

## # ONE

# PENGANTAR ALGORITMA PEMROGRAMAN

### Definisi Program/Pemrograman

- Adalah kumpulan instruksi-instruksi tersendiri yang biasanya disebut source code yang dibuat oleh programmer (pembuat program)

### Paradigma Pemrograman

1. Pemrograman Prosedural
  - Berdasarkan urutan-urutan, sekuensial
  - Program adalah suatu rangkaian prosedur untuk memanipulasi data. Prosedur merupakan kumpulan instruksi yang dikerjakan secara berurutan.
  - Harus mengingat prosedur mana yang sudah dipanggil dan apa yang sudah diubah.
2. Pemrograman Fungsional
  - Berdasarkan teori fungsi matematika
  - Fungsi merupakan dasar utama program.
3. Pemrograman Terstruktur
  - Secara berurutan dan terstruktur.
  - Program dapat dibagi-bagi menjadi prosedur dan fungsi.
  - Contoh: PASCAL dan C
4. Pemrograman Modular
  - Pemrograman ini membentuk banyak modul.
  - Modul merupakan kumpulan dari prosedur dan fungsi yang berdiri sendiri
  - Sebuah program dapat merupakan kumpulan modul-modul.
  - Contoh: MODULA-2 atau ADA
5. Pemrograman Berorientasi Obyek
  - Pemrograman berdasarkan prinsip obyek, dimana obyek memiliki data/variabel/property dan method/event/prosedur yang dapat dimanipulasi

➤ Contoh: C++, Object Pascal, dan Java.

#### 6. Pemrograman Berorientasi Fungsi

➤ Pemrograman ini berfokus pada suatu fungsi tertentu saja. Sangat tergantung pada tujuan pembuatan bahasa pemrograman ini.

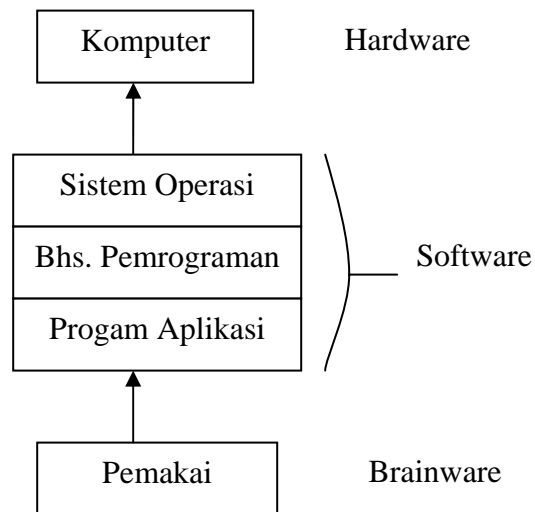
➤ Contoh: SQL (Structured Query Language), HTML, XML dan lain-lain.

#### 7. Pemrograman Deklaratif

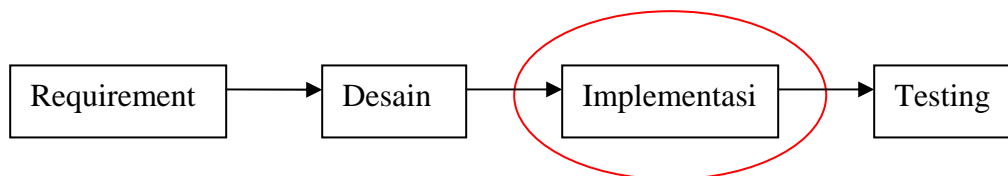
➤ Pemrograman ini mendeskripsikan suatu masalah dengan pernyataan daripada memecahkan masalah dengan implementasi algoritma.

➤ Contoh: PROLOG

### Struktur Sistem Komputer



### Siklus Hidup Perangkat Lunak (Software)



Dilihat dari Struktur Sistem Komputer dan Siklus diatas, Algoritma Pemrograman dan Struktur Data menempati posisi dibagian software dan di bagian implementasi karena bagian implementasi merupakan bagian dimana pemrogram melakukan proses coding (pembuatan program).

## Algoritma

Asal kata Algoritma berasal dari nama Abu Ja'far Mohammed Ibn Musa al-Khowarizmi, ilmuwan Persia yang menulis kitab al jabr w'al-muqabala (rules of restoration and reduction) sekitar tahun 825 M

### Definisi Algoritma

- Algoritma adalah urutan langkah logis tertentu untuk memecahkan suatu masalah. Yang ditekankan adalah urutan langkah logis, yang berarti algoritma harus mengikuti suatu urutan tertentu, tidak boleh melompat-lompat. (*Dari Microsoft Press Computer and Internet Dictionary 1997, 1998*)
- Alur pemikiran dalam menyelesaikan suatu pekerjaan yang dituangkan secara tertulis. Yang ditekankan pertama adalah alur pikiran, sehingga algoritma seseorang dapat juga berbeda dari algoritma orang lain. Sedangkan penekanan kedua adalah tertulis, yang artinya dapat berupa kalimat, gambar, atau tabel tertentu. (*Dari Algoritma dan Struktur Data dengan C, C++, dan Java oleh Moh Sjukani hal 1*)

Contoh Algoritma dalam kehidupan nyata:

- Jika seorang ingin memasak atau membuat kue, baik itu melihat resep ataupun tidak pasti akan melakukan suatu langkah-langkah tertentu sehingga masakannya atau kuenya jadi.
- Jika seseorang ingin mengirim surat kepada kenalannya di tempat lain, langkah yang harus dilakukan adalah:
  - Menulis surat
  - Surat dimasukkan ke dalam amplop tertutup
  - Amplop ditemplei perangko secukupnya.
  - Pergi ke Kantor Pos terdekat untuk mengirimkannya.

Dalam bidang komputer, algoritma sangat diperlukan dalam menyelesaikan berbagai masalah pemrograman, terutama dalam komputasi numeris. Tanpa algoritma yang dirancang baik maka proses pemrograman akan menjadi salah, rusak, atau lambat dan tidak efisien.

Pelaksana algoritma adalah *Komputer*.

Manusia dan komputer berkomunikasi dengan cara: manusia memberikan perintah-perintah kepada komputer berupa instruksi-instruksi yang disebut *program*.

Alat yang digunakan untuk membuat program tersebut adalah bahasa pemrograman. Bahasa pemrograman sangat bermacam-macam: C, C++, Pascal, Java, C#, Basic, Perl, PHP, ASP, JSP, J#, J++ dan masih banyak bahasa lainnya. Dari berbagai bahasa pemrograman cara memberikan instruksinya berbeda-beda namun bertujuan menghasilkan output yang sama.

### **Kriteria Algoritma Menurut Donald E. Knuth**

1. Input: algoritma dapat memiliki nol atau lebih inputan dari luar.
2. Output: algoritma harus memiliki minimal satu buah output keluaran.
3. *Definiteness* (pasti): algoritma memiliki instruksi-instruksi yang jelas dan tidak ambigu.
4. *Finiteness* (ada batas): algoritma harus memiliki titik berhenti (stopping role).
5. *Effectiveness* (tepat dan efisien): algoritma sebisa mungkin harus dapat dilaksanakan dan efektif. Contoh instruksi yang tidak efektif adalah:  $A = A + 0$  atau  $A = A * 1$

Namun ada beberapa program yang memang dirancang untuk untermintable: contoh Sistem Operasi

### **Jenis Proses Algoritma**

1. Sequence Process: instruksi dikerjakan secara sekuensial, berurutan.
2. Selection Process: instruksi dikerjakan jika memenuhi kriteria tertentu
3. Iteration Process: instruksi dikerjakan selama memenuhi suatu kondisi tertentu.
4. Concurrent Process: beberapa instruksi dikerjakan secara bersama.

### **Contoh Algoritma**

Algoritma menghitung luas persegi panjang:

1. Masukkan panjang (P)
2. Masukkan lebar (L)
3.  $L \leftarrow P * L$
4. Tulis L

Dalam Algoritma, tidak dipakai simbol-simbol / sintaks dari suatu bahasa pemrograman tertentu, melainkan bersifat umum dan tidak tergantung pada suatu bahasa pemrograman apapun juga. Notasi-notasi algoritma dapat digunakan untuk seluruh bahasa pemrograman manapun.

### **Definisi *Pseudo-code***

Kode atau tanda yang menyerupai (pseudo) atau merupakan penjelasan cara menyelesaikan suatu masalah. Pseudo-code sering digunakan oleh manusia untuk menuliskan algoritma.

Problem: mencari bilangan terbesar dari dua bilangan yang diinputkan

Contoh Pseudo-code:

1. Masukkan bilangan pertama
2. Masukkan bilangan kedua
3. Jika bilangan pertama > bilangan kedua maka kerjakan langkah 4, jika tidak, kerjakan langkah 5.
4. Tampilkan bilangan pertama
5. Tampilkan bilangan kedua

Contoh Algoritma

1. Masukkan bilangan pertama (a)
2. Masukkan bilangan kedua (b)
3. if a > b then kerjakan langkah 4
4. print a
5. print b

Contoh Lain Algoritma dan Pseudo-code:

<b>Pseudo-code</b>	<b>Algoritma</b>
Nilai A ditambah dengan 5	$A \leftarrow A + 5$
Cetak nilai A bila lebih besar dari 10	IF A > 10 THEN PRINT A
Dari dua bilangan A dan B, cari bilangan yang terbesar	IF A > B THEN PRINT A ELSE PRINT B

## Langkah-langkah dalam pemrograman komputer

### 1. Mendefinisikan masalah

Ini merupakan langkah pertama yang sering dilupakan orang. Menurut hukum Murphy (oleh Henry Ledgard):

“Semakin cepat menulis program, akan semakin lama kita dapat menyelesaikannya”.

Hal tersebut berlaku untuk permasalahan yang kompleks. Tentukan masalahnya, apa saja yang harus dipecahkan dengan menggunakan komputer, dan apa inputan serta outputnya.

### 2. Menemukan solusi

Setelah masalah didefinisikan, maka langkah berikutnya adalah menentukan solusi. Jika masalah terlalu kompleks, maka ada baiknya masalah tersebut dipecah menjadi modul-modul kecil agar lebih mudah diselesaikan.

Contohnya masalah invers matriks, maka kita dapat membagi menjadi beberapa modul:

- meminta masukkan berupa matriks bujur sangkar
- mencari invers matriks
- menampilkan hasil kepada pengguna

Dengan penggunaan modul tersebut program utama akan menjadi lebih singkat dan mudah dilihat.

### 3. Memilih algoritma

Pilihlah algoritma yang benar-benar sesuai dan efisien untuk permasalahan tersebut

### 4. Menulis program

Pilihlah bahasa yang mudah dipelajari, mudah digunakan, dan lebih baik lagi jika sudah dikuasai, memiliki tingkat kompatibilitas tinggi dengan perangkat keras dan platform lainnya.

### 5. Menguji program

Setelah program jadi, silahkan uji program tersebut dengan segala macam kemungkinan yang ada, termasuk error-handlingnya sehingga program tersebut akan benar-benar handal dan layak digunakan.

### 6. Menulis dokumentasi

Menulis dokumentasi sangat penting agar pada suatu saat jika kita akan melakukan perubahan atau membaca source code yang sudah kita tulis dapat kita ingat-ingat lagi dan kita akan mudah membacanya. Caranya adalah dengan menuliskan komentar-

komentar kecil tentang apa maksud kode tersebut, untuk apa, variabel apa saja yang digunakan, untuk apa, dan parameter-parameter yang ada pada suatu prosedur dan fungsi.

#### 7. Merawat program

Program yang sudah jadi perlu dirawat untuk mencegah munculnya bug yang sebelumnya tidak terdeteksi. Atau mungkin juga pengguna membutuhkan fasilitas baru yang dulu tidak ada.

### SOAL ALGORITMA

1. Buatlah algoritma untuk menghitung konversi suhu.dari Celcius menjadi Reamur dan Fahrenheit.

Input: suhu dalam Celcius

Proses:  $R = 4/5 * C$  dan  $F = 9/5 * C + 32$

Output: suhu dalam Reamur dan Fahrenheit

2. Buatlah algoritma untuk mencari sisi miring dari suatu segitiga siku-siku, jika diketahui panjang sisi yang membentuk sudut siku-siku.

Input: a dan b, yaitu panjang sisi pembentuk sudut siku-siku

Proses:  $c = \sqrt{a^2 + b^2}$

Ouput: sisi miring (c)

3. Buatlah algoritma untuk menentukan suatu bilangan genap atau ganjil

Input: suatu bilangan

Ouput: genap / ganjil

4. Buatlah algoritma untuk menentukan suatu bilangan adalah bilangan prima atau bukan.

5. Buatlah algoritma untuk untuk menghitung akar-akar persamaan kuadrat dengan rumus  $D = B^2 - 4 * A * C$

Jika  $D < 0$  maka didapat akar imajiner

Jika  $D = 0$  maka  $X1 = X2$  yang didapat dari  $D = -B / (2 * A)$

Jika  $D > 0$  maka ada dua akar  $X1 = -B + \sqrt{D/2 * A}$  dan  $X2 = -B - \sqrt{D/2 * A}$

### NEXT

*Generasi Bahasa Pemrograman, Proses Kompilasi, FlowChart, mengenal C++ dan C#*