

## **Topik: Review Konsep Dasar Pemrograman Prosedural dalam Bahasa C**

### **Tujuan:**

Memahami kembali konsep dasar pemrograman prosedural yang diimplementasikan dalam bahasa C.

### **Aturan Pelaksanaan:**

1. Tuliskan jawaban setiap soal di sebuah file teks yang diberi nama P00\_<nim>\_XXy.c dengan:
  - <nim> : NIM peserta.
  - XX : nomor soal (2 digit, contoh: 01).
  - y : butir soal (contoh: a), diisi hanya jika dibutuhkan.
2. Untuk setiap file yang Anda buat, cantumkan header sebagai berikut:

/* NIM>Nama : Nama file : Topik : Tanggal : Deskripsi : <div style="text-align: right;">*/</div>
---

3. Tugas ini bersifat INDIVIDUAL. Tidak ada toleransi bagi pencontek. Jika terbukti, baik yang dicontek maupun yang mencontek akan mendapatkan nilai 0.

Selamat bekerja ☺

## Deskripsi Soal

### Soal 1. Translasi Notasi Algoritmik ke Bahasa C

Translasikanlah program-program dalam notasi algoritmik sebagai berikut ke Bahasa C.

#### a. Sekuens

<b>Program</b> HELLOX { Menulis "HELLO" berikut nama yang diberikan dari piranti masukan ke piranti keluaran }
<b>KAMUS</b> name : string { nama informasi yang dituliskan pesannya, panjang=20 }
<b>ALGORITMA</b> <u>input</u> (name) <u>output</u> ("HELLO ", name)

Catatan: Panjang string pesan pada saat deklarasi di kamus sudah diketahui yaitu 20.

#### b. Sekuens

<b>Program</b> Tegangan { Dibaca arus I dan hambatan R, dihitung besarnya tegangan $V = I * R$ dan dituliskan hasilnya }
<b>KAMUS</b> I : <u>real</u> { arus listrik, ampere } R : <u>real</u> { hambatan, ohm } V : <u>real</u> { tegangan listrik, volt }
<b>ALGORITMA</b> <u>output</u> ("Input nilai arus : ") <u>input</u> (I) <u>output</u> ("Input nilai hambatan : ") <u>input</u> (R) $V \leftarrow I * R$ <u>output</u> ("Tegangan listrik = ", V, " volt")

#### c. Type Bentukan

<b>Program</b> DURASI { Input : JamAwal, JamAKhir bertipe Jam (HH:MM:DD); dan JamAKhir SELALU $\geq$ JamAwal Output : Selisih waktu dalam detik antara JamAwal dan JamAKhir }
<b>KAMUS</b> <u>type</u> Jam : < HH : <u>integer</u> [0..23], MM : <u>integer</u> [0..59], DD : <u>integer</u> [0..59] > JamAwal, JamAKhir : Jam Durasi : <u>integer</u> $\geq 0$
<b>ALGORITMA</b> <u>input</u> (JamAwal, JamAKhir) Durasi $\leftarrow$ (JamAKhir.HH * 3600 + JamAKhir.MM * 60 + JamAKhir.DD) - (JamAwal.HH * 3600 + JamAwal.MM * 60 + JamAwal.DD) <u>output</u> (Durasi)

#### d. Analisis Kasus

<b>Program</b> WUJUDAIR { Dibaca T(integer), temperatur air (dalam °C) pada tekanan atmosfer } { Harus dituliskan wujud air pada temperatur T: Beku, Cair atau Uap }
<b>KAMUS</b> T : <u>integer</u>
<b>ALGORITMA</b> <u>input</u> (T) <u>depend on</u> (T) $T \leq 0$ : <u>output</u> ("Beku") $0 < T \leq 100$ : <u>output</u> ("Cair") $T > 100$ : <u>output</u> ("Uap")

#### e. Fungsi

<b>Program</b> UbahdanPeriksaKar { Program membaca sebuah karakter, dan melakukan konversi ke nilai integer serta menuliskannya, jika karakter mewakili angka, bernilai ['0'..'9']. Jika yang diketik bukan bernilai ['0'..'9'], program memeriksa apakah karakter itu A atau bukan kemudian menuliskan pesan }
<b>KAMUS</b> CC : <u>character</u> {data, karakter yang dibaca}  <b>function</b> KarakterToInteger (x : <u>character</u> ['0'..'9']) → <u>integer</u> [0..9] { Diberikan x berupa karakter '0'..'9', menghasilkan harga integer yang sesuai dengan penulisan pada karakter }  <b>function</b> IsAnA (x : <u>character</u> ) → <u>boolean</u> { Menghasilkan <u>true</u> jika x adalah 'A' }
<b>ALGORITMA</b> <u>input</u> (CC) <u>depend on</u> (CC) $CC \in ['0'..'9']$ : <u>output</u> (KarakterToInteger(CC)) $CC \notin ['0'..'9']$ : <u>if</u> (IsAnA) <u>then</u> <u>output</u> ('Bukan angka, tapi merupakan karakter A') <u>else</u> <u>output</u> ('Bukan angka dan bukan karakter A')

<b>function</b> KarakterToInteger (x : <u>character</u> ['0'..'9']) → <u>integer</u> [0..9] { Diberikan x berupa karakter, menghasilkan harga integer yang sesuai dengan penulisan pada karakter }
<b>KAMUS LOKAL</b>
<b>ALGORITMA</b> <u>depend on</u> (x) x = '0' : → 0 x = '1' : → 1 x = '2' : → 2 x = '3' : → 3 x = '4' : → 4 x = '5' : → 5 x = '6' : → 6 x = '7' : → 7 x = '8' : → 8 x = '9' : → 9

**function** IsAnA (x : character) → boolean  
{ Menghasilkan true jika x adalah 'A' }

**KAMUS LOKAL**

**ALGORITMA**

→ (x = 'A')

## f. Prosedur

**Program** PUTARBIL  
{ Program yang membaca 3 buah integer dan "memutar" 3 buah integer tsb. }  
( Jika inputnya: a=A dan b=B, dan c=C, maka outputnya: a=C, b=A, c=B }

**KAMUS**

a, b, c : integer

**procedure** PROCTUKAR (input/output a, b : integer)  
{ Prosedur untuk menukar dua buah harga yang tersimpan dalam dua  
nama integer}  
{ I.S : diberikan a=A dan b=B }  
{ F.S : a=B dan b=A }

**ALGORITMA**

input (a, b, c)  
PROCTUKAR (a,c)            { a = C; b= B, c=A }  
PROCTUKAR (b,c)            { a = C; b= A, c=B }  
output (a, b, c)

**procedure** PROCTUKAR (input/output a, b : integer)  
{ I.S : diberikan a=A dan b=B }  
{ F.S : a=B dan b=A }

**KAMUS LOKAL**

Temp : integer

**ALGORITMA**

Temp ← a    { Temp = a; a = a; b = b }  
a ← b        { Temp = a; a = b; b = b }  
b ← Temp    { Temp = a; a = b; b = a }

### g. Pengulangan

```
Program SUMNBilX
{ Menjumlahkan dan mencacah (melakukan counting) nilai-nilai X yang dibaca.
Mark = 9999 }
```

## KAMUS

```
i : integer      { banyaknya nilai integer yang akan dibaca }
X : integer      { sekumpulan bilangan integer yang dibaca, diakhiri
                  dengan 9999}
Sum : integer    { jumlah }
```

## ALGORITMA

```

i ← 0; SUM ← 0                                { Inisialisasi }
 (X)                                     { First_Elmt }
while (X ≠ 9999) do
    output (X)
    Sum ← Sum + X
    i ← i + 1
     (X)                                { Next_Elmt }
{ i = bilangan ke... yang telah dibaca,
  Sum = X1 + X2 + ... + ... + Xi }
output ("Jumlah : ", Sum)                     { Terminasi }
output ("Banyaknya bilangan : ", i )

```

### h. Pengulangan

```

Program HITUNGDERET
{ Menghitung deret  $S = 1 - 1/2 + 1/3 - 1/4 + \dots + 1/999 - 1/1000$  }

```

## KAMUS

```
constant N : integer = 1000
i : integer           { indeks suku berikut yang akan dihitung }
TANDA : integer[-1,1] { tanda suku deret}
S : real ≥ 0.0        { Jumlah deret}
```

## ALGORITMA

```
TANDA ← 1
S ← 1      { i = 1, TANDA = 1, S = 1 }
i ← 2      { suku berikutnya yang akan dihitung }
while (i ≤ N) do
    { S = 1 - 1/2 + 1/3 - 1/4 + ... + tanda/(i-1) }
    TANDA ← -TANDA
    S ← S + TANDA/i
    i ← i + 1
{ i = N + 1 and S = 1 - 1/2 + 1/3 - 1/4 + ... + 1/999 - 1/N, N = 1000 }
output (S)
```

**Soal 2. Cek Batas Nilai Tipe Dasar (BONUS) – tulis sebagai satu *source file* P00\_<NIM>\_02.c**

Buatlah sebuah program dalam bahasa C yang melakukan pengetesan terhadap batas maksimum dan minimum nilai-nilai beberapa tipe dasar yang disediakan bahasa C dengan menggunakan beberapa konstanta yang didefinisikan dalam library **limits.h**, yaitu:

- a. Nilai maksimum integer (INT\_MAX)
- b. Nilai minimum integer (INT\_MIN)
- c. Nilai maksimum short integer (SHRT\_MAX)
- d. Nilai minimum short integer (SHRT\_MIN)
- e. Nilai maksimum long integer (LONG\_MAX)
- f. Nilai minimum long integer (LONG\_MIN)
- g. Nilai maksimum unsigned integer (UINT\_MAX)
- h. Nilai maksimum unsigned short integer (USHRT\_MAX)
- i. Nilai maksimum unsigned long integer (ULONG\_MAX)
- j. Nilai maksimum character (CHAR\_MAX)
- k. Nilai minimum character (CHAR\_MIN)
- l. Nilai maksimum signed character (SCHAR\_MAX)
- m. Nilai minimum signed character (SCHAR\_MIN)
- n. Nilai maksimum unsigned character (UCHAR\_MAX)
- o. Nilai maksimum float (FLT\_MAX)
- p. Nilai minimum float (FLT\_MIN)
- q. Nilai maksimum double float (DBL\_MAX)
- r. Nilai minimum double float (DBL\_MIN)
- s. Nilai maksimum long double float (LDBL\_MAX)
- t. Nilai minimum long double float (LDBL\_MIN)

Kerangka program:

```
#include <limits.h>
... /* include library lain jika diperlukan */

int main () {
    /* KAMUS */

    /* ALGORITMA */
    /* Pengecekan Integer */
    printf("Maximum integer: %d\n", INT_MAX);
    printf("Minimum integer: %d\n", INT_MIN);
    ... /* lanjutkan dengan pemeriksaan batas-batas lain */

    return 0;
}
```

Tuliskan sebagai komentar pada program ini, hasil pemeriksaan terhadap nilai-nilai batas tersebut.