

# Pemrograman Modular

**MK Dasar Algoritma dan Pemrograman**

# MK Dasar Algoritma dan Pemrograman

Kode Mata Kuliah : TKE60001  
Beban Studi : 3 sks  
Sifat : Wajib  
Tujuan :

- Memberikan pengertian tentang dasar algoritma dan memberikan kemampuan dan keterampilan melakukan pemecahan masalah secara sistematis kemudian meingimplementasikannya ke dalam bahasa pemrograman.

## Capaian

- Mampu menyusun algoritma dengan menggunakan diagram alir dan pseudocode untuk berbagai kasus.
- Mampu mengimplementasikan algoritma menggunakan Bahasa Pemrograman

## Pokok Bahasan :

- Pengenalan komputer;
- Pengertian algoritma;
- Algoritma-algoritma dasar: percabangan, perulangan, pengurutan (sorting), pencarian (searching), rekursi;
- Pengenalan Bahasa Pemrograman: Struktur bahasa pemrograman, Type data, Operator, **Fungsi dan Prosedur**, Array, User defined data type.;
- Konsep pemograman terstruktur;
- Pemrograman berukuran besar.

# Tools

- Installer program Dev C++ Version 4.9.9.2 Dapat didownload di:  
<https://sourceforge.net/projects/orwellddevcpp/>
- Flowchart bisa dibuat di <https://app.diagrams.net/>
- Materi reading, source code, dan ppt bisa diunduh di:  
<https://vlm2.ub.ac.id/course/view.php?id=xxx>
- Referensi buku :  
Bab 6, “Modular Programming”, Problem Solving and Program Design in C, Jeri R. Hanly dan Elliot B. Koffman, Addison Wesley, 2002

# Pertemuan sebelumnya

- Pendahuluan
- Algoritma dan Representasinya
- Komputer dan Pemrograman
- Pemrograman Bahasa C
- Perancangan Top-down Menggunakan Fungsi
- Struktur Percabangan
- Struktur Perulangan
- **Pemrograman Modular**
- ...

# Pemrograman Modular

## Tujuan

Memberikan pemahaman mengenai berbagai macam tipe fungsi berkaitan dengan argumen input dan “result value” serta bagaimana mengimplementasikannya dalam bahasa C.

## Indikator

Setelah mengikuti materi ini, mahasiswa diharapkan dapat mendesain program yang kompleks secara modular

## Metode Pembelajaran

Ceramah, diskusi, Problem based learning

## Waktu

5-10 menit

# Pemrograman Modular

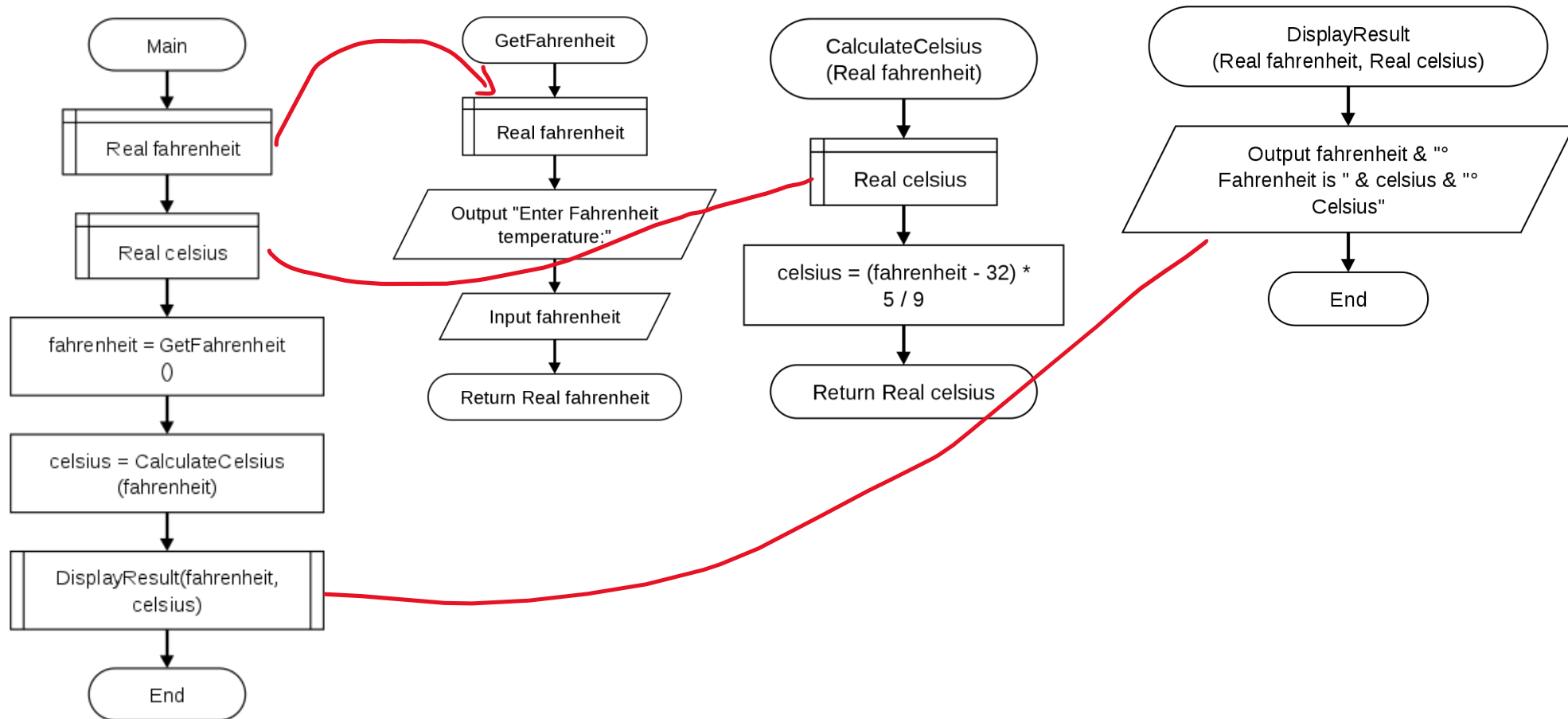
## Materi Pembelajaran:

1. Pemrograman dengan Fungsi
2. Jenis-jenis Fungsi
3. Pemanggilan Fungsi
4. Contoh pemrograman dengan Fungsi

# Pemrograman Modular -> Fungsi

- Fungsi (Function) merupakan blok dari kode yang dirancang untuk melaksanakan tugas khusus.
  - Mengurangi pengulangan penulisan program
  - Program menjadi terstruktur, sehingga mudah dipahami dan dikembangkan.

# Pemrograman Modular -> Fungsi





# Pemrograman Modular -> Fungsi

## Ketentuan

- Nama fungsi, boleh dituliskan secara bebas dengan ketentuan, tidak menggunakan spasi dan nama-nama fungsi yang mempunyai arti sendiri.
- Argumen/Parameter, diletakan diantara tanda kurung “( )” yang terletak dibelakang nama fungsi. Argumen boleh diisi dengan suatu data atau dibiarkan kosong.
- Pernyataan / perintah, diletakan diantara tanda kurung ‘{ }’.
- Pada pemanggilan sebuah fungsi, cukup dengan menuliskan nama fungsinya.

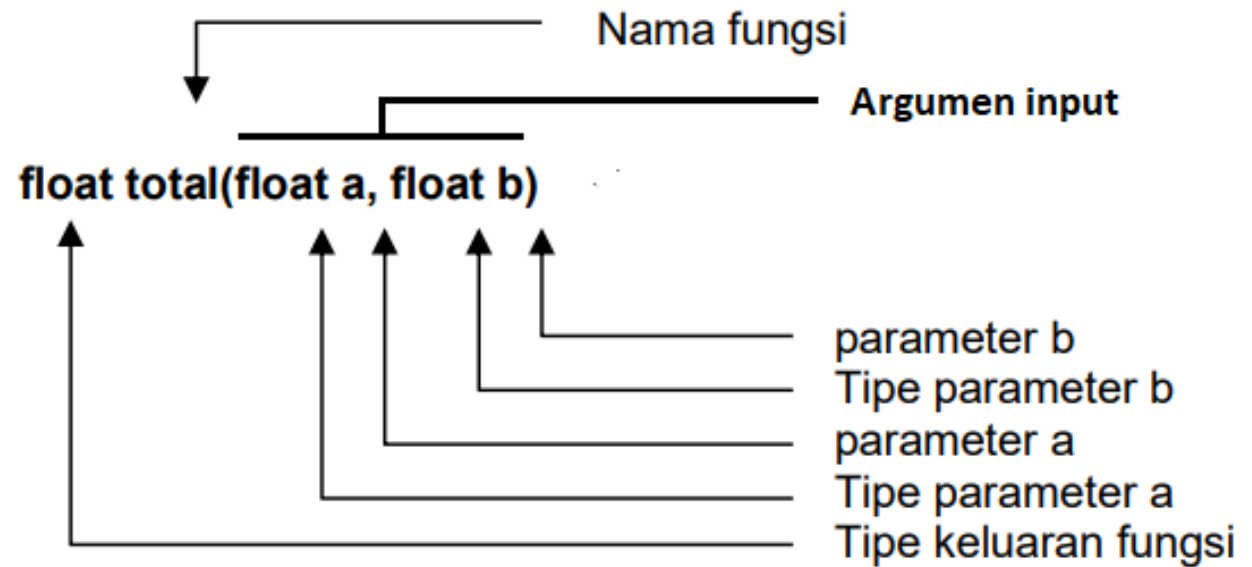


# Pemrograman Modular -> Fungsi

## Struktur Fungsi

```
float total(float a, float b)
{
    float jumlah;
    jumlah = a + b;
    return jumlah;
}
```

```
main()
{
    printf("Jumlah dari 2 dan 3 = %f\n", total(2,3));
}
```



# Pemrograman Modular -> Jenis-jenis Fungsi

## Struktur Fungsi

```
float total(float a, float b)
{
    float jumlah;
    jumlah = a + b;
    return jumlah;
}
```

Berdasarkan argumen input dan *result value*

- Fungsi tanpa *result value*, void (*procedure*)
- Fungsi dengan beberapa argumen input dan satu hasil/*return value*
- Fungsi dengan satu argumen input dan beberapa hasil
- Fungsi dengan parameter input/output



# Pemrograman Modular -> Pemanggilan Fungsi

```
main()
{
    printf("Jumlah dari 2 dan 3 = %f\n", total(2,3));
}
```

return jumlah;

## ❖ Argumen/Parameter :

- Memungkinkan terjadinya komunikasi antara “main function” dan fungsi-fungsi lain dalam program
- Membuat fungsi menjadi lebih serba guna karena daftar argumen memungkinkan fungsi untuk dapat memanipulasi data yang berbeda setiap kali fungsi tersebut dipanggil

## ❖ Return statement : mengembalikan satu “result value” dari suatu fungsi



# Pemrograman Modular -> Pemanggilan Fungsi

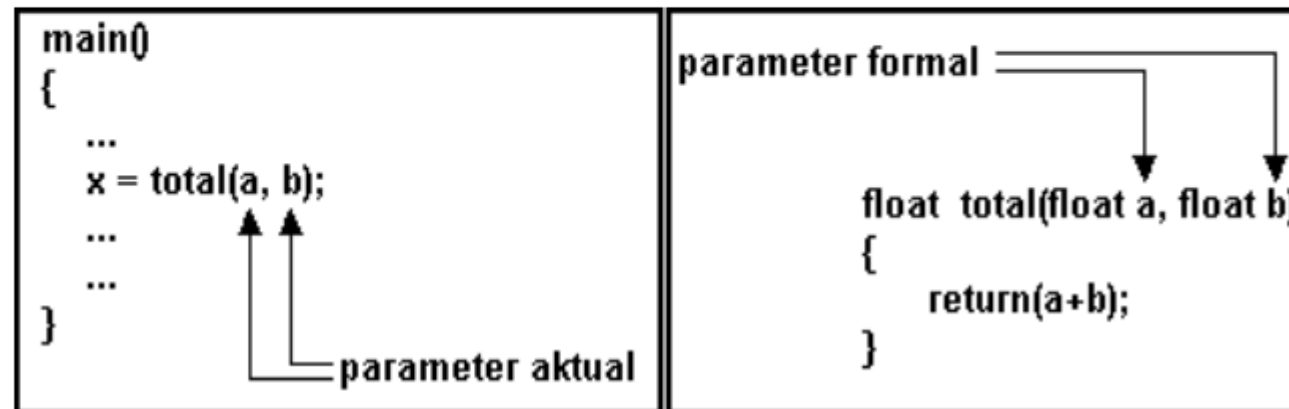
```
main()
{
    printf("Jumlah dari 2 dan 3 = %f\n", total(2,3));
}
```

```
return jumlah;
```

**Penjelasan** Terdapat dua macam parameter fungsi, yaitu :

- **Parameter formal** adalah variabel yang ada pada daftar parameter dalam definisi fungsi.
- **Parameter Aktual** adalah variabel yang dipakai dalam pemanggilan fungsi.

Bentuk penulisan Parameter Formal dan Parameter Aktual.



# Pemrograman Modular -> Pemanggilan Fungsi

## Pemanggilan Fungsi

Pemanggilan dengan nilai ( Call by Value )

Pemanggilan dengan Referensi (Call by Reference)

## Pengiriman Data Ke Fungsi

Pengiriman Data Konstanta

Pengiriman Data Variabel

# Contoh kasus

- Buatlah program untuk membandingkan manakah yang lebih luas dari segitiga jika diketahui Panjang ke-3 sisi-sisinya: (13,12,5); (7,7,7);

```
main
{
    int ls1 = luas_segitiga(13,12,5);
    int ls2 = luas_segitiga(7,7,7);
    printf("Luas segitiga %1", (ls1>ls2)?ls1;ls2);
}
```

- Hint: gunakan rumus Heron

$$\text{Luas} = \sqrt{s(s-a)(s-b)(s-c)}$$

# Pemrograman Modular

## Materi Pembelajaran:

1. Pemrograman dengan Fungsi; reusable, simplify
2. Jenis-jenis Fungsi; procedure,
3. Pemanggilan Fungsi, call by reference/value, pass a constant/variable
4. Contoh pemrograman dengan Fungsi