

Algoritme dan Pemrograman

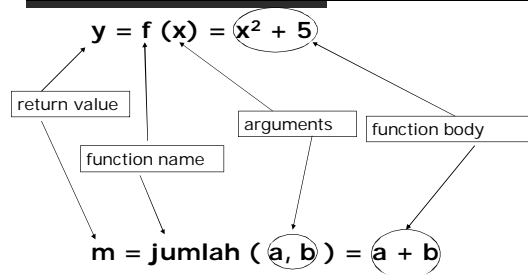
Kuliah #6
Fungsi

Fungsi

- Contoh fungsi $\rightarrow y=f(x)=x^2+5$.
- Nilai yang mungkin untuk variabel x disebut daerah fungsi (df), dan untuk variabel y atau $f(x)$ disebut wilayah fungsi (wf).
- Nilai yang diberikan pada df akan menentukan nilai yang dihasilkan pada wf.
- Pada contoh tersebut:
 - f disebut nama fungsi
 - x disebut argumen fungsi
 - x^2+5 disebut tubuh fungsi (function body)
 - Tipe data dari y disebut return type.

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Fungsi



DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Bentuk umum fungsi dalam C

```
return-type function-name parameter-list
{
    local definitions;
    statements;
    return value;
}
```

- Return type \rightarrow tipe dari hasil fungsi.
- Function name \rightarrow nama fungsi (berupa identifier).
- Parameter list \rightarrow daftar variabel, dituliskan di dalam tanda kurung $()$, dipisahkan oleh tanda koma $(,)$.

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Contoh : Fungsi menjumlah dua bilangan

```
#include <stdio.h>

int jumlah(int x, int y) { // Fungsi jumlah
    int t;
    t = x+y;
    return t;
}

main() { // Fungsi utama
    int a, b, c;
    scanf("%d %d", &a, &b);

    c=jumlah(a,b); // Memanggil fungsi
    printf( "%d\n", c);
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Function prototype

- Prototipe fungsi terdiri dari tiga elemen:
 - return type,
 - function name, dan
 - parameter list.
- Setiap fungsi harus dituliskan sebelum fungsi main, berupa fungsi lengkap atau hanya prototipe.
- Jika berupa prototipe, maka fungsi lengkapnya dituliskan di bawah fungsi main.

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Function prototype

```
#include <stdio.h>

int jumlah(int, int); // Function prototype

main() {
    int a, b, c;
    scanf("%d %d", &a, &b);
    c=jumlah(a,b);
    printf( "%d\n", c);
    return 0;
}

int jumlah(int x, int y) {
    int t;
    t = x+y;
    return t;
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Variabel lokal

- ❑ Variabel yang didefinisikan dan dideklarasikan di dalam suatu fungsi hanya berlaku di dalam fungsi tersebut.
- ❑ Disebut sebagai variabel lokal.
- ❑ Hal ini berarti bahwa variabel yang dibuat di dalam satu fungsi tidak ada hubungannya dengan variabel di fungsi lainnya walaupun menggunakan nama yang sama.

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Variabel lokal : Apa output program berikut?

5 10 15 30

```
#include <stdio.h>

int test(int a, int b) {
    int c;
    c=kali(a,b);
    return c;
}

int kali(int x, int y) {
    int z;
    z=x*y;
    return z;
}

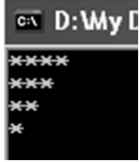
main() {
    int a=5, b=10, c=15, d;
    d = test(a,b);
    printf("%d %d %d %d\n", a, b, c, d);
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Void

- ❑ Jika fungsi tidak mengembalikan nilai apapun, maka return-type adalah VOID.
- ❑ Contoh (apa yang dilakukan?):

```
void pola(int n) {
    int i, j;
    for (i=1; i<=n; i++) {
        for (j=i; j<=n; j++)
            printf("*");
        printf("\n");
    }
}
```



DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Fungsi rekursif

- ❑ Perhatikan fungsi berikut:
jumlah(a,b) = a+b
- ❑ Jika b=0, maka jumlah(a,b)=a.
- ❑ Jika b≠0, maka
jumlah(a,b) = a+b
= 1+(a+(b-1))
= 1+jumlah(a,b-1)

$$jumlah(a,b) = \begin{cases} a & \text{utk } b = 0 \\ 1 + jumlah(a,b-1) & \text{utk } b \neq 0 \end{cases}$$

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Fungsi rekursif

- ❑ Untuk b≠0, fungsi jumlah masih memanggil fungsi yang sama (dirinya sendiri).
- ❑ Fungsi yang memanggil dirinya sendiri disebut fungsi rekursif.
- ❑ Program yang disusun menggunakan fungsi rekursif akan lebih ringkas, tetapi kecepatan program lebih lambat karena komputer membutuhkan waktu overhead menyusun dalam struktur stack.

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Fungsi rekursif: Contoh

```
int jumlah(int a, int b) {
    if (b==0)
        return a;
    else
        return 1+jumlah(a,b-1);
}
```

Telusuri fungsi tersebut jika dipanggil dengan jumlah(3,2)

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Elemen fungsi rekursif

Fungsi rekursif membutuhkan tiga elemen:

- *Terminating conditions* (kondisi berhenti), yaitu suatu kondisi untuk menentukan kapan proses rekursif akan dihentikan. Contoh: $b=0$.
- *Rekursif conditions* (kondisi rekursif), yaitu suatu kondisi dimana fungsi akan terus memanggil dirinya sendiri. Contoh: $b \neq 0$.
- *Recursive expressions* (ekspresi rekursif), yaitu ekspresi yang ditulis untuk memanggil fungsi itu sendiri. Contoh: $jumlah(a,b)=1+jumlah(a,b-1)$.

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Contoh lain: Fungsi faktorial

- Contoh: $5! = 5.4.3.2.1 = 120$

$$faktorial(n) = \begin{cases} 1 & \text{utk } n = 0 \\ n * faktorial(n-1) & \text{utk } n \neq 0 \end{cases}$$

- Implementasi dalam C:

```
long int faktorial(long int n) {
    if (n==0)
        return 1;
    else
        return n*faktorial(n-1);
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Latihan 1

Lengkapi program berikut dengan mendefinisikan fungsi `jumFaktor` untuk menghitung banyaknya faktor pembagi dari bilangan bulat n selain 1 dan bilangan itu sendiri. Sebagai contoh, jika diberi input 12 maka output program adalah 4 (ada 4 bilangan yang dapat membagi 12, yaitu 2, 3, 4, dan 6). Sedangkan jika diberi input 7 maka output program adalah 0.

```
#include <stdio.h>

main() {
    int n;
    scanf("%d", &n);
    printf("%d\n", jumFaktor(n));
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Latihan 2

Lengkapi program berikut dengan mendefinisikan fungsi `jumDigit` untuk menghitung jumlah dari setiap digit bilangan bulat. Sebagai contoh, jika diberi input 125 maka output program adalah 8 (dihitung dari $1+2+5$). Sedangkan jika diberi input 20 maka output program adalah 2.

```
#include <stdio.h>

main() {
    int n;
    scanf("%d", &n);
    printf("%d\n", jumDigit(n));
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR

Latihan 3

Lengkapi program berikut dengan mendefinisikan fungsi `segitiga` untuk mencetak bentuk segitiga berukuran n (bilangan bulat) seperti pada contoh. Jika diberi input 4 maka output program adalah:

```
xxxxx
xxxx
xx
x

#include <stdio.h>

main() {
    int n;
    scanf("%d", &n);
    segitiga(n);
    return 0;
}
```

DEPARTEMEN ILMU KOMPUTER
INSTITUT PERTANIAN BOGOR