner Loop

Karena inner loop berjalan $2i$ kali untuk setiap nilai i , maka jumlah total operasi dalam inner loop adalah: $2 \times 2i = 4i$ operasi.

Hitung Total Operasi Outer Loop

Outer loop berjalan dari $i=1$ hingga $i=n$. Untuk setiap iterasi i , inner loop memberikan $4i$ operasi. Total jumlah operasi untuk semua iterasi adalah:

$$\sum_{i=1}^n 4i = 4 \sum_{i=1}^n i$$

Selesaikan Jumlah Sigma

Kita tahu rumus sigma untuk $\sum_{i=1}^n i = \frac{n(n+1)}{2}$. Substitusikan ini ke dalam persamaan:

$$\sum_{i=1}^n 4i = 4 \cdot \frac{n(n+1)}{2} = 2n(n+1) = 2n^2 + 2n$$

Kompleksitas $T(n)$

Total jumlah operasi adalah $T(n) = 2n^2 + 2n$.

Dalam notasi O-besar, hanya suku dengan orde tertinggi yang diperhitungkan, sehingga:

$$T(n) \in O(n^2).$$

Nilai C dan $f(n)$

Dari hasil perhitungan, $f(n) = n^2$. Untuk menghitung C , bandingkan koefisien suku dominan dengan $f(n)$. Koefisien suku n^2 dalam $T(n)$ adalah 2, sehingga $C = 2$.

7. Gambarkanlah sebuah pohon yang memiliki sifat-sifat berikut. Jika pohon tidak dapat dibuat, jelaskan mengapa!

(a) Memiliki 10 simpul dan memiliki jumlah derajat semua simpul bernilai 24.

(b) Memiliki 12 simpul dan 15 sisi

(c) Memiliki 8 simpul dan 7 sisi

(d) Memiliki 4 simpul dan memiliki jumlah derajat semua simpul bernilai 3

(Nilai: 10)

Jawaban:

a. Memiliki 10 simpul dan memiliki jumlah derajat semua simpul bernilai 24.

Berdasarkan Teori **Handshaking Lemma**, suatu pohon pasti memiliki jumlah derajat semua simpul sama dengan dua kali lipat jumlah sisi pada pohon.

$$\sum_{v \in V} \deg v = 2|E|,$$

Untuk suatu pohon dengan jumlah simpul 10, maka terdapat $n-1 = 10-1 = 9$ sisi.

Oleh karena itu jumlah derajat semua simpul harus bernilai $2 * 9 = 18$ derajat.

Sehingga pohon **tidak dapat dibuat**.

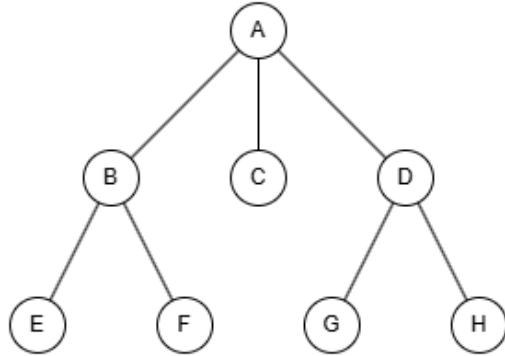
b. Memiliki 12 simpul dan 15 sisi

Salah satu sifat dari pohon adalah, suatu pohon dengan n simpul harus memiliki tepat $n-1$ jumlah sisi. Oleh karena itu, untuk pohon dengan 12 simpul, maka jumlah sisi harus berjumlah $12-1=11$ sisi.

Sehingga pohon **tidak dapat dibuat**.

- c. Memiliki 8 simpul dan 7 sisi

Berikut adalah salah satu contoh dari pohon yang dapat dibuat:



- d. Memiliki 4 simpul dan memiliki jumlah derajat semua simpul bernilai 3

Sama seperti bagian a, pohon dengan 4 simpul pasti memiliki $n-1 = 4-1 = 3$ sisi. Berdasarkan Teori Handshaking Lemma, jumlah derajat semua simpul harus bernilai $2*m = 2*3 = 6$ derajat.

Sehingga pohon **tidak dapat dibuat**.

8. Diberikan kode program di bawah ini:

```
function looploop(n){
  for (let i = 1; i <= n; i *= 2){
    for (let r = 1; r <= i; r++){
      for (let k = 1; k <= n; k++){
        console.log("Matdis seru!");
      }
    }
  }
}
```

Waktu proses $T(n)$ menyatakan jumlah kata "Matdis seru!" yang tercetak pada layar. Tentukan persamaan $T(n)$ dan nyatakan kompleksitasnya dalam notasi O -besar. **(Nilai: 10)**

Jawaban:

Struktur looping:

1. Loop pertama (i):

Nilai i akan berjalan dari $i = 1$ sampai $i \leq n$, dengan nilai i dikali 2 tiap iterasi ($i = 1, 2, 4, 8, \dots, n$). Jumlah iterasinya adalah $\log(n)$

2. Loop kedua (r):

Nilai r akan berjalan dari $r = 1$ sampai $r \leq i$, dengan nilai r ditambah 1 tiap iterasi ($r = 1, 2, 3, 4, \dots, i$) untuk setiap nilai i . Nilai i akan dikali 2 setiap iterasinya, jadi jumlah iterasi r secara rata-rata adalah $n/2$

3. Loop ketiga (k):

Nilai k akan berjalan dari $k = 1$ sampai $k \leq n$ (konstan), dengan nilai k ditambah tiap iterasi.
Jumlah iterasinya adalah n

$$T(n) = \log(n) * (n/2) * (n)$$

$$O(n) = n^2 \log(n)$$