

# **NOTASI ALGORITMIK**

# **NOTASI ALGORITMIK**

## **Pseudo-Code**

Notasi yang baik adalah notsi yang mudah dibaca dan dipahami, sehingga bisa ditranslasikan ke dalam notasi bahasa pemrograman. Notasi algoritmik ini merupakan *pseudo-code* yang mempunyai keterkaitan dengan notasi bahsa pemrograman, sehingga dapat memeprmudah proses penerjemahan dari *pseudo-code* ke dalam kode program.

Contoh : Sebuah pernyataan dalam notasi deskriptif ‘tulis nilai X dan Y’ , maka *pseudo-code* untuk pernytaan tersebut adalah

write(x,y)

Notasi write berarti nilai dicetak sebagai keluaran ke piranti keluaran.

## **Struktur Teks**

Hampir semua pembelajaran algoritma dengan menggunakan bahasa pemrograman apapun, algoritmaa pertama adalah mencetak tulisan :

 Hello world

Program “*Hello, world*” pertama kali dikemukakan oleh Brian W. Kerninghan, seorang penemu Bahasa C.

Algoritma untuk menulis “*Hello, world*” :

PROGRAM HelloWorld

*{ Program untuk mencetak “Hello world”*

 *Masukan :-*

 *Keluaran : string “Hello word”*

*}*

*DEKLARASI*

*{ tidak ada }*

ALGORITMA :

Write (“Hello world”)

## **Bagian Judul**

Judul adalah bagian penamaan program dan penjelasan (spesifikasi) dari program tersebut. Judul dibuat dengan mengawali kata kunci PROGRAM dan nama program. Kata PROGRAM ini sebagai pernyataan bahwa kita sedang menulis algoritma sebuah program pemecahan masalah.

Nama program sebaiknya ditulis singkat, tetapi dapat mewakili apa yang dilakukan oleh program. Contoh :

PROGRAM HelloWorld

*{ Program untuk mencetak “Hello world”*

 *Masukan program ini tidak ada, Keluarannya adalah tulisan*

 *‘Hello word’ tercetak dilayar.*

*}*

## **Bagian Deklarasi**

Deklarasi digunakan untuk memberitahu semua nama yang akan digunakan dalam algoritma beserta propertinya (tipe data). Nama tersebut dapat berupa nama kontanta, nama peubah, nama tipe, nama prosedur dan nama fungsi. Semua nama dalam algortima harus dikenali sebelum semuanya digunakan. Contoh :

DEKLARASI

 *{ nama konstanta }*

const Npeg = 100 *{ jumlah pegawai }*

const phi = 3.14 *{ nilai π }*

 *{ nama tipe }*

type Titik : record *{ koordinat titik di bidang kartesian }*

< x:integer, *{ absis }*

y:integer > *{ ordinat }*

 *{ nama peubah (variabel) }*

c : char *{ karakter yang dibaca }*

Q : Titik *{ titik dalam koordinat kartesian }*

ketemu : boolean *{ keadaan hasil pencarian }*

 *FUNCTION* Apakah\_A (input c, char) boolean

 *{ Mengembalikan nilai true bila c adalah karakter ‘A’, atau*

 *false bila sebaliknya }*

*PROCEDURE* Tukar (input/output A : integer, input/output B :

 integer)

 *{ Mempertukarkan nlai A dan B. Paramater A dan B sudah*

 *Terdefinisi nilainya. Setelah pertukaran, A berisi nilai B*

 *Semula, B berisi nilai A semula. }*

Oleh karena sebuah penamaan atau sebuah deklarasi adalah suatu kesatuan yang leksikal, maka nama tidak boleh mengandung spasi, tanda bacadan operator. Kecuali karakter garis bawah (“\_”) dapat digunakan untuk mengganti spasi. Selain itu, untuk bahasa pemrograman Pascal tidak membedakan huruf besar dan huruf kecil. Namun perbedaan huruf bisa dianggap sangat berbeda karena merupakan dua karakter yang berbeda untuk bahasa pemrograman yang *case sensitive*.

## **Bagian Algoritma**

Bagian algoritma ini adalah bagian inti dari sebuah algoritma, karena didalam bagian ini berisikan intruksi-intruksi pemecahan masalah dalam notasi *pseudo-code*. Program HelloWorld mengnandung bagian algoritma yang hanya berisi satu baris instruksi saja :

ALGORITMA :

Write (“Hello world”)

## **Implementasi**

Teks algoritma merupakan merupakan penggambaran untuk penyelesaian masalah agar dapat dieksekusi oleh komputer. Algoritma yang akan dieksekusi harus diterjemahkan dulu ke dalam notasi bahsa pemrograman yang biasanya disebut **translasi**.

**ALGORITMIK :** Program konversi suhu Fahrenheit-Celcius

PROGRAM FahrenheitCelcius

*{ Program untuk mencetak tabel Fahrenheit-Celcius dari x sampai*

*Y dengan kenaikan sebesar step. Masukan program ini adalah*

*suhu awal, suhu akhir. Keluarannya adalah tabel konversi suhu*

*dalam C dan F. }*

DEKLARASI

 F, C : real

 X, y, step : integer

ALGORITMA :

 read(x, y, step)

 F x

 while F ≤ y do

 C = 5/9 \* (F - 32)

 write(F, C)

 F F + step

 endwhile

**PASCAL :** Program konversi suhu Fahrenheit-Celcius

**PROGRAM** FahrenheitCelcius

*{ Program untuk mencetak tabel Fahrenheit-Celcius dari x sampai*

*Y dengan kenaikan sebesar step. Masukan program ini adalah*

*suhu awal, suhu akhir. Keluarannya adalah tabel konversi suhu*

*dalam C dan F. }*

{\* DEKLARASI \*}

**var**

 F, C : **real;**

 X, y, step : **integer;**

**begin**

{\* ALGORITMA : \*}

 read(x) :

 read(y) :

 read(step) :

 F := x;

 **while** F <= y **do**

 C := 5/9 \* (F - 32);

 **writeln**(F, ‘ ‘, C);

 F := F + step;

 **end;**

**end.**