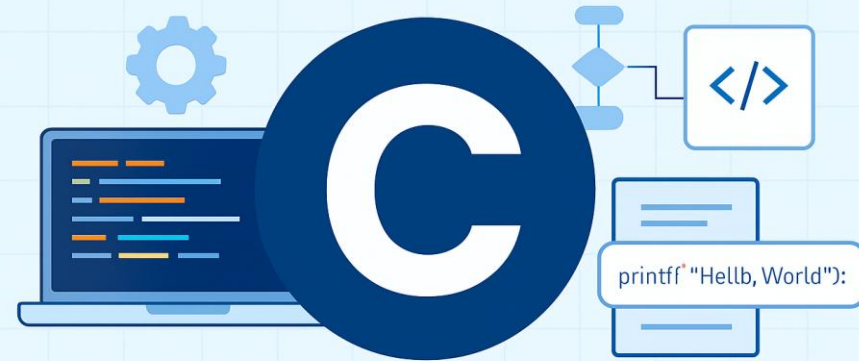


Konsep Pemrograman – Pertemuan 1

Pengenalan Bahasa C

Yunia Ikawati

PENGANTAR BAHASA C





Link Penting

1. Link Modul Pembelajaran :
 - <https://yunia.lecturer.pens.ac.id/>
2. Link Absensi dan Pengumpulan Tugas:
 - <https://ethol.pens.ac.id/>

Kontrak Perkuliahan - Evaluasi dan Penilaian

Kehadiran dan partisipasi aktif: 10%

Tugas dan kuis: 25%

Ujian Tengah Semester (UTS): 30%

Ujian Akhir Semester (UAS): 35%

Aturan dan Etika Akademik

- Mahasiswa wajib hadir minimal 75% dari total pertemuan
- Mahasiswa hadir tepat waktu, maksimal keterlambatan 15 menit
- Tugas dikumpulkan sesuai tenggat waktu yang ditentukan
- Plagiarisme tidak diperbolehkan dan akan dikenakan sanksi akademik
- Mahasiswa diharapkan menjaga etika dan sopan santun dalam proses pembelajaran

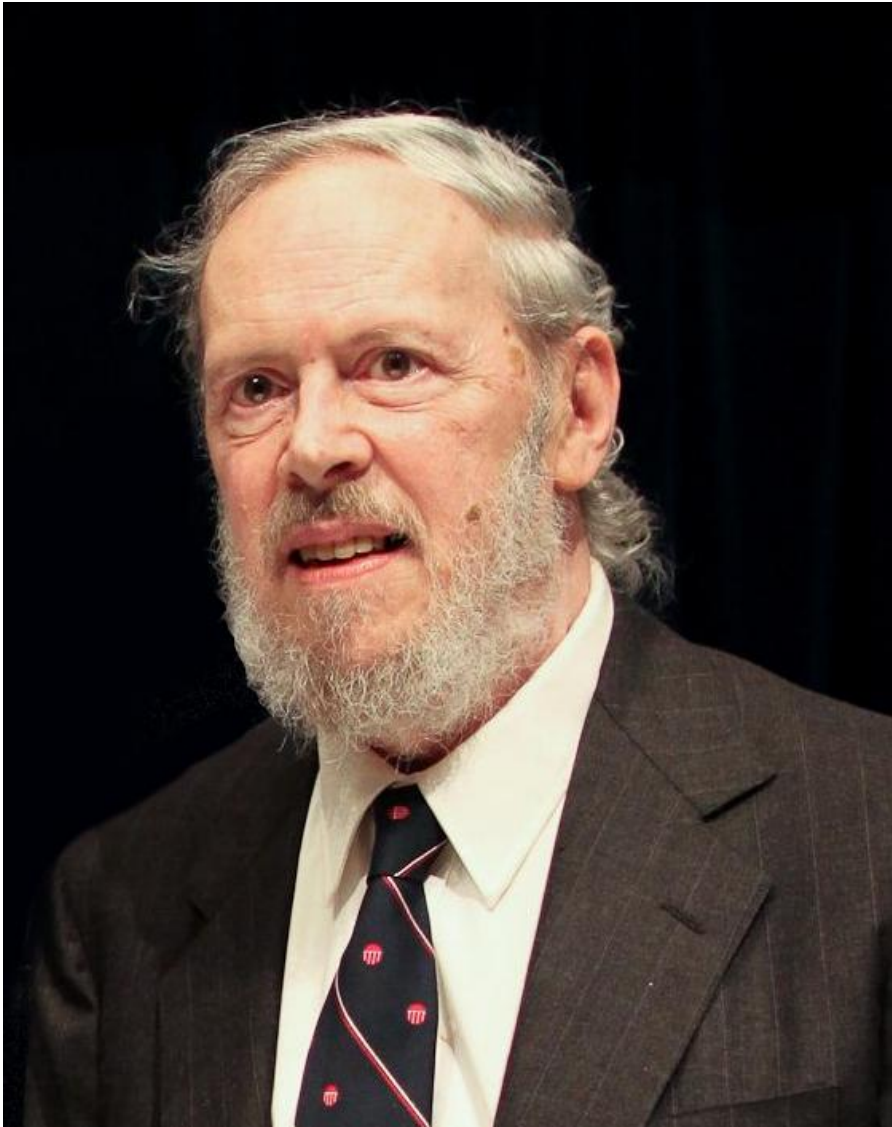
Sub CPMK (Capaian Pembelajaran Mata Kuliah)

Sub-CMPK-1 : Mampu menggunakan dan membuat program pada lingkungan Bahasa C

- Mahasiswa memahami sejarah Bahasa C
- Mahasiswa menjelaskan proses kompilasi dalam Bahasa C
- Mahasiswa menjelaskan struktur penulisan program C
- Mahasiswa menulis dan memahami contoh program dalam Bahasa C



Awal Mula Bahasa C



- Diciptakan oleh **Dennis Ritchie** pada awal tahun 1970-an di Bell Labs (Amerika Serikat).
- **Tujuannya:** membuat bahasa pemrograman yang efisien, portabel, dan cocok untuk pengembangan sistem operasi.



Lahir dari Bahasa Sebelumnya

- Bahasa C dikembangkan dari bahasa B, yang merupakan turunan dari **bahasa BCPL**.
- Bahasa BCPL yang dikembangkan oleh **Martin Richards** pada tahun 1967.
- Bahasa B terlalu terbatas, jadi Ritchie menambahkan fitur baru dan menyempurnakannya menjadi bahasa C.





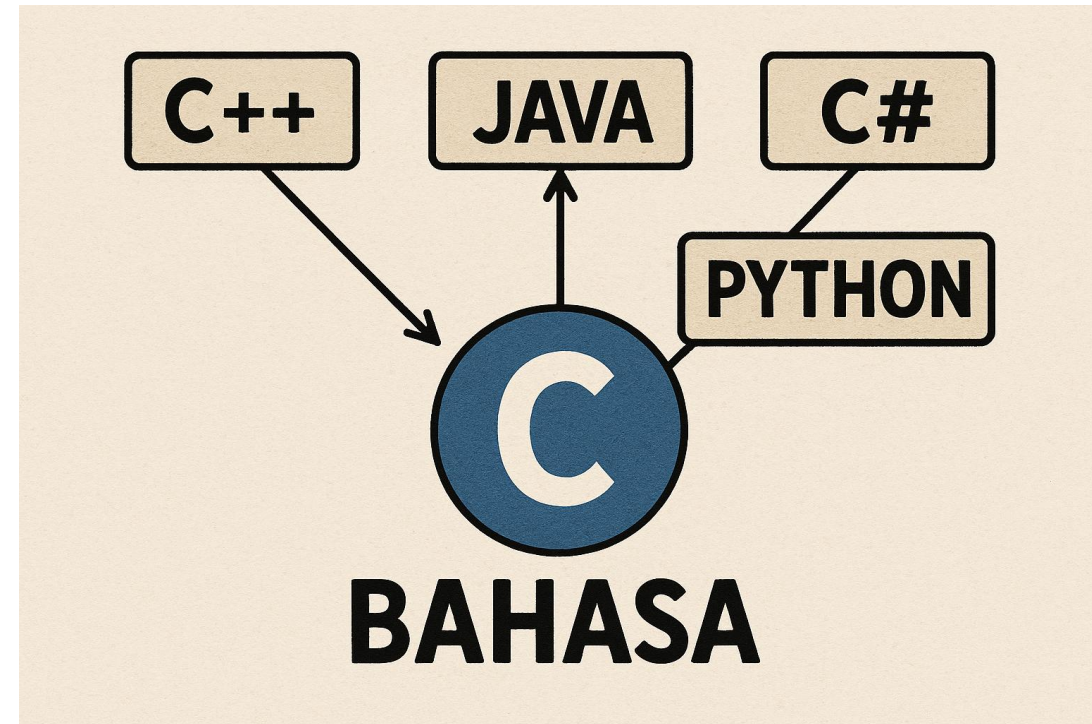
Programming

Bahasa C

- Bahasa C adalah bahasa pemrograman tingkat menengah yang dirancang untuk membuat program yang efisien, cepat, dan dekat dengan perangkat keras, tetapi tetap memiliki fitur dari bahasa tingkat tinggi.
- Bahasa C digunakan untuk membuat berbagai jenis perangkat lunak, terutama sistem operasi, aplikasi desktop, dan embedded system.

Bahasa C

- Bahasa C menjadi dasar bagi banyak bahasa modern seperti C++, Java, C#, dan Python.
- Sampai sekarang, C masih digunakan untuk pemrograman sistem, embedded system, dan pengembangan perangkat lunak yang butuh performa tinggi.





Karakteristik Utama Bahasa C

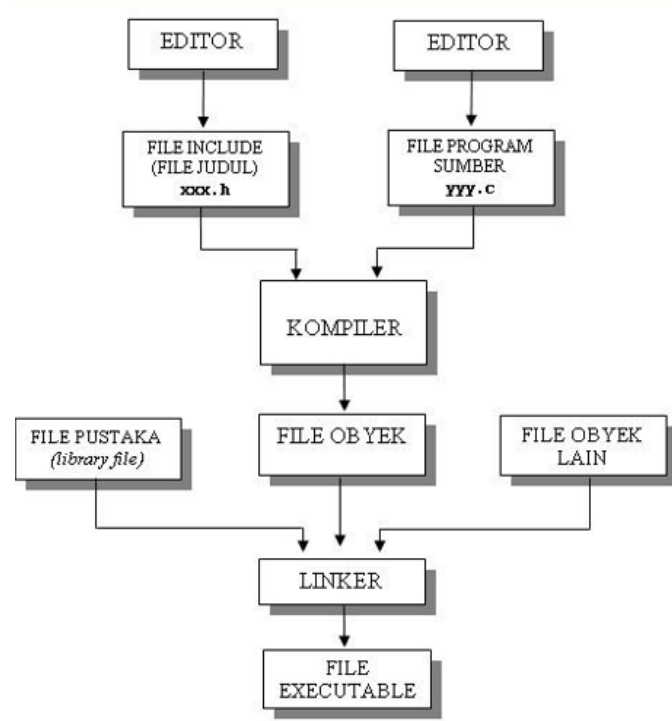
- ✓ **Struktur prosedural:** Program ditulis dalam bentuk fungsi-fungsi.
- ✓ **Cepat dan efisien:** Cocok untuk aplikasi yang butuh performa tinggi.
- ✓ **Portabel:** Bisa dijalankan di berbagai jenis komputer.
- ✓ **Dekat dengan hardware:** Bisa mengakses memori dan perangkat keras secara langsung.
- ✓ **Minimalis:** Sintaksnya sederhana, tapi sangat fleksibel

Bahasa Tingkat Rendah, Menengah, dan Tinggi

Tingkat Bahasa	Contoh	Mudah Dipahami	Akses ke Hardware	Kecepatan Eksekusi
Rendah	Assembly	✗ Sulit	✓ Langsung	✓ Sangat cepat
Menengah	C	⚠ Sedang	✓ Bisa	✓ Cepat
Tinggi	Python, Java	✓ Mudah	✗ Terbatas	⚠ Relatif lambat

Proses Kompilasi dan Linking

Agar suatu program dalam bahasa pemrograman dapat dimengerti oleh komputer, diperlukan sebuah penerjemah/translator yang akan menerjemahkan ke dalam kode mesin.



Gambar 1.1 Proses Kompilasi-Linking dari program C

```
source.c
  ↓ (Preprocessing)
source.i
  ↓ (Compilation)
source.s
  ↓ (Assembly)
source.o
  ↓ (Linking)
program.exe
```

Tahapan Proses Kompilasi dan Linking

Bahasa C menggunakan compiler untuk mengubah kode sumber menjadi file eksekusi. Proses ini terdiri dari beberapa tahap:

1. Preprocessing (Pra-pemrosesan)

- ◆ Tujuan: Menangani direktif seperti `#include`, `#define`, dan `#ifdef`

```
#include <stdio.h>
#define PI 3.14
```

2. Compilation (Kompilasi)

- ◆ Tujuan: Mengubah kode C menjadi **assembly language**

```
int main() {
    printf("Hello, world!");
    return 0;
}
```

Tahapan Proses Kompilasi dan Linking

3. Assembly (Perakitan)

- ◆ Tujuan: Mengubah kode assembly menjadi object code (kode mesin).
- ◆ Hasil: File .o atau .obj (object file)

4. Linking (Pengaitan)

- ◆ Tujuan: Menggabungkan semua object file dan library eksternal (seperti printf dari stdio.h) menjadi satu file eksekusi.
- ◆ Hasil: File .exe (Windows) atau file biner eksekusi (Linux/macOS)

Struktur Penulisan Program C

- Program C tersusun atas sejumlah **blok fungsi**, dan sebuah program minimal mengandung **sebuah fungsi**.
- Fungsi pertama yang harus ada dalam program C dan sudah ditentukan namanya adalah **main()**.
- Setiap fungsi terdiri atas **satu atau beberapa pernyataan**, yang secara keseluruhan dimaksudkan untuk melaksanakan tugas khusus.
- Bagian **pernyataan fungsi** (sering disebut tubuh fungsi) diawali dengan **tanda kurung kurawal buka ({)** dan **diakhiri dengan tanda kurung kurawal tutup (})** yang mengisyaratkan **awal dan akhir definisi fungsi**.
- Di antara kurung kurawal itu dapat dituliskan statemen-statemen program C.

```
#include <.... .h>           //preprocessor directive :  
main()                      //fungsi utama  
{  
    statemen-statemen;  
}  
  
fungsi_fungsi_lain()  
{  
    statemen-statemen; //user defined function  
}
```


Struktur Dasar Program C

```
#include <stdio.h>    // 1. Direktif Preprocessor

// 2. Fungsi utama
int main() {
    // 3. Pernyataan (statements)
    printf("Halo, dunia!\n");
    return 0;
}
```

Penjelasan Struktur Dasar Program C

1. Direktif Preprocessor

- Ditulis dengan #, misalnya **#include <stdio.h>**.
- Digunakan untuk menyertakan file header atau mendefinisikan makro.
 - **#include** adalah preprocessor directive yang memberi tahu compiler untuk menyisipkan isi file header sebelum proses kompilasi.
 - **<stdio.h>** adalah Standard Input Output Header, yaitu file yang berisi deklarasi fungsi-fungsi untuk operasi input/output. Contoh: perintah **printf()**

2. Fungsi **main()**

- Titik awal eksekusi program C.
- Wajib ada dalam setiap program.

Penjelasan Struktur Dasar Program C

3. Pernyataan (Statements)

- Instruksi yang dijalankan oleh program.

Contoh: **printf("Halo, dunia!\n");** menampilkan teks ke layar.

- Diakhiri dengan tanda titik koma ;
- Fungsi perintah **printf** => Menampilkan teks atau data ke layar (output)

4. Komentar (Opsional tapi Penting)

- Digunakan untuk memberi penjelasan dalam kode.
- Tidak dieksekusi oleh program.

- Format:

```
// Ini komentar satu baris
/* Ini komentar
   lebih dari satu baris */
```

Struktur Penulisan Program C

-
- Tanda \ pada string yang dilewatkan sebagai argumen *printf()* mempunyai makna yang khusus.
 - Tanda ini bisa digunakan untuk menyatakan karakter khusus seperti karakter baris-baru ataupun karakter *backslash* (miring kiri).
 - Jadi karakter seperti \n sebenarnya menyatakan sebuah karakter.

Contoh karakter yang ditulis dengan diawali tanda \ adalah:

\" menyatakan karakter petik-ganda

\\ menyatakan karakter backslash

\t menyatakan karakter tab

Struktur Penulisan Program C

Dalam bentuk yang lebih umum, format *printf()*

`printf("string kontrol", daftar argumen);`

dengan string kontrol dapat berupa satu atau sejumlah karakter yang akan ditampilkan ataupun berupa penentu format yang akan mengatur penampilan dari argumen yang terletak pada daftar argumen. Mengenai penentu format di antaranya berupa:

`%d` untuk menampilkan bilangan bulat (integer)

`%f` untuk menampilkan bilangan titik-mengambang (pecahan)

`%c` untuk menampilkan sebuah karakter

`%s` untuk menampilkan sebuah string

-

Aplikasi untuk menjalankan Bahasa C

Nama Aplikasi	Jenis	Fitur Utama
Code::Blocks	IDE Gratis	Mendukung GCC, debugger terintegrasi, ringan dan cocok untuk pemula
Dev-C++	IDE Gratis	Kompilasi cepat, antarmuka sederhana, mendukung C dan C++
Visual Studio Code	Editor Kode	Butuh ekstensi dan compiler (misalnya MinGW), fleksibel dan modern
CLion (JetBrains)	IDE Berbayar	Fitur canggih seperti refactoring, debugging, dan integrasi CMake
Eclipse CDT	IDE Gratis	Mendukung banyak bahasa, cocok untuk proyek besar
Replit	Online IDE	Bisa langsung coding C di browser tanpa instalasi

Link Tutorial Install C pada Visual Studio Code:

<https://www.youtube.com/watch?v=SSZgfqiBNww>

Contoh

contoh1.cpp

```
1  #include<stdio.h>
2
3  int main()
4  {
5      printf("Saya bangga menjadi mahasiswa PENS");
6  }
```

contoh2.cpp

```
1  #include <stdio.h>
2  int main()
3  {
4      int angka = 8;
5      printf("Nilai Angka : %d\n",angka);
6      return 0;
7  }
```

Output contoh1.cpp

C:\DATAKU\Materi Kuliah\2025-2026 ganjil\Konsep Pemrograman\project C\contoh1.exe

Saya bangga menjadi mahasiswa PENS

Process exited after 0.01463 seconds with return value 0
Press any key to continue . . .

Output contoh2.cpp

C:\DATAKU\Materi Kuliah\2025-2026 ganjil\Konsep Pemrograman\project C\contoh2.exe

Nilai Angka : 8

Process exited after 0.01531 seconds with return value 0
Press any key to continue . . .

"Is everything clear about the material we discussed?"

<https://www.mentimeter.com/>

Code:

1779 5691

Referensi

Brian W. Kerningham, Dennis M. Ritchie (2012): The C Programming Language : Ansi C Version 2 Edition, PHI Learning

Byron Gottfried (2010) : Programming with C, Tata McGraw - Hill Education

Kochan Stephen (2004) : Programming in C, 3rd Edition, Sams

K. N. King (2008) : C Programming: A Modern Approach, 2nd Edition, W. W. Norton & Company

Abdul Kadir (2012) : Algoritma & Pemrograman Menggunakan C & C++, Andi Publisher, Yogyakarta

<http://www.gdsw.at/languages/c/programming-bbrowne/>

<https://www.petanikode.com/tutorial/c/>

<http://www.cprogramming.com/tutorial/c-tutorial.html>