LEMBAR KERJA SISWA — MODUL 3

Algoritma Manual (*Greedy*, DP, Graf, *Bitwise*) & *Deep Code Tracing*

Bimbingan OSN Informatika — Universitas Bengkulu | Tanpa Kalkulator & Tanpa Kompilator

Petunjuk Pengerjaan & Identitas Siswa

Nama: _____ **Sekolah:** _____ **Waktu:** 75 Menit

- Lembar kerja ini melatih dua tulang punggung OSN-K: **eksekusi algoritma manual** dan **deduksi pola fungsi rekursif** tanpa kompilator (Taksonomi Bloom **C1–C6**).
- Untuk kode berinput besar seperti `f(2026)`, buatlah tabel $n = 1, 2, \ldots, 8$ terlebih dahulu untuk menemukan polanya!

Bagian 1: Fondasi & Pemahaman Konsep (C1 – C2)

Soal 1. C1 – Mengingat **Identifikasi Sifat *Bitwise* & Graf**

Lengkapi pernyataan singkat berikut dengan tepat:

(a) Jika `x > 0` dan `(x & (x - 1)) == 0`, maka `x` pasti berupa bilangan _____

(b) Nilai dari ekspresi `a ^ b ^ a` selalu sama dengan _____

(c) Graf terhubung memiliki **Lintasan Euler** (bukan siklus) jika tepat _____ simpul berderajat ganjil.

(d) Sebuah *Minimum Spanning Tree* pada graf terhubung dengan V simpul selalu memiliki tepat _____ sisi.

Soal 2. C2 – Memahami **Jebakan *Greedy* pada Penjadwalan Aktivitas**

Diberikan 3 aktivitas dengan rentang [mulai, selesai]: $A = [1, 5]$, $B = [4, 6]$, dan $C = [5, 9]$.

(a) Jika siswa memakai strategi *greedy* **"pilih aktivitas berdurasi paling pendek lebih dulu"**, aktivitas mana yang terpilih dan berapa total aktivitas yang didapat?

(b) Tunjukkan urutan optimal dengan strategi **"pilih waktu selesai paling awal"** dan jelaskan mengapa strategi durasi terpendek gagal!

Bagian 2: Penerapan Algoritma Manual OSN-K (C3)

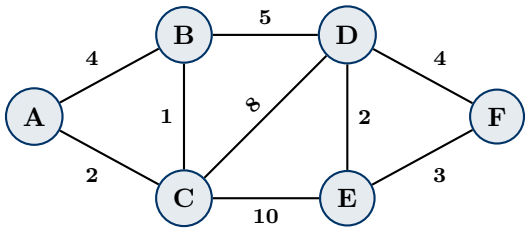
Soal 3. C3 – Menerapkan Isian Singkat **Evaluasi Ekspresi *Bitwise* Manual**

Tentukan nilai akhir variabel `y` dari potongan kode C++ berikut tanpa bantuan komputer:

```
int x = (25 & 14) | (12 ^ 7);
int y = (x << 1) ^ (x >> 1);
```

Jawaban Akhir:

Soal 4. C3 – Menerapkan | Isian Singkat **MST vs. Lintasan Terpendek Graf Berbobot**
Perhatikan graf tak berarah berbobot 6 simpul di bawah ini:



(a) Tentukan **total bobot Minimum Spanning Tree (MST)** graf tersebut!

(b) Tentukan **jarak lintasan terpendek** dari simpul A ke simpul F!

Jawaban Akhir:

Soal 5. C3 – Menerapkan | Isian Singkat **0/1 Knapsack Manual vs. *Greedy* Rasio**
Kapasitas tas maksimal $W = 7$. Terdapat 4 barang yang masing-masing hanya boleh diambil paling banyak 1 kali: $B_1 : (w = 5, v = 9)$, $B_2 : (w = 4, v = 7)$, $B_3 : (w = 3, v = 5)$, $B_4 : (w = 2, v = 2)$.
Tentukan **nilai maksimum** yang bisa dibawa menggunakan Tabel DP (lalu cek berapa yang didapat jika memakai *greedy* rasio v/w)!

Jawaban Akhir:

Soal 6. C3 – Menerapkan | Isian Singkat **Tracing Loop Bersarang (Makna Kelipatan)**
Berapakah nilai akhir variabel `ans` setelah kode berikut selesai dieksekusi?

```
int ans = 0;
for (int i = 1; i <= 20; i++) {
    for (int j = i; j <= 20; j += i) {
        ans++;
    }
}
```

Jawaban Akhir:

Bagian 3: Analisis Mendalam — Deep Code Tracing OSN-K (C4)

Soal 7. C4 – Menganalisis | Isian Singkat **Deduksi Pola Rekursi Input Besar (`f(2026)`)**

Perhatikan fungsi rekursif berikut ini:

```
int f(int n) {
    if (n <= 1) return 1;
    if (n % 2 == 0) return 2 * f(n / 2) - 1;
    return 2 * f(n / 2) + 1;
}
```

Hitung tabel $f(1)$ sampai $f(9)$, temukan hubungannya dengan pangkat 2, lalu tentukan nilai eksak dari `f(2026)`!

Jawaban Akhir:

Soal 8. C4 – Menganalisis | Isian Singkat **Rekursi 2 Parameter (Formula Tersembunyi)**

Perhatikan fungsi rekursif berikut:

```
int h(int n, int k) {
    if (n == 0) return 0;
    return (n / k) + h(n / k, k);
}
```

(a) Tentukan nilai eksak kembalian dari `h(2026, 5)`!

(b) Jelaskan secara singkat apa makna matematis dari `h(n, 5)` terkait bilangan $n!$ (faktorial)!

Jawaban Akhir:

Soal 9. C4 – Menganalisis | Isian Singkat **Kombinasi XOR Array & Loop *Bitwise***

Tentukan nilai akhir variabel `res` dan `cnt` setelah kedua potongan kode berikut dijalankan berurutan:

```
int a[] = {3, 5, 2, 5, 7, 3, 2, 9, 7};
int res = 0;
for (int i = 0; i < 9; i++) res = res ^ a[i];

int cnt = 0, x = res * 10;
while (x > 0) {
    x = x & (x - 1);
    cnt++;
}
```

Jawaban Akhir:

Bagian 4: Evaluasi & Kreasi Strategi (C5 – C6)

Soal 10. C5 – Mengevaluasi **Audit Strategi *Greedy* Lokal pada Lintasan Terpendek**

Seorang siswa mengusulkan algoritma mencari lintasan terpendek dari S ke T pada graf berbobot positif: *"Dari simpul saat ini, selalu ambil sisi keluar dengan bobot paling kecil yang menuju simpul belum dikunjungi sampai tiba di T."*

(a) Evaluasi algoritma tersebut! Gambarlah **graf tandingan (*counterexample*)** berisi 3–4 simpul di mana algoritma siswa gagal, lalu tunjukkan mengapa **Dijkstra** berhasil pada graf yang sama!

(b) Mengapa pemilihan sisi terkecil berhasil pada MST (Kruskal), tetapi tidak selalu tepat untuk lintasan terpendek?

Soal 11. C5 – Mengevaluasi **Jumlah Pemanggilan Rekursif vs. *Memoization***

Perhatikan fungsi rekursif berikut dengan variabel global `calls = 0`:

```
int calls = 0;
int T(int n) {
    calls++;
    if (n <= 1) return 1;
    return T(n - 1) + T(n - 2);
}
```

(a) Jika dipanggil `T(6)` tanpa *memoization*, hitung nilai `T(6)` **dan** nilai akhir `calls`!

(b) Jika digunakan tabel *memoization*, berapa banyak *state* unik yang dihitung untuk `T(6)` dan `T(100)`? Jelaskan keuntungan efisiensinya!

Jawaban Akhir:

Soal 12. C6 – Menciptakan **Merancang DP Manual & Soal *Deep Tracing* Orisinal**

(a) **Desain DP (*Non-Adjacent Max Sum*):** Diberikan 7 nilai rumah berurutan: $6, 7, 1, 30, 8, 2, 4$. Pilih beberapa rumah agar total nilainya **maksimal** dengan syarat **tidak boleh memilih dua rumah bersebelahan**. Tuliskan relasi transisi $dp[i]$, isi tabel $dp[1..7]$, dan tentukan hasil maksimalnya!

(b) **Kreasi Soal Tracing:** Rancanglah **1 fungsi rekursif singkat (4–6 baris)** bertipe OSN-K yang menyembunyikan suatu operasi matematika (mis. jumlah digit, membalik bilangan, atau pangkat), lalu buat pertanyaan `f(2026)` beserta pembahasannya!

"Di bagian Pemahaman Program OSN-K, komputer terbaik bukan yang paling cepat menghitung baris demi baris, melainkan yang paling cepat melihat POLA di balik kode."