**FUNGSI**

* Bagian dari program yang mempunyai bentuk/struktur sama dengan fungsi main
* Terdiri dari tipe fungsi, nama fungsi, parameter/argumen fungsi dan isi fungsi

Deklarasi fungsi

Tipe\_fungsi Nama\_fungsi(argumen/parameter fungsi)

{ // awal fungsi

Isi\_fungsi  
} // akhir fungsi

Contoh :

void Cetak(int Bil)

{

for(int I=1 ; I <=Bil ; I++)

{

cout<<”Pemrograman Dasar” ; // Isi Fungsi

}

}

* Dieksekusi dengan cara memanggil nama fungsi (juga ditentukan oleh tipe fungsi , parameter fungsi)
* Pemanggilan sebuah fungsi menyebabkan eksekusi program berpindah ke fungsi yang dipanggil. Setelah selesai, eksekusi program kembali ke fungsi pemanggil (setelah akhir dari fungsi **‘}’** atau setelah ada pernyataan **return**)

**Contoh : contoh3.cpp**

* Pemanggilan sebuah fungsi bisa disertai dengan pengiriman nilai (melalui parameter/argumen fungsi)

**Contoh : contoh1.cpp, contoh1A.cpp, contoh1B.cpp,contoh1C.cpp**

* Pengiriman nilai tidak berdasarkan nama, tetapi berdasarkan posisi

**Contoh : contoh1D.cpp**

* Apabila sebuah fungsi selesai dieksekusi, maka fungsi tersebut bisa mengirimkan nilai balik ke fungsi pemanggil (**nilai balik / return value**)

Contoh : contoh2.cpp, contoh2A.cpp

* Fungsi yang dipanggil harus dituliskan di atas fungsi pemanggil, kecuali ada prototype fungsi

Contoh : contoh4.cpp,contoh4A.cpp

ALASAN

* Mendukung konsep pemecahan masalah secara modular
* Program menjadi lebih terstruktur
* File Header 🡪 class

TIPE FUNGSI

* Digunakan sebagai nilai balik fungsi (return value)
* Satu fungsi, hanya mempunyai satu nilai balik tipe fungsi
* Nilai balik tipe fungsi dituliskan dengan perintah **return**
* Tipe data nilai balik fungsi menggunakan tipe data yang ada di C++
* Fungsi void adalah fungsi yg tidak punya tipe (tdk punya nilai balik tipe fungsi)

**Contoh : contoh7.cpp**

Nilai Balik/return value

* Tipe fungsi (hanya satu untuk satu fungsi) 🡪 contoh2.cpp, contoh2A.cpp, Contoh7.cpp
* Argumen/parameter fungsi (bisa lebih dari satu) 🡪 contoh8.cpp

ARGUMEN/PARAMETER FUNGSI

* Pass by Value (tidak ada nilai balik parameter fungsi)

Penulisan parameter :

void Total (int Bil1, int Bil2, int Hasil)

Pengiriman parameter :

Total(Bil1,Bil2,Hasil)

Total(10,20,30);

* Pass by Reference (ada nilai balik parameter fungsi)

Penulisan parameter :

void Total (int Bil1, int Bil2, int &Hasil)

Pengiriman parameter :

Total(Bil1,Bil2,Hasil)

* Pass By Pointer (ada nilai balik parameter fungsi)

Penulisan parameter :

void Total (int Bil1, int Bil2, int \*Hasil)

Pengiriman parameter :

Total(Bil1,Bil2,&Hasil)

**Contoh : contoh8.cpp , contoh9.cpp contoh10.cpp**

PROTOTIPE FUNGSI

* Untuk mengenalkan fungsi (tipe fungsi, nama fungsi, argumen/parameter fungsi)

ARRAY SEBAGAI PARAMETER FUNGSI

* Otomatis mempunyai nilai balik parameter fungsi

LATIHAN

Buat program dengan menggunakan fungsi

1. Menentukan bilangan terbesar dari N buah data

**Lat1.cpp, Lat1A.cpp**

1. Mencetak N bilangan yang pertama dari deret bilangan kelipatan M

**Lat2.cpp**

1. Mencari FPB terbesar dari 2 buah bilangan

**Lat3.cpp , Lat3A.cpp**