



Algoritma



# Algoritma Pemrograman

Oleh : Iosi Pratama

<http://www.iosinotes.blogspot.com>





Pengertian

# Algoritma

Algoritma adalah urutan aksi-aksi yang dinyatakan dengan jelas dan tidak rancu untuk memecahkan suatu masalah dalam rentang waktu tertentu .

suatu prosedur yang jelas untuk menyelesaikan persoalan dengan menggunakan langkah-langkah tertentu dan terbatas jumlahnya

Ciri – ciri

# Algoritma

Menurut Donald E . Knuth

1. Algoritma mempunyai awal dan akhir . Suatu algoritma harus berhenti setelah mengerjakan serangkaian tugas dengan kata lain algoritma harus memiliki langkah terbatas .
2. Setiap langkah harus didefinisikan dengan tepat sehingga tidak memiliki arti ganda (not ambiguous)
3. Memiliki masukan (input) atau kondisi awal
4. Memiliki keluaran (output) atau kondisi akhir
5. Algoritma harus efektif , bila diikuti benar-benar akan menyelesaikan masalah .



# Sifat - sifat Algoritma

## Sifat Algoritma

1. Input / Masukan : Suatu algoritma memiliki input atau kondisi awal sebelum algoritma dilaksanakan dan bisa berupa nilai-nilai pengubah yang diambil dari himpunan khusus
2. Output / Pengeluaran : Suatu algoritma akan menghasilkan output setelah dilaksanakan , atau algoritma akan mengubah kondisi awal menjadi kondisi akhir , dimana nilai output diperoleh dari nilai input yang diproses melalui algoritma
3. Definiteness / Pendefinisian : langkah-langkah yang dituliskan dalam algoritma terdefinisi dengan jelas sehingga mudah dilaksanakan oleh pengguna algoritma
4. Finiteness / Penyelesaian : Suatu algoritma harus memberi kondisi akhir atau output setelah melakukan sejumlah langkah yang terbatas jumlahnya untuk setiap kondisi awal atau input yang diberikan
5. Effectiveness / Efektifitas : Setiap langkah dalam algoritma bisa dilaksanakan dalam suatu selang waktu tertentu sehingga pada akhirnya memberi solusi sesuai yang yang diharapkan
6. Generality / Umum : Langkah-langkah algoritma berlaku untuk setiap himpunan input yang sesuai dengan persoalan yang akan diberikan , tidak hanya untuk himpunan tertentu .

## Cara Penulisan **Algoritma**



Deskriptif

Flowchart

Pseudocode





## Deskriptif

Algoritma bertipe deskriptif maksudnya adalah algoritma yang ditulis dalam bahasa manusia sehari-hari (misalnya bahasa Indonesia atau bahasa Inggris) dan dalam bentuk kalimat. Setiap langkah algoritmanya diterangkan dalam satu atau beberapa kalimat.

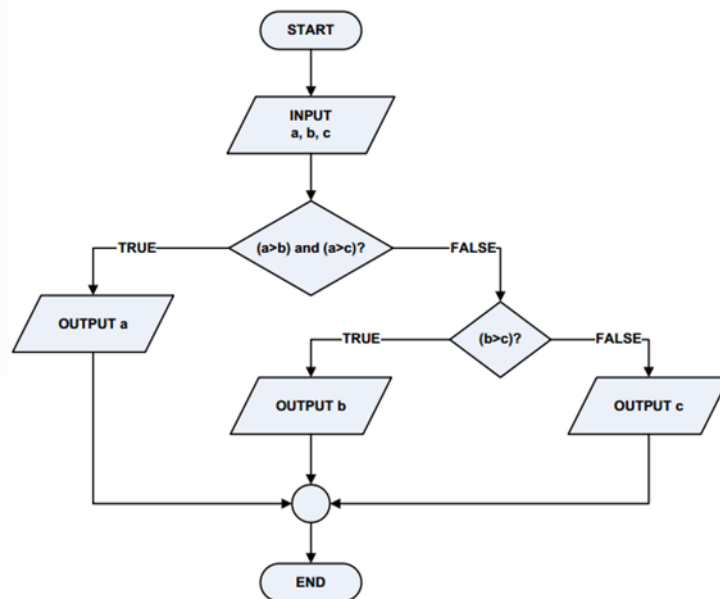
Sebagai contoh misalnya algoritma menentukan bilangan terbesar dari 3 bilangan berikut ini:

Algoritma Menentukan\_bilangan\_terbesar\_dari\_3\_bilangan

1. Meminta input 3 bilangan dari user, misalkan bilangan a, b, dan c.
2. Apabila bilangan a lebih besar dari b maupun c, maka bilangan a merupakan bilangan terbesar
3. Jika tidak (bilangan a tidak lebih besar dari b atau c) berarti bilangan a sudah pasti bukan bilangan terbesar. Kemungkinannya tinggal bilangan b atau c. Apabila bilangan b lebih besar dari c, maka b merupakan bilangan terbesar. Sebaliknya apabila bilangan b tidak lebih besar dari c, maka bilangan c merupakan yang terbesar.
4. Selesai.

## Flowchart

Selain dalam bentuk tulisan, algoritma juga dapat ditulis dalam bentuk diagram-diagram dengan anak panah sebagai penunjuk urutan langkah algoritmanya. Algoritma yang ditulis dengan simbol-simbol demikian yang dinamakan flow chart. Mengenai lambang-lambang yang digunakan akan dibahas pada bagian selanjutnya. Sekarang diberikan suatu contoh algoritma menentukan bilangan terbesar dari 3 bilangan seperti yang dicontohkan sebelumnya, tetapi ditulis dalam bentuk flow chart.





## Pseudocode

Pseudo berarti imitasi dan code berarti kode yang dihubungkan dengan instruksi yang ditulis dalam bahasa komputer (kode bahasa pemrograman). Apabila diterjemahkan secara bebas, maka pseudocode berarti tiruan atau imitasi dari kode bahasa pemrograman

Contoh algoritma menentukan bilangan terbesar dari tiga bilangan yang ditulis dalam bentuk pseudocode bergaya buku ini.

```
01| ALGORITMA Menentukan_terbesar_dari_3_bilangan
02| Deklarasi:
03| a,b,c, terbesar : integer
04|
05| Deskripsi:
06|   Read(a,b,c)
07|   If (a>b) and (a>c) then
08|     Terbesar ← a
09|   Else
10|     If b>c then
11|       Terbesar ← b
12|     Else
13|       Terbesar ← c
14|     Endif
15|   Endif
16|   Write(terbesar)
```





# Struktur Algoritma

1.

Runtunan / Sekuensial

2.

Seleksi / Selection

3.

Perulangan / Repeteteion

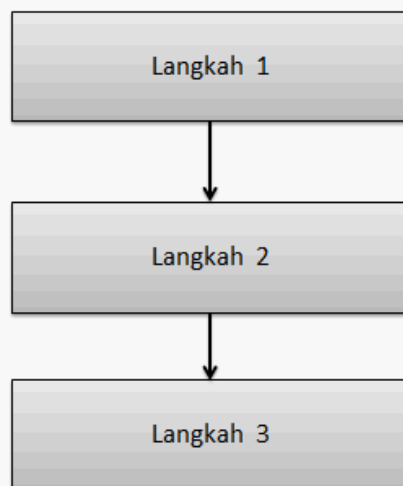
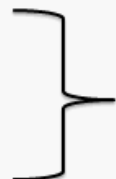
## 1. Runtunan / Sekuensial

Runtunan atau istilah lainnya Sekuensial . Pada Struktur Runtunan , langkah-langkah dilakukan dan diproses secara berurutan . Dimulai dari langkah pertama , kemudian kedua , ketiga , dan seterusnya hingga akhir . Pada dasarnya suatu program memang menjalankan suatu proses dari yang dasar seperti struktur seperti ini .

Langkah 1

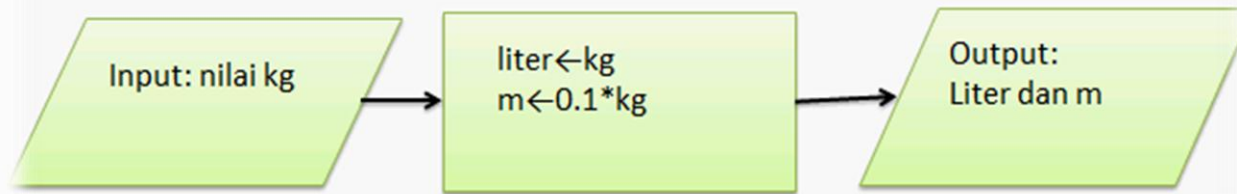
Langkah 2

Langkah 3



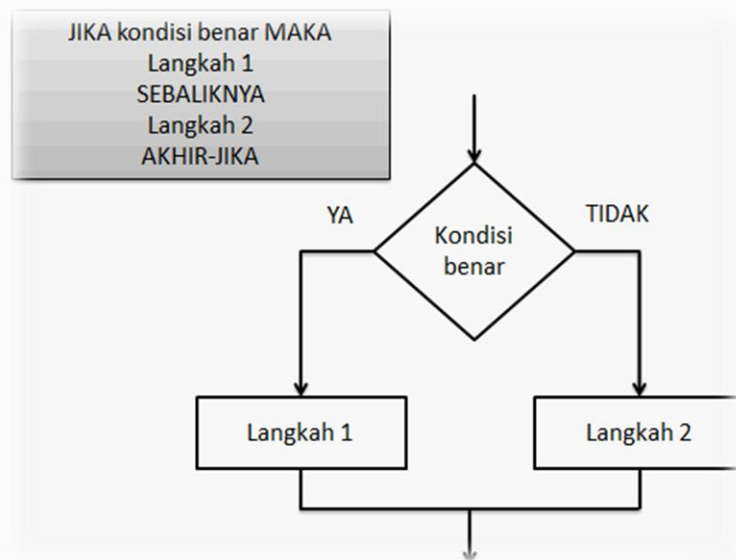
Contoh program yang dijalankan menggunakan struktur Runtunan (Sekuensial)

Flowchart konversi kg ke liter dan m

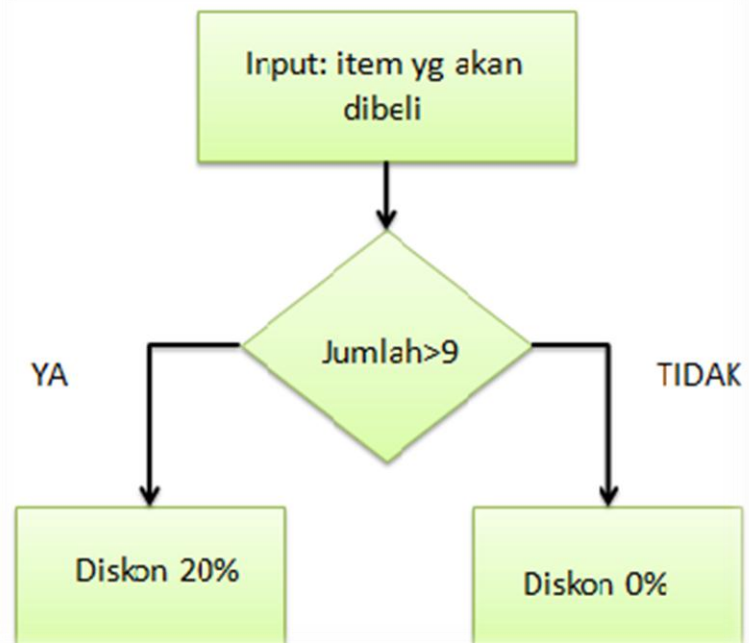


## 2. Seleksi / Selection

Struktur Seleksi dalam bahasa inggris (selection) atau Percabangan adalah struktur algoritma yang menyatakan pemilihan langkah yang didasarkan oleh suatu kondisi atau pengambilan suatu keputusan . Ciri utama dari struktur Percabangan adalah adanya bentuk flowchart belah ketupat (decision).

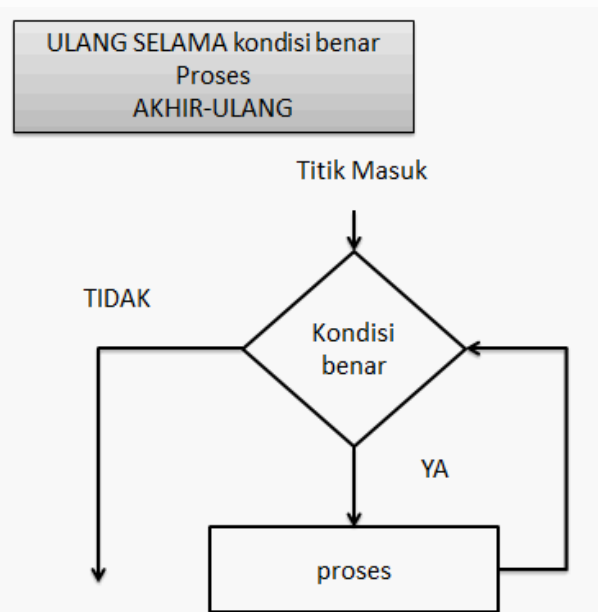


Contoh dari Struktur Seleksi / Percabangan pada saat pengambilan keputusan diantaranya bila terdapat diskon yang berbeda berdasarkan jumlah barang yang ingin dibeli



### 3. Perulangan / Repeteteion

Struktur perulangan memberikan suatu perintah atau tindakan yang dilakukan beberapa kali . Misalnya bila kita menulis “Saya lagi belajar Algoritma Pemrograman” , diproses 10 kali , maka agar efisien kita menggunakan struktur perulangan daripada menuliskannya satu-persatu hingga sepuluh kali.





Anda dapat menyebarkan PPT ini dengan tujuan sebagai bahan pembelajaran

Semoga Materi yang saya sampaikan tersebut dapat dimengerti dan mudah dipahami oleh semua pembaca

Untuk kedepannya saya sangat mengharapkan kritik dan saran dari pembaca agar materi ini semakin baik lagi

TTD



Iosi Notes

iosinotes.blogspot.com