

Draft Jawaban UTS Matematika Diskrit

Nama: [Nama Anda]

NIM: [NIM Anda]

Mata Kuliah: Matematika Diskrit

1. Analisis Himpunan: Evolusi Ekosistem Microsoft Windows

Konteks: Soal ini bukan sekadar operasi himpunan, melainkan sebuah "Audit Kompatibilitas Sistem". Kita memetakan evolusi OS Windows dari masa lalu ke masa kini.

Diketahui (Data Aset):

- Semesta (U): Seluruh Sistem Produksi Microsoft.
- A (Era Klasik/Milenium): $\{\text{win97, win98, win98SE, winME}\}$
- B (Era Transisi/Modern Awal): $\{\text{winME, win2000, winXP, win7}\}$
- C (Era Modern/Touch): $\{\text{win7, win8, win10, win11}\}$

Jawaban & Analisis:

a. $(A \cup B) - B$

- Logika:** Menggabungkan semua OS di Era A dan B, lalu menghapus OS yang masih ada di Era B. Ini sama dengan mencari OS yang **benar-benar usang** dan sudah ditinggalkan total oleh Era B.
- Perhitungan:**
 - $A \cup B = \{\text{win97, win98, win98SE, winME, win2000, winXP, win7}\}$
 - Dikurangi $B: \{\text{winME, win2000, winXP, win7}\}$
- Hasil:** $\{\text{win97, win98, win98SE}\}$
- Interpretasi:** Ini adalah daftar "Museum Ware" yang murni dari era 90-an.

b. $(A \cap B) \cup C'$

- Logika:** Mencari jembatan antara Era A dan B (irisan), digabung dengan segala sesuatu yang **BUKAN** Era Modern C (non-modern).
- Perhitungan:**
 - $A \cap B = \{\text{winME}\}$ (Satu-satunya OS "jembatan" yang penuh bug).
 - C' (Komplemen C di Semesta soal): Semua yang bukan win7, 8, 10, 11.
 $C' = \{\text{win97, win98, win98SE, winME, win2000, winXP}\}$
 - Gabungan $\{\text{winME}\}$ dengan C' adalah C' itu sendiri.
- Hasil:** $\{\text{win97, win98, win98SE, winME, win2000, winXP}\}$

c. $(A \oplus B) - C$

- Logika:** Beda Setangkup ($A \oplus B$) adalah OS yang hanya eksklusif di A atau eksklusif di B (tidak boleh ada di keduanya). Lalu buang OS yang masih dipakai di Era C.
- Perhitungan:**

- $A \oplus B = (A \cup B) - (A \cap B) = \{\text{win97, win98, win98SE, win2000, winXP, win7}\}$ (Membuang winME).
- Dikurangi C ($\{\text{win7, win8, win10, win11}\}$): Hapus win7.
- Hasil: $\{\text{win97, win98, win98SE, win2000, winXP}\}$
- Interpretasi: Daftar OS unik non-irisan yang benar-benar sudah tidak didukung oleh standar modern.

d. $(B - C) \oplus A$

- Logika: Ambil OS di B yang sudah tidak ada di C, lalu cari eksklusivitasnya terhadap A.
- Perhitungan:
 - $B - C = \{\text{winME, win2000, winXP}\}$ (Win7 dihapus karena ada di C).
 - $(B - C) \oplus A$: Bandingkan $\{\text{winME, win2000, winXP}\}$ dengan $\{\text{win97, win98, win98SE, winME}\}$.
 - Irisannya adalah $\{\text{winME}\}$.
 - Elemen unik: $\{\text{win2000, winXP}\}$ dari kiri dan $\{\text{win97, win98, win98SE}\}$ dari kanan.
- Hasil: $\{\text{win97, win98, win98SE, win2000, winXP}\}$

e. $(A \cap B) \cup (A \cap C)'$

- Logika: Irisan A dan B, digabung dengan seluruh semesta kecuali irisan A dan C.
- Perhitungan:
 - $A \cap B = \{\text{winME}\}$
 - $A \cap C = \emptyset$ (Himpunan kosong, tidak ada OS A yang langsung lompat ke C).
 - $(A \cap C)' = \emptyset' = U$ (Semesta).
 - $\{\text{winME}\} \cup U = U$.
- Hasil: **Semesta Pembicaraan (Seluruh OS yang terdaftar)**
- Interpretasi: Secara logika, ini adalah tautologi himpunan dalam konteks ini.

f. $(A - B) \cap C'$

- Logika: OS murni A (bukan B), yang juga bukan C.
- Perhitungan:
 - $A - B = \{\text{win97, win98, win98SE}\}$
 - $C' = \text{Semua kecuali C}$.
 - Irisan keduanya tetap $A - B$ karena $A - B$ adalah subset dari C' .
- Hasil: $\{\text{win97, win98, win98SE}\}$

2. Forensik Fungsi: Relasi Antara ID dan Server

Analogi Kasus: Bayangkan A adalah ID Pengguna $\{1, 2, 3\}$ dan B adalah Server $\{u, v, w, x\}$. Relasi $f = \{(1, w), (2, u), (3, v)\}$ adalah aturan "Load Balancing".

Analisis:

1. Apakah Fungsi?

- **YA.**
- *Alasan:* Setiap User (domain A) terhubung ke tepat satu Server. Tidak ada user yang *request*-nya mendua atau tidak terlayani.
 - $1 \rightarrow w$
 - $2 \rightarrow u$
 - $3 \rightarrow v$

2. Apakah Satu-ke-Satu (Injektif)?

- **YA.**
- *Alasan:* Tidak ada dua user yang berebut server yang sama. Setiap user mendapat server unik (w, u, v masing-masing berbeda). Tidak terjadi *collision*.

3. Apakah Pada (Surjektif)?

- **TIDAK.**
- *Alasan:* Lihat himpunan Kodomain $B = \{u, v, w, x\}$. Ada server x yang *idle* (menganggur) karena tidak menerima koneksi dari user manapun. Agar surjektif, semua server harus terpakai.

Kesimpulan: f adalah fungsi injektif (satu-ke-satu) tetapi bukan surjektif (bukan pada).

3. Pembuktian Induksi Matematika: "The Domino Effect"

Proposisi: $\sum_{i=1}^n i(i+1) = \frac{n(n+1)(n+2)}{3}$

Langkah 1: Basis Induksi (Trigger Awal) Untuk $n = 1$:

- Ruas Kiri (LHS): $1 \cdot (1 + 1) = 1 \cdot 2 = 2$
- Ruas Kanan (RHS): $\frac{1(1+1)(1+2)}{3} = \frac{1 \cdot 2 \cdot 3}{3} = \frac{6}{3} = 2$
- **Terbukti Benar** untuk $n = 1$.

Langkah 2: Hipotesis Induksi (Asumsi Pola) Asumsikan rumus benar untuk $n = k$:

$$1 \cdot 2 + 2 \cdot 3 + \dots + k(k+1) = \frac{k(k+1)(k+2)}{3}$$

Langkah 3: Langkah Induksi (Membuktikan Domino Selanjutnya) Kita harus membuktikan untuk $n = k + 1$. Target yang dituju: $\frac{(k+1)((k+1)+1)((k+1)+2)}{3} = \frac{(k+1)(k+2)(k+3)}{3}$

Bukti: Tambahkan suku ke- $(k + 1)$ pada kedua ruas hipotesis:

$$\begin{aligned} & [1 \cdot 2 + \dots + k(k+1)] + (k+1)(k+2) \\ &= \frac{k(k+1)(k+2)}{3} + (k+1)(k+2) \end{aligned}$$

Faktorkan $(k+1)(k+2)$ keluar (teknik *common factor*):

$$\begin{aligned}
 &= (k+1)(k+2) \left[\frac{k}{3} + 1 \right] \\
 &= (k+1)(k+2) \left[\frac{k+3}{3} \right] \\
 &= \frac{(k+1)(k+2)(k+3)}{3}
 \end{aligned}$$

Kesimpulan: Karena Basis benar dan Langkah Induksi konsisten mencapai target, maka rumus ini **VALID** untuk seluruh bilangan bulat positif n . Quod Erat Demonstrandum.

4. Logika Proposisi: Paradoks Mahasiswa Informatika

Konteks Persona: Mari kita asumsikan saya adalah "Budi", mahasiswa Informatika Usia semester 3 yang rajin (sedang mengerjakan ujian ini).

- p : Saya adalah mahasiswa Informatika Usia semester 3.
 - **Nilai:** Benar (T) — Karena saya sedang ujian ini.
- q : Saya tidak mengambil mata kuliah matematika diskrit semester ini.
 - **Nilai:** Salah (F) — Mustahil saya ujian MatDis kalau tidak ambil mata kuliahnya.

Evaluasi Nilai Kebenaran:

a. $\sim p$ (Negasi p)

- "Saya BUKAN mahasiswa Informatika Usia semester 3."
- $\sim T = \text{Salah}(\text{False})$

b. $p \vee q$ (Disjungsi)

- "Saya mahasiswa... ATAU saya tidak mengambil matkul..."
- $T \vee F = \text{Benar}(\text{True})$ (Asal salah satu benar, maka kalimat aman).

c. $p \wedge q$ (Konjungsi)

- "Saya mahasiswa... DAN saya tidak mengambil matkul..."
- $T \wedge F = \text{Salah}(\text{False})$ (Kontradiksi fakta).

d. $p \rightarrow q$ (Implikasi)

- "JIKI saya mahasiswa..., MAKA saya tidak mengambil matkul..."
- $T \rightarrow F = \text{Salah}(\text{False})$ (Janji diingkari: Syarat terpenuhi, tapi akibatnya salah).

e. $p \leftrightarrow q$ (Bi-implikasi)

- "Saya mahasiswa... JIKA DAN HANYA JIKA saya tidak mengambil matkul..."
- $T \leftrightarrow F = \text{Salah}(\text{False})$ (Kedua sisi tidak setara).

5. Rangkaian Logika Digital: Efisiensi Hardware

Fungsi: $f(x, y) = x'y + xy' + y'$

Analisis "Out of the Box" (Penyederhanaan): Sebelum merangkai kabel (gerbang logika), seorang *engineer* cerdas akan menghemat budget gerbang dengan menyederhanakan fungsi:

1. Lihat $x'y + y'$.
2. Hukum Absorpsi/Distributif: $y'(x + 1) + x'y = y' + x'y$.
3. Hukum De Morgan/Redundansi: $y' + x'y = (y' + x')(y' + y) = x' + y'$.
4. Jadi, fungsi aslinya setara dengan $x' + y'$ (Gerbang NAND logikanya mirip, atau OR dengan input inverted).

Namun, sesuai permintaan soal untuk menyatakan fungsi aslinya, berikut adalah desain rangkaiannya:

Rangkaian Logika (Original):

- **Input:** X, Y
- **Gerbang NOT:** Untuk membuat x' dan y' .
- **Gerbang AND 1:** Menerima x' dan $y \rightarrow$ Output: $x'y$
- **Gerbang AND 2:** Menerima x dan $y' \rightarrow$ Output: xy'
- **Gerbang OR (3-input):** Menggabungkan $(x'y)$, (xy') , dan $(y') \rightarrow$ Output Akhir f .

(Dalam lembar jawaban ujian, gambarlah simbol gerbang standar: Segitiga+bulatan untuk NOT, Setengah lingkaran untuk AND, Bulan sabit untuk OR).

6. Deret Fibonacci: Kode Rahasia Alam Semesta

Definisi Formal: Deret Fibonacci adalah barisan rekursif di mana setiap angka adalah penjumlahan dua angka sebelumnya: $F_n = F_{n-1} + F_{n-2}$, dengan seed awal 0, 1. Barisan: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Pemahaman Mendalam & Aplikasi Aktual (Out of the Box):

Bagi saya, Fibonacci bukan sekadar deret angka, melainkan "**Sidik Jari Tuhan**" atau algoritma efisiensi alam semesta.

1. **Optimasi Alam (Biomimikri):** Alam menggunakan Fibonacci untuk efisiensi *packing*. Bunga matahari menyusun bijinya dalam spiral Fibonacci agar bisa memuat biji terbanyak dalam ruang tersempit tanpa saling menindih. Ini adalah solusi masalah *optimization* dalam matematika.
2. **Analisis Teknikal Saham (Fibonacci Retracement):** Dalam dunia *trading* (Forex/Kripto), trader menggunakan rasio Fibonacci (Golden Ratio 1.618 atau level 0.618) untuk memprediksi kapan harga saham akan "memantul". Psikologi massa pasar entah kenapa mematuhi rasio matematika ini secara tidak sadar.
3. **Efisiensi Komputasi (Rekursif vs Dinamis):** Di ilmu komputer, Fibonacci adalah studi kasus klasik untuk membedakan algoritma yang buruk dan baik.
 - *Naive Recursive:* $O(2^n)$ — Sangat lambat untuk n besar (kompleksitas eksponensial).