

Contoh dan Latihan Struktur Dasar Algoritma

Agus Sumaryanto, S.Kom
mas.anto72@gmail.com

RUNTUNAN (sequence)

Contoh Runtunan (1)

```
01| Algoritma Contoh_Runtunan_1
02| Deklarasi:
03|   a, b : integer
04| Deskripsi
05|   a ← 10
06|   a ← 2*a
07|   b ← a
08|   write(b) {mencetak nilai 20}
```

```
01| Algoritma Contoh_Runtunan_2
02| Deklarasi:
03|   a, b : integer
04| Deskripsi
05|   a ← 10
06|   b ← a
07|   a ← 2*a
08|   write(b) {mencetak nilai 10}
```

Contoh Runtunan (2)

```
01| Algoritma Menghitung_Luas_Segitiga
02| {Menghitung luas segitiga jika diketahui panjang alas dan tinggi
    segitiga. Alas dan tinggi diinput dari user. Kemudian Luas
    dihitung menggunakan rumus Luas = 0,5 x alas x tinggi. Kemudian
    mencetak output berupa luas segitiga}
03| Deklarasi:
04|   a, t, L : real
05| Deskripsi
06|   read(a,t)
07|   L ← 0.5 * a * t
08|   write(L)
```

Contoh Runtunan (3)

Buatlah algoritma untuk menghitung komisi yang diterima salesman berdasarkan jumlah penjualan yang dicapainya. Salesman tersebut mendapat komisi 10% dari hasil penjualannya. Input algoritma ini adalah nama salesman dan jumlah penjualan yang dicapainya. Sedangkan outputnya adalah nama salesman dan besar komisi yang diperolehnya.

Analisis: input: nama salesman (nama) dan jumlah penjualan (j)

Output: nama salesman (nama) dan besar komisi (komisi) rumus: komisi = 10% x jumlah penjualan

Langkah pengerjaan:

1. menginput data nama salesman dan jumlah penjualan
2. menghitung komisi menggunakan rumus
3. mencetak nama dan komisi

Contoh Runtunan (3.1)

```

01| Algoritma Menghitung_Komisi_Salesman
02| {Menghitung besarnya komisi yang diperoleh salesman berdasarkan
    jumlah penjualan. Besarnya komisi adalah 10% dari jumlah penjualan
    yang dicapainya. Kemudian algoritma akan mencetak nama salesman dan
    komisi sebagai outputnya.}
03| Deklarasi:
04|   nama : string   {nama salesman}
05|   j : integer    {jumlah penjualan}
06|   komisi : real  {komisi yang diperoleh}
07| Deskripsi
08|   read(nama, j)
09|   komisi ← 0.1 * j
10|   write(nama, komisi)
  
```

Latihan Runtunan (1)

1. Buatlah algoritma untuk menghitung gaji karyawan. Diberikan nama karyawan dan besarnya gaji pokok. Gaji bersih yang diterima pegawai adalah gaji pokok ditambah besarnya tunjangan kemudian dikurangi pajak. Tunjangan karyawan dihitung 20% dari gaji pokok, sedangkan pajak adalah 15% dari gaji pokok ditambah tunjangan. Keluaran yang diharapkan adalah nama karyawan, besarnya tunjangan, pajak, dan gaji bersihnya.

Analisis:

input: nama karyawan (nama) dan besarnya gaji pokok (gaji_pokok)

output: nama karyawan (nama), tunjangan (tunj), pajak (pjk), dan gaji bersih (gaji_bersih)

rumus: $tunj = 20\% \times \text{gaji_pokok}$

$pjk = 15\% \times (\text{gaji_pokok} + \text{tunj})$

$\text{gaji_bersih} = \text{gaji_pokok} + \text{tunj} - \text{pajak}$

Langkah pengerjaan: 1. menginput nama karyawan dan gaji pokok

2. menghitung tunjangan

3. menghitung pajak

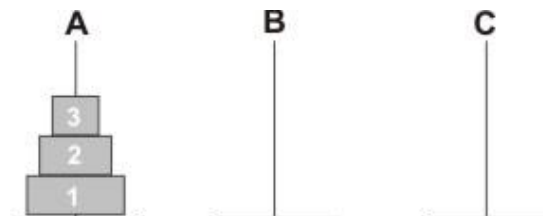
4. menghitung gaji bersih

5. mencetak nama, tunjangan, pajak, dan gaji bersih

Latihan Runtunan (2)

2. Menara Hanoi. Jika ada 3 buah balok disusun pada sebuah tonggak A dengan aturan pokok bahwa balok yang lebih kecil harus berada diatas yang lebih besar, pindahkan balok tersebut dari tonggak A ke tonggak C, dengan bantuan tonggak B. Aturan pokok tidak boleh dilanggar. Berapa kali langkah untuk memindahkan balok tersebut.

Untuk lebih jelas perhatikan gambar dibawah ini :



3. Tuliskan langkah-langkah pemindahan untuk 7 balok.

PEMILIHAN (selection)

Contoh Pemilihan (1)

- Buatlah algoritma untuk mencetak kata “negatif” apabila user menginputkan suatu bilangan negatif. Analisis: Suatu bilangan dikatakan negatif apabila bilangan tersebut kurang daripada nol. Langkah pengerjaan: 1. Input suatu bilangan, misalkan n 2. Cek kondisi $n < 0$. Apabila bernilai true, cetaklah kata “negatif”.

```
01| Algoritma Pengecekan_Negatif
02| {Diinput suatu bilangan, apabila negatif, maka cetaklah kata
    "negatif"}
03| Deklarasi:
04|   n : real
05| Deskripsi
06|   Read(n)
07|   If  $n < 0$  then
08|     write('negatif')
09|   end if
```

Contoh Pemilihan (2)

Sebuah toko yang sedang melakukan promosi memberikan diskon sebesar 10% untuk pembeli yang melakukan transaksi minimal Rp100.000,00. Buatlah algoritma untuk menampilkan diskon dan total yang harus dibayar pembeli. Analisis: input: Total belanja pembeli (belanja) output: besarnya diskon (diskon) dan total pembayaran (total) Langkah pengerjaan:

1. input total belanja pembeli
2. cek kondisi belanja = Rp100.000,00. Apabila bernilai true maka menghitung diskon dengan rumus $\text{diskon} = 10\% \times \text{belanja}$.
3. Hitung total pembayaran dengan rumus $\text{total} = \text{belanja} - \text{diskon}$
4. Cetak nilai diskon dan total pembayaran.

Contoh Pemilihan (2.1)

```

01| Algoritma Pemberian_Diskon
02| {Diinput total belanja pembelian. Diskon sebesar 10% diberikan
    dengan syarat total belanja minimal Rp100.000,-. Algoritma
    memberikan output besarnya diskon yang diperoleh dan total yang
    harus dibayarkan pembeli.}
03| Deklarasi:
04|   belanja : integer
05|   diskon, total : real
06| Deskripsi
07|   read(belanja)
08|   if belanja >= 100000 then
09|     diskon ← 0.1 * belanja
10|   end if
11|   total ← belanja - diskon
12|   write(diskon, total)
  
```

Latihan Pemilihan (1)

Karyawan PT "ABC" digaji berdasarkan jumlah jam kerjanya selama satu minggu. Upah per jam adalah Rp2.000,00. Bila jumlah jam kerja lebih besar dari 48 jam, maka sisanya dianggap sebagai jam lembur. Upah lembur adalah Rp3.000,00. Buatlah algoritma untuk menampilkan upah normal, uang lembur, dan total upah yang diterima karyawan. Analisis: input: jumlah jam kerja (n) output: upah normal (upah), uang lembur (lembur), dan total upah (total) Upah per jam, upah lembur, dan batas jam lembur dapat dijadikan sebagai konstanta. Kasus ini memberikan kita dua kemungkinan, yaitu apakah karyawan menerima lembur atau tidak menerima lembur (lembur = 0). Syarat seorang karyawan menerima lembur adalah apabila $n > 48$. Uang lembur yang diterima adalah selisih jam kerja dengan batas jam lembur dikalikan dengan upah lembur. Oleh karena itu, dapat kita simpulkan bahwa rumus yang dipakai adalah Apabila karyawan tidak mendapat uang lembur, maka lembur = 0, upah = $n \times 2000$ Apabila karyawan mendapatkan uang lembur, maka lembur = $(n - 48) \times 3000$, sedangkan upah = 48×2000 , **bukan upah = $n \times 2000$** .

Latihan Pemilihan (2)

Suatu perusahaan menentukan gaji pokok karyawannya menurut golongannya. Besarnya gaji pokok berdasarkan golongannya dapat dilihat pada tabel berikut.

Golongan	Gaji Pokok
A	Rp400.000,00
B	Rp500.000,00
C	Rp750.000,00
D	Rp900.000,00

Buatlah algoritma untuk menentukan gaji pokok berdasarkan golongan yang diinput user.

PENGULANGAN (repetition)

Contoh Pengulangan (1)

Buatlah algoritma yang meminta input n buah data bilangan dari user kemudian menghitung rata-ratanya. Oleh karena itu, kita harus mengulang perintah **read(data)** sebanyak n kali. **Kemudian rumus penjumlahannya adalah $\text{jumlah} = \text{jumlah} + \text{data}$.**

Setelah memperoleh jumlah, mencari rata-rata dapat menggunakan rumus **jumlah/n .**

Contoh Pengulangan (1.1)

```

01| Algoritma Hitung_Rata_rata_dari_n_data
02| {meminta input n buah data dari user kemudian menghitung rata-
    rata_ratanya.}
03| Deklarasi:
04|   data, jumlah : real {karena soal hanya menyebut bilangan}
05|   k, n : integer {harus integer karena pencacah}
06|   rata : real {untuk menampung hasil perhitungan rata-rata}
07| Deskripsi
08|   read(n)
09|   jumlah ← 0
10|   for k ← 1 to n do
11|     read(data)
12|     jumlah ← jumlah + data
13|   end for
14|   rata ← jumlah / n
15|   write(rata)
  
```

Contoh Pengulangan (2)

Buatlah algoritma untuk meminta input n buah data nilai mahasiswa kemudian menghitung banyaknya mahasiswa yang lulus. Syarat untuk lulus adalah nilai yang diperoleh adalah minimal 60.

Analisis: Soal ini menggunakan teknik yang sama dengan algoritma nomor 4. di atas. Tetapi, ingat bahwa yang dijumlah bukanlah nilainya, tetapi angka 1, yaitu menggunakan rumus banyak = banyak + 1. Rumus ini hanya dikerjakan apabila mahasiswa lulus, dengan kata lain nilai mahasiswa lebih besar atau sama dengan 60.

Contoh Pengulangan (2.1)

```
01| Algoritma Hitung_Banyaknya_Lulus
02| {menginput n buah data nilai, lalu menghitung banyaknya mahasiswa
    yang lulus}
03| Deklarasi:
04|   banyak : integer
05|   k, n : integer
06|   nilai : real
07| Deskripsi
08|   read(n)
09|   banyak ← 0
10|   for k ← 1 to n do
11|     read(nilai)
12|     if nilai >= 60 then
13|       banyak ← banyak + 1
14|     end if
15|   end for
16|   write(banyak)
```