



**UJIAN TENGAH SEMESTER (UTS)**  
**DEPARTEMEN INFORMATIKA FAKULTAS SAINS DAN MATEMATIKA**  
**UNIVERSITAS DIPONEGORO SEMARANG**  
**SEMESTER GENAP TAHUN 2023/2024**

Mata Kuliah	: PAIK6102-Algoritma dan Pemrograman (4 sks)
Kelas	: A, B, C, D, E dan F
Pengampu	: Dr. Aris Puji Widodo, MT. Drs. Eko Adi Sarwoko M.Komp.
Departemen/Program Studi	: Informatika/S1 Informatika
Hari/Tanggal	: Senin/01 April 2024
Jam/Ruang	: 10.00 – 11.30 (90 menit)
Sifat Ujian	: <b>Open Books</b> ( <i>Tidak diperbolehkan membuka Handphone/PC/Laptop</i> )

<b>Capaian Pembelajaran Lulusan (CPL)</b>	<p><b>CPL-P05:</b> Mampu menerapkan konsep teoretis bidang ilmu komputer dalam mengidentifikasi solusi permasalahan kompleks dengan prinsip komputasi dan ilmu lain yang relevan.</p> <p><b>CPL-P10:</b> Mampu menghasilkan rancangan, mengimplementasikan, dan mengevaluasi solusi berbasis algoritma dengan mempertimbangkan aspek kompleksitas.</p>
<b>Capaian Pembelajaran Mata Kuliah (CPMK) dan Sub-CPMK</b>	<p><b>CPMK05-2:</b> Mampu menerapkan (C3) konsep teoretis bidang pengetahuan dan keterampilan Ilmu Komputer dalam menyelesaikan permasalahan (P4) kompleks dengan pemikiran komputasional untuk pengambilan keputusan.</p> <p><b>SUB CPMK05-2:</b></p> <ol style="list-style-type: none"><li>1. (C3) Mampu memahami dan menerapkan konsep pemrograman terstruktur (procedural) yang meliputi: fungsi, prosedur, dan Abstract Data Type (ADT).</li><li>2. (C3) Mampu memahami dan menerapkan konsep algoritma searching.</li><li>3. (C3) Mampu memahami dan menerapkan konsep algoritma sorting.</li><li>4. (C3) Mampu memahami dan menerapkan konsep proses rekursif.</li></ol> <p><b>CPMK10-1:</b> Mampu menghasilkan rancangan dan mengimplementasi solusi berbasis algoritma untuk permasalahan sederhana.</p> <p><b>SUB CPMK10-1:</b></p>

	<ol style="list-style-type: none"> <li>1. (C3) Mampu memahami dan menerapkan penyelesaian masalah dengan menggunakan pendekatan algoritma yang terstruktur dan sistematis.</li> <li>2. (C3) Mampu memahami dan menerapkan notasi algoritmik untuk merepresentasikan solusi permasalahan secara algoritma.</li> <li>3. (C3) Mampu memahami dan menerapkan instruksi dasar yang meliputi: variable, konstanta, type data, operator, assignment, dan proses I/O.</li> <li>4. (C3) Mampu memahami dan menerapkan instruksi Analisa kasus (struktur kontrol), perulangan, pemrosesan sekuensial, dan konsep array.</li> </ol>
--	--

**Perhatian, Sebelum mengerjakan BACALAH bagian di bawah ini**

1. Kerjakanlah sendiri **dengan jujur**, jika diketahui **terjadi kecurangan** diberikan **nilai NOL**.
  2. Selain nomor 1, kerjakan dengan menggunakan **NOTASI ALGORITMIK**, jika terdapat notasi lain diberikan **nilai NOL**.
  3. Kerjakan dengan menggunakan **PENSIL**, jika menggunakan yang lain diberikan **nilai NOL**.
1. [CPMK10-1 bobot 15%] Diberikan potongan teks algoritma di bawah ini, pada akhir eksekusi berapa nilai akhir variable **jum**, jawaban anda disertai dengan proses step by stepnya untuk mendapatkan nilai akhir variable **jum** tersebut.

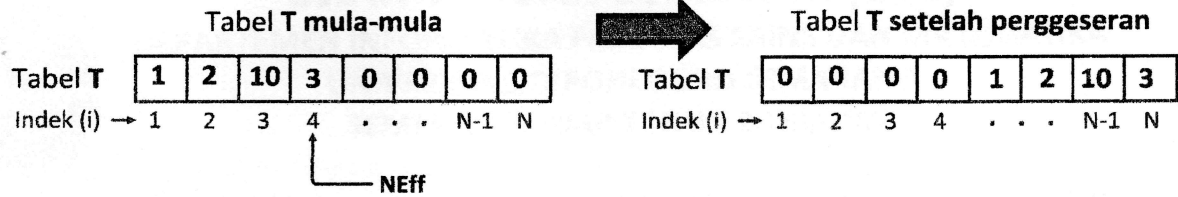
```

{Kamus}
  i, k, jum: integer
{Algoritma}
  i ← 1; j ← 0 {inisialisasi}
  while (i ≤ 4) do
    if ((i mod 2) = 0) then {i genap}
      k traversal [1..i]
      jum ← jum + i + k
    else {(i mod 2) ≠ 0} {i ganjil}
      jum ← jum + i + 1
    i ← i + 1
  {EndWhile}

```

2. [CPMK10-1 bobot 30%] Diberikan sebuah tabel integer T dengan ukuran 1 s/d N. Tabel T berisi elemen mulai dari posisi ke 1 s/d NEff (dimana  $N_{eff} \leq N$ , dan elemen pada posisi yang belum terisi atau setelah  $N_{eff}$  s/d N di set dengan nilai 0). Buatlah teks algoritma dalam bentuk program utama untuk melakukan pergeseran elemen-elemen tabel T dengan cara meletakkan elemen posisi  $N_{eff}$  pada posisi N,  $N_{eff}-1$  pada posisi N-1,  $N_{eff}-2$  pada posisi N-2, ... dst. Kemudian nilai elemen pada posisi sisa dari hasil pergeseran di set menjadi 0.

Contoh:



3. [CPMK10-1 bobot 25%] Diberikan sebuah type `TabInt:array [1..100]` of integer.  
Buatlah teks algoritma dalam bentuk sub program untuk fungsi dibawah ini.

**Function** `IsTableSimetri(T1: TabInt, T2: TabInt) → boolean`  
 {mengirimkan **TRUE** jika nilai setiap elemen T1 sama dengan T2}  
 {Tabel **T1** dan **T2** selalu memiliki panjang sama, sehingga tidak perlu}  
 {dilakukan pengecekan panjang T1 dan T2}  
**{Contoh:}**  
 {**T1** = <1 3 4 5 7 8 9>, **T2** = <1 3 4 5 7 8 9> maka **TRUE**}  
 {**T1** = <1 3 4 5 4 3 1>, **T2** = <1 3 4 5 6 8 9> maka **False**}  
  
**{Kamus Lokal}**  
  
**{Algoritma}**

4. [CPMK10-1 bobot 30%] Buatlah teks algoritma dalam bentuk program utama untuk menghitung banyaknya kata yang diakhiri dengan pasangan karakter 'LE' dari sebuah pita karakter. Definisi kata yang digunakan pada persolan ini adalah jika ketemu spasi, koma, dan titik. Anda dapat langsung menggunakan primitif Mesin Karakter (`START()`, `ADV()`, dan `EOP()`) tanpa harus melakukan realisasi.

**Contoh.** Pita karakter='Sale pisang lebih enak dibandingkan ikan lele, punya Sule.'

Dari pita karakter diatas akan menghasilkan 3 kata yang diakhiri dengan pasangan 'LE'.