

PERTEMUAN 2
NOTASI ALGORITMIK

Algoritma dan Pemrograman I

Teknik Informatika

UNIKOM



Algoritma disusun berdasarkan 3 bagian, antara lain :

1. Judul Algoritma
2. Kamus / Deklarasi
3. Algoritma / Deskripsi

Format Syntax Algoritma

JUDUL ALGORITMA

Nama Algoritma

DEKLARASI/ KAMUS {Lokal / Umum} ALGORITMA

```
Type namatipe  : tipe [subrange]
Type namatipe  : array [min..maks] of tipe
Namavar        : tipe
Namavar        : array [ min..maks] of tipe
Const nama     = nilai
Procedure namaproc(daftar_nama_parameter:tipe)
```

TUBUH ALGORITMA/ DESKRIPSI

Notasi Assigment
Notasi Kondisional/Pemilihan
Notasi Pengulangan
Notasi Pemanggilan

URAIAN ALGORITMA

Kumpulan algoritma masing-masing Procedure ataupun Function yang dipanggil dari Tubuh Algoritma

Algoritma untuk menulis Hello world:

Algoritma Hello_world
{ program untuk mencetak "Hello world"}

Kamus:
{tidak ada}

Algoritma:
Output("Hello world")

Algoritma untuk menghitung luas segitiga:

LuasSegi3
{algoritma untuk menghitung luas segitiga dengan diketahui alas dan tingginya}
KAMUS:
panjang,lebar : integer
luas : real
ALGORITMA:
panjang ← 10
lebar ← 5
luas ← 0.5*alas * tinggi
output("Luas Segitiga : ",luas)

Translasi Teks Algoritma ke dalam Teks Program Bahasa C++

Translasi Notasi Kamus

No	Algoritma	C++
1		#include <nama_unit>
2	CONST namaconst = nilai	#define namaconst nilai
3	TYPE namatipe : tipedata	Typedef tipedata namatipe;
4	namavar : tipedata namavar : namatipe	tipedata namavar; namatipe namavar;

Contoh :

1		#include <iostream.h>
2	CONST phi = 3.14	#define phi 3.14
3	TYPE jumlah : integer	Typedef int jumlah;
4	n : integer n : jumlah	int n; jumlah n;

Translasi Notasi Pengendalian (lanjutan)

No	Algoritma	C++

5	IF (kondisi) THEN
	aksi1
	ELSE
	aksi2
	ENDIF
		If (kondisi)
		aksi1;
		else
		aksi2;
		
		atau
		
		If (kondisi)
		{
		aksi1;
		}
		else
		{
		aksi2;
		}

Translasi Notasi Pengendalian (lanjutan)

	Algoritma	C++

6	DEPEND ON <ekspresi>
	<ekspresi 1> : aksi_1
	<ekspresi 2> : aksi_2
	:
	<ekspresi n> : aksi_n
	ENDDEPEND
	atau
	CASE namavarcase OF
	expkonstan 1 : aksi_1
	expkonstan 2 : aksi_2
	:
	expkonstan n : aksi_n
	ELSE
	aksi_lain
	ENDCASE
		Switch (ekspresi)
		{
		case nilai1:
		aksi_1;
		break;
		case nilai2:
		aksi_2;
		break;
		:
		case nilain:
		aksi_n;
		break;
		default: aksi_lain;
		}

Translasi Notasi Pengendalian (lanjutan)

	Algoritma	C++

7	[inisialisasi]
	WHILE (*kondisi_ulang*) DO
	daftar_aksi
	{ada aksi thd var kondisi}
	ENDWHILE
		[inisialisasi]
		while (*kondisi_ulang*)
		{
		daftar_aksi;
		/*ada aksi thd var kondisi*/
		}

8	[inisialisasi] REPEAT *daftar_aksi* {ada aksi thd var kondisi} UNTIL (*kondisi_stop*)	[inisialisasi] do { *daftar_aksi;* /*ada aksi thd var kondisi*/ }while (*kondisi_ulang*);

Translasi Notasi Pengendalian (lanjutan)

	Algoritma

9	namavar TRAVERSAL [awal..akhir]
	daftar_aksi;
	
	atau
	
	FOR var ← awal TO/DOWNTO akhir STEP counter DO
	daftar_aksi
	ENDFOR

	C++

	for(awal;kondisiulang;step)
	{
	daftar_aksi;
	}

Contoh Algoritma

Algoritma Luas_Lingkaran
{menghitung luas lingkaran diketahui jari-jarinya}

Kamus:
Const
phi = 3.14
r : integer
Luas : real

Algoritma:
r ← 5
Luas ← phi * r * r
output(”Jari –jari = ”,r);
output(”Luas lingkaran = ”,Luas);

Contoh program C++

```
#include <iostream.h>
#define phi 3.14

main ()
{
   int r;
  float Luas;
  r = 5;

   Luas = phi * r * r;

  cout << "Jari-jari : " << r;
  cout << "\nLuas lingkaran : " << Luas;
}
```