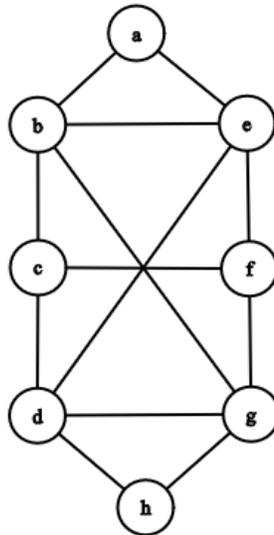


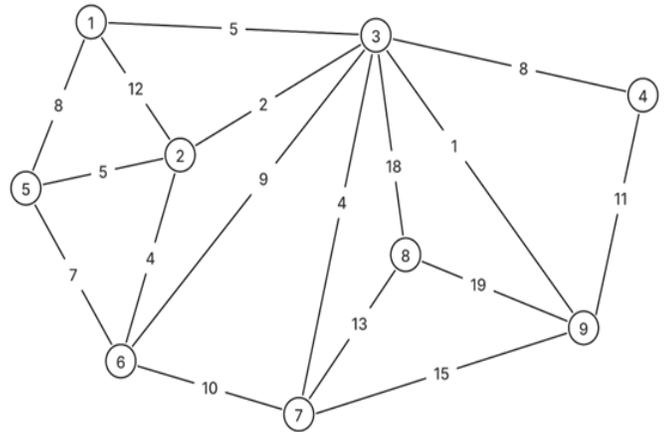
Kuis ke-4 IF1220 Matematika Diskrit (3 SKS) – Graf, Pohon, dan Kompleksitas Algoritma
 Dosen: Rinaldi, Arrival Dwi Sentosa
 Kamis, 5 Juni 2025
 Waktu: 90 menit

1. Diberikan gambar sebuah graf G seperti pada **Gambar 1** di samping kanan ini.
- Tunjukkan dengan ketidaksamaan Euler bahwa graf G tidak planar.
 - Tunjukkan dengan Teorema Kuratowski bahwa graf G tidak planar.
 - Tentukan bilangan kromatik dari graf G .

(Nilai: 15)



Gambar 1



Gambar 2

2. Diberikan graf tak berarah dengan simpul $\{A, B, C, D, E, F, G, H\}$ dan sisi $(A, B), (A, C), (A, D), (B, C), (B, E), (B, F), (C, D), (C, G), (D, E), (D, H), (E, F), (E, G), (F, H), (G, H)\}$. (Nilai: 15)
- Apakah graf ini Eulerian (mengandung sirkuit Euler)? Jika ya, berikan sirkuit Euler. Jika tidak, berikan alasan mengapa.
 - Apakah graf ini Hamiltonian? Jika ya, berikan sirkuit Hamilton. Jika tidak, berikan alasan mengapa.
 - Jika graf ini tidak Eulerian, tidak Hamiltonian, atau tidak keduanya, modifikasi graf dengan menambahkan sisi minimum agar menjadi Eulerian dan Hamiltonian.

3. Tentukanlah pohon merentang minimum serta bobot total akhir dari graf pada **Gambar 2** di atas dengan menggunakan Algoritma Kruskal! Buatlah tabel yang memperlihatkan proses pembentukan pohon merentang minimum step by step (Jika terdapat sisi berbobot sama, pilihlah sisi yang memiliki simpul terkecil) (Nilai: 15)

4. Terdapat sebuah pesan “ALJABAR BOOLEAN” dalam sebuah *script*. Berapakah panjang kode pesan tersebut jika didekoden dengan kode Huffman (termasuk spasi)? Tanpa petik dua “ tidak termasuk pesan. (Nilai: 15)

5. Diberikan masukan berupa rangkaian karakter dengan urutan sebagai berikut: T, E, K, N, I, F, O, R, M, A, B, J, G
- Gambarkan pohon pencarian (*binary search tree*) yang terbentuk.
 - Tentukan hasil penelusuran *preorder*, *inorder*, dan *postorder*, dari pohon jawaban (a) di atas. (Nilai: 15)

6. Diberikan dua algoritma berikut:

```
def p1(n):
    total = 1
    for i in range(1, n + 1):
        term = 1
        for _ in range(1, i + 1):
            term *= n
        total += term
    return total
```

```
def p2(n):
    result = 1
    for _ in range(n, 0, -1):
        result = 1 + result * n
    return result
```

Untuk sebarang nilai $n \geq 0$, kedua fungsi tersebut akan menghasilkan nilai yang sama. Dari kedua algoritma tersebut, tentukan: (Nilai: 10)

- Jumlah operasi perkalian dan penjumlahan yang dilakukan kedua algoritma tersebut sebagai fungsi dari n !
- Nyatakan kompleksitas waktu algoritma tersebut dalam notasi Big-O, Big- Ω , dan Big- Θ !

7. Diketahui sebuah program Python dengan 4 buah fungsi sebagai berikut:

<pre>from typing import List def get_factor_sums(num: int) -> int: factor_num = 0 for i in range(1, math.sqrt(num)): if num % i == 0: factor_num += i if num // i != i: factor_num += (num // i) return factor_num</pre>	<pre>def triangular_sum(m: int, n: int) -> int: sume = 0 for i in range(m): for j in range(n): sume += (i * m + j) + 1 return sume</pre>
<pre>def right_shift_amount(num: int) -> int: amount = 0 while (num > 0): num = num // 2 amount += 1 return amount</pre>	<pre>def sum_sum_tulang(num: int) -> int: sume = 0 for i in range(num): block = int(i ** (1/3)) for j in range(block): sume += j return sume</pre>
<p>a. Tentukan notasi Big-O dan urutkan fungsi diatas dari yang paling mahal (paling lambat) sampai yang paling murah!</p> <p>b. Berikan penjelasan mengapa <code>get_factor_sum</code> memiliki <code>for</code> dari 1 sampai <code>sqrt(num)</code> dan bukan <code>num</code> saja (Nilai: 15)</p>	
<p>8. (Bonus 5) Diberikan bilangan asli 2 sampai 10. Buktikan bahwa tidak mungkin membagi himpunan bilangan ini menjadi 2 himpunan sehingga tidak ada bilangan yang dapat membagi bilangan lain di dalam satu himpunan yang sama. Gunakan teori graf untuk membuktikan pernyataan tersebut</p>	
<p>9. (Bonus 5) Diberikan beberapa waktu proses $T(n)$ dari 5 algoritma. Nyatakan ekspresi tersebut dalam notasi Big-O dan urutkan notasi Big-O tersebut dari yang tercepat!</p> <ol style="list-style-type: none"> 1) $T(n) = 10n \log n + n^2$ 2) $T(n) = 2^n + n^5$ 3) $T(n) = \log^2 n + n$ 4) $T(n) = n!$ 5) $T(n) = n \log n + n^3$ 	

Total nilai: 100 + bonus 10 = 110

=====

Kerjakan mulai dari halaman dibalik ini, lalu pada kertas tambahan. Jika masih kurang, silakan pakai kertas sendiri