

Bab 3

ALGORITMA RUNTUNAN

Pada Bab ini anda akan mempelajari

1. Pengertian algoritma runtunan
2. Ciri-ciri algoritma runtunan
3. Beberapa contoh soal algoritma runtunan

Pada dasarnya algoritma penyelesaian suatu masalah, betapapun kompleksnya, dapat dipecah ke dalam tiga struktur dasar algoritma, yaitu

1. Algoritma runtunan (sekuensial)
2. Algoritma seleksi kondisi atau percabangan
3. Algoritma perulangan

Pada bab ini akan dibahas struktur algoritma yang paling sederhana : runtunan (sekuensial).

1. Algoritma Runtunan

Diantara ketiga struktur dasar algoritma di atas, algoritma runtunan merupakan algoritma yang paling sederhana. Algoritma runtunan adalah sekumpulan perintah atau pernyataan yang dikerjakan komputer berdasarkan dengan urutan perintahnya. Jadi, jika terdapat algoritma dengan urutan perintahnya adalah :

Perintah1

Perintah2

Perintah3

maka perintah yang dieksekusi pertama adalah Perintah1, diikuti kemudian Perintah2 dan terakhir Perintah3.

Studi Kasus

Berikut ini adalah algoritma menghitung luas lingkaran. Secara umum algoritma tersebut dapat kita nyatakan sebagai berikut.

1. Mulai
2. Masukkan nilai jari-jari
3. Hitung 3.14 dikali jari-jari, simpan sebagai Luas.
4. Tampilkan nilai Luas
5. Selesai

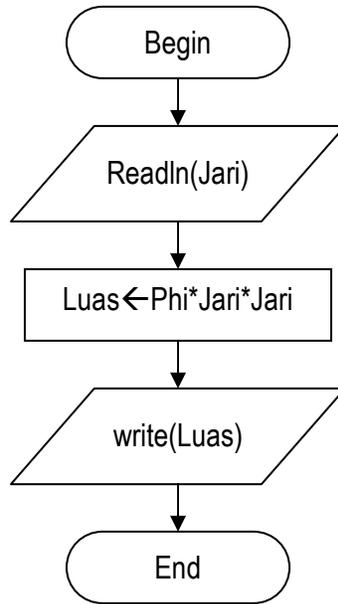
Dalam algoritma teks standar, solusi permasalahan di atas dapat kita tuliskan

Algoritma 3.1.

```
Algoritma LuasLingkaran
Deklarasi
  const
    Phi = 3.14
  var
    Luas, Jari : real

Deskripsi
  Readln(Jari)
  Luas ← Phi*Jari*Jari
  Write(Luas)
```

Dalam bentuk flowchart, algoritma teks di atas dapat kita tuliskan



Phi adalah konstanta, sedangkan **Luas** dan **Jari** merupakan variabel.

Dari contoh di atas dapat dilihat bahwa perintah kedua dikerjakan setelah perintah pertama, perintah ketiga setelah perintah kedua. Urutan ini memegang peranan penting karena urutan berbeda dapat menghasilkan keluaran yang berbeda (walaupun tidak selalu demikian). Jika urutan perintah pertama dan kedua dibalik, maka akan menghasilkan kesalahan logika karena tidaklah mungkin menghitung perkalian 3.14 dengan jari-jari sedangkan nilai jari-jarinya sendiri belum diketahui.

Berikut ini adalah contoh perbandingan antara dua algoritma dengan perintah-perintah yang sama tetapi berbeda dalam urutan sehingga menghasilkan keluaran yang berbeda..

```

Algoritma Runtunan1

Deklarasi
var
  A,B : integer

Deskripsi
  A ← 10
  A ← 2 * A
  B ← A
  Write (B)
  
```

```

Algoritma Runtunan2

Deklarasi
var
  A,B : integer

Deskripsi
  A ← 10
  B ← A
  A ← 2 * A
  Write (B)
  
```

Perhatikan algoritma **Runtunan1**. Mula-mula A kita beri nilai 10. Pada pernyataan berikutnya nilai A menjadi dua kali A sebelumnya sehingga menjadi 20. Kemudian pada B kita simpan nilai yang sama dengan nilai A yang terakhir, yaitu 20. Pada perintah akhir kita tampilkan isi dari B, sehingga muncul angka 20. Perhatikan tabel media penyimpanan berikut ini.

Table 3.1.

Perintah	A	B	Output
$A \leftarrow 10$	10		
$A \leftarrow 2 * A$	20		
$B \leftarrow A$		20	
Write (B)			20

Sekarang beralih ke **Runtunan2**. Mula-mula A diberi nilai 10. Kemudian pada B disimpan nilai yang sama dengan A, yaitu 10. Lalu A diberi nilai dua kali nilai A sebelumnya sehingga menjadi 20. Akhir algoritma memerintahkan menampilkan B, yaitu nilainya 10. Jadi keluaran **Runtunan1** akan menampilkan output 20, sedangkan **Runtunan2** menghasilkan output 10. Perhatikan media penyimpanan data berikut ini.

Tabel 3.2.

Perintah	A	B	Output
$A \leftarrow 10$	10		
$B \leftarrow A$		10	
$A \leftarrow 2 * A$	20		
Write (B)			10

Dari kedua contoh tersebut terlihat bahwa meskipun perintah yang ada pada **Runtunan1** dan **Runtunan2** adalah sama, tetapi urutan yang berbeda menyebabkan ouputnya juga berbeda.

Studi Kasus

Seorang pedagang mangga menjual dagangannya yang setiap kg mangga dihargai dengan harga tertentu. Setiap pembeli membayar harga mangga yang dibelinya berdasarkan berat. Tentukan perintah apa yang diberikan pedagang mangga kepada komputer agar ia dapat dengan mudah menentukan harga yang harus pembelinya.

Identifikasi masalah

Input : harga per kg mangga (hrg), berat pembelian (brt)

Output : harga yang dibayar pembeli (hrg)

Perintah teks umumnya dapat ditulis sebagai berikut

1. Mulai
2. Masukkan harga mangga per kg (hrg)
3. Masukkan berat pembelian (brt)
4. Kalikan hrg dengan brt, simpan sebagai harga yang harus dibayar pembeli (byr)
5. Tampilkan nilai byr
6. Selesai

Sedangkan perintah algoritma teks standarnya dapat ditulis

Algoritma 3.2.

Algoritma BeliMangga

Deklarasi

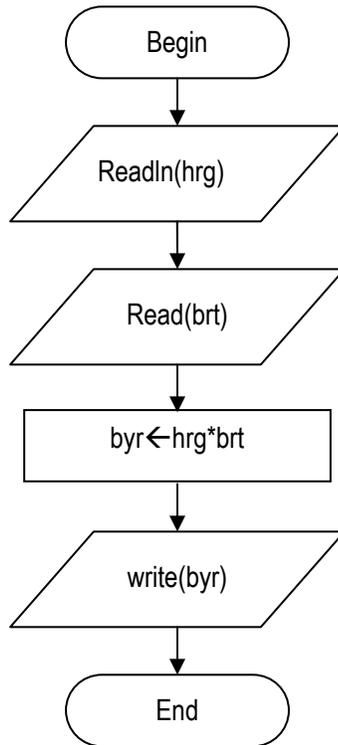
```
var  
    Hrg, brt, byr : real
```

Deskripsi

```
    read(hrg)  
    read(brt)  
    byr ← hrg*brt  
    write(byr)
```

Perhatikan : untuk operasi perkalian gunakan tanda (*) sebagai operator pengali.

Dalam bentuk flowchart, algoritmanya dapat ditulis sebagai berikut.



Studi Kasus

Soni mempunyai kelereng sebanyak 15 buah. Jumlah itu 10 buah lebih banyak dari kelereng Adi. Sedangkan Anis memiliki kelereng sebanyak 2 x kelereng Soni dan Adi. Luki memiliki kelereng sebanyak 5 buah lebih sedikit dari jumlah kelereng Soni, Adi dan Anis. Carilah banyak kelereng Adi, Anis dan Luki, jika diketahui jumlah kelereng Soni.

Identifikasi masalah

Input : banyak kelereng Soni

Output : banyak kelereng Adi, Anis, Luki

Algoritma 3.3.

Algoritma HitungKelereng

{Algoritma menghitung kelereng Adi, Anis, dan Luki}

Deklarasi

Const

Ksoni = 15

var

Kadi, Kanis, Kluki : integer

Deskripsi

Kadi \leftarrow Ksoni-10

Kanis \leftarrow 2*(Ksoni+Kadi)

Kluki \leftarrow Ksoni+Kadi+Kanis-5

Write(Kadi, Kanis, Kluki)

Tabel 3.3.

Perintah	KSoni	KAdi	KAnis	KLuki	Output
Readln(Ksoni)	15				
KAdi \leftarrow Ksoni-10		5			
KAnis \leftarrow 2*(Ksoni+KAdi)			40		
KLuki \leftarrow Ksoni+KAdi+KAnis-5				55	
Write(KAdi, KAnis, Kluki)					5 40 55

Dengan demikian keluaran dari algoritma di atas adalah : 5 40 55

Studi Kasus

Berikut ini adalah algoritma menukar angka yang disimpan dalam dua buah variabel A dan B

Identifikasi masalah

Input : Nilai A dan B (lama)

Output : Nilai A dan B (baru)

Algoritma 3.4.

Algoritma TukarIsi

Deklarasi

```
var  
    A, B, temp : integer
```

Deskripsi

```
read(A,B)  
temp ← A  
A ← B  
B ← temp  
write(A,B)
```

Tabel 3.4.

Perintah	A	B	Temp	Output
read(A,B)	500	200		
temp ← A			500	
A ← B	200			
B ← temp		500		
write(A,B)				200 500

Studi Kasus

Berikut ini adalah algoritma untuk mengkonversi waktu dalam satuan detik ke dalam satuan jam, menit, detik.

Identifikasi masalah

Input : satuan waktu dalam detik

Output : satuan waktu dalam jam, menit, dan detik

Algoritma 3.5.

Algoritma KonversiDetik

{Algoritma mengkonversi waktu dari satuan detik ke satuan jam, menit, dan detik}

Deklarasi

var

Detik, KonvJam, KonvMnt, KonvDtk, Sisa : integer
{Mengapa tipe integer, bukan real?}

Deskripsi

read(Detik)
KonvJam \leftarrow Detik div 3600
Sisa \leftarrow Detik mod 3600
KonvMnt \leftarrow Sisa div 60
KonvDtk \leftarrow Sisa mod 60
write(KonvJam, KonvMnt, KonvDtk)

Dengan masukan Detik = 5000, maka tabel penyimpanan data dapat ditulis sebagai berikut :

Tabel 3.5.

Perintah	Detik	KonvJam	Sisa	KonvMnt	KonvDtk	Output
read(Detik)	5000					
KonvJam \leftarrow Detik div 3600		1				
Sisa \leftarrow Detik mod 3600)			1400			
KonvMnt \leftarrow Sisa div 60				23		
KonvDtk \leftarrow Sisa mod 60					20	
write(KonvJam, KonvMnt, KonvDtk)						1 23 20

Dengan demikian keluaran algoritma adalah : 1 23 20

Latihan

1. Tuliskan keluaran algoritma di atas jika input Detik =12000
2. Buatlah flowchart dari algoritma di atas.

Soal Latihan

1. Diketahui algoritma berikut ini

Algoritma UtakAtik1

Deklarasi

```
var
  A, B, C : integer
  X : boolean
```

Deskripsi

```
Read(A)
B ← 0.5 * A + 15
C ← A * A - 3
X ← B > C
Write(B, C, X)
```

- a. Tentukan keluaran algoritma untuk $A = 4$
- b. Tentukan keluaran algoritma untuk $A = 5$

2. Diketahui algoritma berikut ini

Algoritma UtakAtik2

Deklarasi

```
var
  A, B, C : integer
  X : boolean
```

Deskripsi

```
Read(A, B)
A ← A mod 5
C ← B div A
Y ← A <> B
X ← Y or A > C
Write(B, C, X)
```

- a. Tentukan keluaran algoritma untuk $A = 14, B = 10$
- b. Tentukan keluaran algoritma untuk $A = 23, B = 11$

3. Buatlah algoritma menampilkan tulisan “Halo” sebanyak tiga kali

4. Buatlah algoritma menghitung panjang sisi miring dari sebuah segitiga siku-siku.
Input : alas, tinggi
5. Buatlah algoritma untuk menghitung luas segitiga yang panjang sisi terpanjangnya sama dengan diameter sebuah lingkaran. Input : jari-jari lingkaran yang dimaksud.
6. Buatlah algoritma untuk menentukan sumbu x
7. Buatlah algoritma mencari jarak terdekat antara dua buah titik A dan B yang terletak dalam satu bidang datar. Input : koordinat A(x,y) dan B(x,y).
8. Buatlah algoritma untuk menghitung besar hambatan pengganti untuk tiga buah hambatan yang dipasang secara paralel. Input : ketiga hambatan R1, R2, R3.
9. Buatlah algoritma untuk mengkonversi waktu dari satuan jam, menit, dan detik ke dalam satuan detik
10. Buatlah algoritma mengkonversi mata uang Rupiah ke dalam mata uang Yen dengan input : 1) nilai Yen untuk setiap satu Dolar, 2) Nilai Rupiah untuk setiap satu Dolar.