

{Algoritma Pemrograman 1}

```
#include<stdio.h>
int main(){
    int belajar;
    printf("Belajar kah hari ini ?\n");
    printf("1. ya\n");
    printf("2. tidak\n");
    scanf("%d",&belajar);

    if(belajar==1){
        printf("Pasti bisa !\n");
    }else
    if(belajar==2){
        printf("Harus belajar :)\n");
    }
    return 0;
}
```

PENDAHULUAN

Algoritma, Bahasa C, Compiler, komentar, Variabel, Tipe Data, Operator, IPO.

Menurut *Microsoft Press Computer and Internet Dictionary* 1997, 1998 , Algoritma adalah urutan langkah logis tertentu untuk memecahkan suatu masalah. Menurut **Abu Ja'far Muhammad Ibnu Musa Al Khawarizmi**, Algoritma adalah suatu metode khusus untuk menyelesaikan suatu persoalan.

A. Sifat-sifat Algoritma

Algoritma memiliki sifat sebagai berikut:

- Algoritma harus mengikuti suatu urutan tertentu dan tidak boleh melompat-lompat.
- Algoritma seseorang dan orang lainnya bisa saja berbeda, karena pada dasarnya setiap orang memiliki alur berpikir yang berbeda.
- Algoritma dapat berupa tabel, gambar, atau kalimat tertentu.

B. Bahasa Pemrograman dan Compiler

Bahasa pemrograman merupakan bahasa yang digunakan untuk menulis instruksi yang dimengerti oleh komputer. Bahasa yang akan kita gunakan di praktikum Algoritma Pemrograman ini adalah bahasa C/C++.

Compiler merupakan program yang digunakan untuk menerjemahkan perintah-perintah dalam bahasa pemrograman ke dalam Bahasa mesin. Salah satu compiler C/C++ yang akan kita gunakan adalah DevC++.

C. Karakteristik Bahasa C:

- C adalah salah satu Bahasa pemrograman yang terstruktur.
- Setiap program C mempunyai satu fungsi utama dengan nama "main" (program utama).
- Program akan dieksekusi dimulai dari statement pertama pada fungsi "main" tsb.
- Huruf besar dengan huruf kecil diartikan berbeda (case sensitive).
- Setiap statement diakhiri dengan semi-colon (titikkoma(;)).

```
#include<stdio.h> //Memanggil file header
/*
NAMA      : RADEN RIZKY IRIANTI PRAYUDI
JURUSAN   : PENDIDIKAN ILMU KOMPUTER UPI  |--> Komentar
*/
int main(){ //Fungsi Utama
    int a; //Tipe Data dan Variabel
    int b,c;
    c=b+a; //Operator
    printf("Hello World\n"); //Output
    system("pause"); //fungsi membekukan program
    return 0; //mengembalikan nilai
}
```

a. Header File

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
```

Nama file yang digunakan dalam `#include` seperti `conio.h` dan `stdio.h`, disebut sebagai header file karena ditempatkan dipaling atas program. Extension H berarti header. Dalam file header ini, terdapat fungsi atau prototipe yang bisa digunakan dalam program. Sebuah file header memiliki lebih dari 1 fungsi atau variabel global.

Khusus untuk header ini tidak dipergunakan tanda semicolon ";". Header tidak menggunakan tanda semicolon karena baris ini bukanlah suatu bentuk pernyataan. Baris ini adalah baris yang menginstruksikan kepada kompiler untuk menyisipkan file lain yang berisi deklarasi.

File header `stdio.h` digunakan untuk penanganan input / output standar seperti penulisan ke layar, ke file atau pembacaan data dari keyboard atau file. File header `conio.h` digunakan untuk penanganan ke layar seperti pengaturan warna, waktu jeda (delay), suara internal. Masih banyak file header standar selain `stdio.h` dan `conio.h`.

b. Komentar

Komentar adalah statement yang ditulis dalam Bahasa pemrograman yang tidak akan diterjemahkan ke dalam Bahasa mesin (karena memang bukan untuk dimengerti oleh mesin, melainkan untuk programmer itu sendiri). Fungsi komentar adalah untuk memberikan penjelasan tambahan tentang statement yang ada dalam Bahasa pemrograman sehingga programmer yang lain dapat dengan mudah memahami program kita. Cara penulisan komentar:

`//ini tata cara penulisan komentar jika hanya satu baris`

`/*Ini tata cara penulisan komentar jika lebih dari satu baris.*/`

c. Variabel, Deklarasi, Inisialisasi

Variabel adalah *identifier* yang digunakan untuk menampung data atau informasi.

Variabel bisa dideklarasikan dengan format:

Tipe_Data Nama_Variabel;	→Deklarasi biasa.	Int e;
Tipe_Data Nama_Variabel = Data;	→Deklarasi dengan inisialisasi.	Int d=5;
Tipe_Data *Nama_Variabel;	→Deklarasi Pointer. (Materi Pointer akan di jelaskan di matakuliah lain)	Int *d;

PENDAHULUAN

```
int a; //deklarasi biasa
int b,c; //deklarasi biasa dgn 2 variabel
int d=5; //deklarasi sekaligus inisialisasi
```

Selain menggunakan variable, deklarasi pun bisa menggunakan konstanta.

Contoh:

```
Const float phi=3.14;
```

Perbedaan konstanta dengan variable adalah jika sudah diinisialisasi konstanta tidak bisa berubah.

Selain bisa juga menggunakan #define.

Contoh:

```
#define x=2.12
```

Keuntungan menggunakan #define dibandingkan yang lain adalah kecepatan kompilasinya. Karena sebelum kompilasi dilakukan compiler akan mencari #define karena derajatnya sama dengan header. Ini ada beberapa kesalahan dalam pendeklarasian Variabel.

Deklarasi	B/S	Alasan
int angka pertama;	S	Tidak boleh menggunakan spasi
int 123;	S	Tidak boleh diawali oleh angka
int 1234567890qwertyuiopasdfghjkl;	S	Tidak boleh lebih dari 32 karakter
int printf;	S	Tidak boleh menggunakan perintah yang sudah di deklarasikan dalam bahasa C
int n@m@!;	S	Tidak boleh menggunakan karakter (#,\$, dll)

Inisialisasi Variabel

Nama_Variabel=Data;	→Inisialisasi setelah deklarasi
Nama_Variabel1>Nama_Variabel2	→Inisialisasi dengan data variabel lain

```
int d=5; //Inisialisasi langsung saat deklarasi
int e; //
e=5; //Inisialisasi setelah deklarasi variabel
int f=d; //Inisialisasi dengan data variabel lain
```

d. Tipe Data

Tipe data ada beberapa jenis yaitu :

- ✓ Tipe data karakter

Tipe data karakter digunakan untuk menampung sebuah karakter.

PENDAHULUAN

Tipe Data		Memori	Format	JangkauanNilai
KARAKTER	Char	1 byte	%c	-128 s/d 127
	unsigned char	1 byte	%u	0 s/d 255

✓ Tipe data string

Tipe data string digunakan untuk menampung beberapa karakter-karakter(array). (Pembahasan lebih lanjut di Pematerian tentang Array)

Tipe Data		Memori	Format	JangkauanNilai
String	Char	1 byte	%s	-128 s/d 127

✓ Tipe data bilangan bulat.

Tipe Data		Memori	Format	JangkauanNilai
BILANGAN BULAT	Int	2 byte	%d	-32.768 s/d 32.767
	unsigned int	2 byte	%u	0 s/d 65.535
	unsigned long	4 byte	%lu	0 s/d 4.294.967.295
	Long	4 byte	%ld	-2.147.483.648 s/d 2.147.483.647

✓ Tipe data bilangan pecahan.

Tipe Data		Memori	Format	JangkauanNilai
BILANGAN PECAHAN	Float	4 byte	%f	$3.4 \times (10^{-38}) - 3.4 \times (10^{+38})$
	double	8 byte	%f	$1.7 \times (10^{-308}) - 1.7 \times (10^{+308})$
	long double	10 byte	%lf	$3.4 \times (10^{-4932}) - 1.1 \times (10^{+4932})$

- ✓ Tipe data bentukan.

Kita bisa membuat tipe data sendiri. Jika kita membutuhkan suatu tipe data yang baru yang tidak terdapat pada tipe data standar maka kita dapat membuat tipe data baru dengan menggunakan tipe data terstruktur.

```
#include<stdio.h>

typedef struct{
    int tgl;
    char bln[10];
    int thn;
}waktu;

int main(){
    waktu date;

    scanf("%d",&date.tgl);
    scanf("%s",&date.bln);
    scanf("%d",&date.thn);
    printf("%d - %s - %d\n",date.tgl,date.bln,date.thn);

    system("pause");
    return 0;
}
```

e. Operator

Operator adalah simbol yang digunakan untuk mengolah nilai pada operand dan menghasilkan suatu nilai baru.

a. Operator Aritmatika

Operator	Contoh	Arti
+	c=a+b	Variable c diisidariisi variable a ditambahisi variable b
-	c=a-b	Variable c diisi dari variable a dikurangiisi variable b
*	c=a*b	Variable c diisi dari variable a dikalidenganisi variable b
/	c=a/b	Variable c diisi dari variable a dibagiolehisi variable b
++	a++	Isi variable a ditambah 1. Perintahinisamadengan a=a+1 atau a+=1
--	a--	Isi variable a dikurang. Perintahinisamadengan a=a-1 atau a-=1
%	c=a % b	Variable c diisidarisipembagian variable a dibagi variable b

PENDAHULUAN

+=	c+=a	Variable c ditambah dengan isi variable a. Sama dengan c=c+a
/=	c/=a	Variable c dibagi dengan isi variable a. Sama dengan c=c/a
-=	c-=a	Variable c dikurang dengan isi variable a. Sama dengan c=c-a
=	c=a	Variable c dikali dengan isi variable a. Sama dengan c=c*a
%=	c%=a	Variable c diisidari sisa pembagian c dibagi isi variable a. Sama dengan c=c%a

b. Operator Logika

Operator	Contoh	Arti
&&	(a==b) && (c==d)	a sama dengan b dan c sama dengan d
 	(a==b) (c==d)	a sama dengan b atau c sama dengan d
!	!(a == !(c))	Negasi a sama dengan negasi c

c. Operator Relasi

Operator	Contoh	Arti
==	a==b	Isi variabel a sama dengan isi variabel b
!=	a!=b	Isi variabel a tidak sama dengan isi variabel b
>	a>b	Isi variabel a Lebih besar dari isi variabel b
<	a<b	Isi variabel a Lebih kecil dari isi variabel b
>=	a>=b	Isi variabel a Lebih besar atau sama dengan isi variabel b
<=	a<=b	Isi variabel a Lebih kecil atau sama dengan isi variabel b

d. Operator

Operator	Operasi
<<	Geser bit kekiri
>>	Geser bit kekanan
&	Dan (AND)
	Atau (OR)
^	XOR
~	NOT (Komplemen)

e. Statement Input, Proses, Output

a. Statement Input

Statement Input merupakan statement yang berhubungan dengan pengisian data oleh user. Bentuk umum penggunaan fungsi input data adalah:

```
scanf("format",&namavariabel);
```

Format	Kegunaan
%c	Digunakan untuk pemasukan data bertipe char
%d	Digunakan untuk pemasukan data bertipe int.
%u	Digunakan untuk pemasukan data berupa unsigned int atau unsigned char.
%f	Digunakan untuk pemasukan data berupa bilangan pecahan (float)
%s	Digunakan untuk pemasukan data berupa string.
%ld	Digunakan untuk pemasukan data bertipe long int.
%lf	Digunakan untuk pemasukan data bertipe long float.

b. Statement Proses

Adalah statement atau fungsi yang digunakan untuk mengolah masukan data dari keyboard. Contoh fungsi yang bisa digunakan adalah Fungsi operator.

c. Statement Output

Statement ini digunakan untuk mencetak informasi ke layar. Bentuk umum statement output:

```
printf("Informasi yang akan dicetak");
```

Sumber : Modul Tim Asisten Algoritma Pemrograman 2015