

## PENGENALAN BAHASA C

### Paradigma Pemrograman

- Pemrograman Deklaratif/Fungsional
  - Persoalan diselesaikan dengan mengekspresikan logika persoalan
- Pemrograman Prosedural/Terstruktur
  - Persoalan diselesaikan dengan menentukan langkah per langkah
- Pemrograman Berorientasi Objek
  - Penyelesaian persoalan dilakukan dengan membagi menjadi objek-objek

### Pemrograman Prosedural/Terstruktur

- Pemrograman terstruktur merupakan bagian dari pemrograman prosedural
- Pemrograman prosedural memungkinkan seorang pemrogram mendefinisikan secara tepat langkah-langkah penyelesaian persoalan
- Fokusnya adalah membagi persoalan menjadi kumpulan variabel, struktur data, dan subrutin.

### Komponen Pemrograman Terstruktur

- Sequence (urutan) :
  - Eksekusi statement/instruksi secara terurut
- Selection (seleksi) :
  - Eksekusi salah satu statement bergantung pada kondisi tertentu
- Repetition (pengulangan) :
  - Eksekusi sebuah statement hingga mencapai kondisi tertentu

### Sejarah Bahasa C

- Akar dari bahasa C adalah bahasa BCPL-Basic Combine Programming Language (dikembangkan sejak th. 1967)
- Pada tahun 1970, Ken Thompson mengembangkan bahasa B.
- Bahasa C pertama kali digunakan pada komputer Digital Equipment Corporation (DEC) PDP-11 yang menggunakan s.o UNIX
- Standar Bahasa C yang asli adalah standar dari UNIX
- ANSI (American National Standards Institute) membentuk suatu komite (ANSI committee X3J11) pada tahun 1983 yang kemudian menetapkan standar ANSI untuk bahasa C.



## Karakteristik C

- Digolongkan bahasa tingkat menengah
- Lebih sulit dari bahasa Pascal dan Basic yang termasuk tingkat tinggi
- C mengkombinasikan bahasa tingkat tinggi dan tingkat rendah. Tujuannya untuk mengambil kelebihan dari masing-masing tingkat bahasa.
  - Tingkat tinggi : lebih mudah penulisan
  - Tingkat rendah : lebih cepat eksekusinya

## Kelebihan Bahasa C

- Tersedia hampir di semua jenis komputer
- Kodenya portabel yaitu dapat digunakan antar komputer dengan sedikit perubahan
- Struktur bahasanya baik, sehingga mudah dipelajari dan dipahami
- Dibanding bahasa mesin, C lebih mudah dipahami
- Dapat memanipulasi bit dan alamat pointer

## Kelemahan C

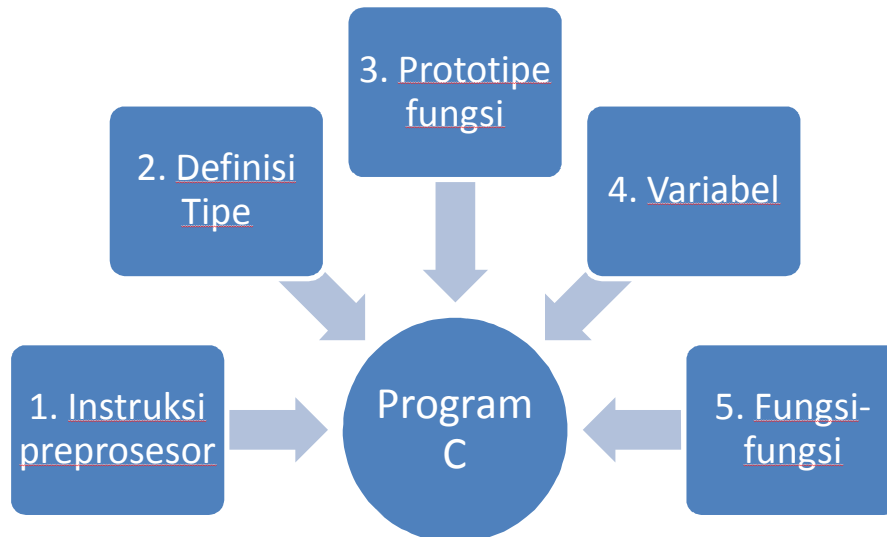
- Banyaknya operator serta fleksibilitas penulisan yang kadang membingungkan pengguna
- Pointer. Pemrogram pemula umumnya tidak terbiasa menggunakan pointer.

## Struktur Penulisan Program C

- Program C tersusun atas sejumlah fungsi.
- Paling tidak ada satu fungsi utama yaitu fungsi bernama **main**.
- Setiap fungsi tersusun atas beberapa statement/pernyataan.
- Penulisan setiap fungsi diawali dengan tanda '{' dan diakhiri dengan tanda '}'



## Kerangka Program C



### Preprocessor : **#define**

- Digunakan untuk mendefinisikan konstanta atau makro.
- Rumus :  
**#define <macro> <replacement name>**

Contoh :

```

#define TRUE 1
#define FALSE !TRUE
#define PI 3.14
  
```

### Preprocessor : **#include**

- **#include** merupakan salah satu pengarah praprosesor (preprocessor directive)
- Dipakai untuk membaca file yang diantaranya berisi deklarasi fungsi dan konstanta.
- File yang dibaca adalah file berekstensi **.h** (istilahnya file header)
- C menyediakan beberapa file header siap pakai, contoh `stdio.h`, `stdlib.h`, dll

### Contoh Penggunaan Directive

- **#include <stdio.h>**, artinya adalah memerintahkan pada kompiler untuk membaca file **stdio.h** pada saat kompilasi.
- Bentuk umum directive :
  - **#include <nama\_file\_header>**



## Pola Deklarasi Fungsi

```
tipe_data nama_fungsi (parameter){
    //letakkan variabel lokal di sini
    //letakkan pernyataan C di sini
}
```

## Fungsi main()

- Merupakan fungsi utama yang harus ada pada program C.
- Fungsi ini menjadi titik awal dan titik akhir eksekusi program.
- Biasanya ditempatkan di paling atas kode, untuk memudahkan penelusuran
- Sintaks :
 

```
void main ()
{
    /* tubuh fungsi */
}
```
- **void** menunjukkan bahwa fungsi **main** tidak mengembalikan nilai apapun

## Fungsi printf()

- Merupakan fungsi standar C untuk menampilkan suatu keluaran pada layar/monitor
- Bentuk umum :
  - printf(“string keluaran”, daftar argumen)
- String keluaran dapat berupa :
  - %d untuk menampilkan bil. bulat(integer)
  - %f untuk menampilkan bil. pecahan
  - %c untuk menampilkan karakter
  - %s untuk menampilkan string/teks
- Contoh :
 

```
#include <stdio.h>
void main(){
    printf(“Hello World”);
}
```
- Menampilkan tulisan “Hello World”
  - printf(“Hello World”);
- Penggunaan tanda ‘\’ :
  - \” menyatakan karakter petik ganda
  - \\ menyatakan karakter backslash
  - \t menyatakan karakter tab



## Komentar pada Program

- Digunakan untuk keperluan dokumentasi
- Diawali dengan tanda “/\*” dan diakhiri dengan tanda “\*/”
- Untuk komentar satu baris : “//”
- **//penggunaan directive**

```
#include <stdio.h>
void main(){
    /*mencetak ke layar*/
    printf(“Hello World”);
}
```

