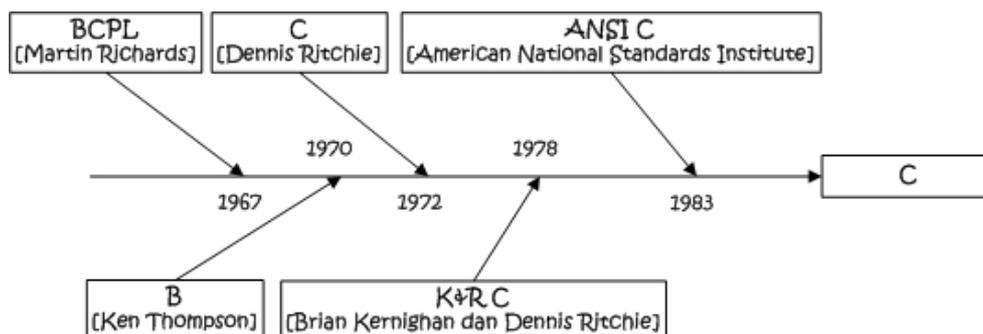


Bab 1

Pendahuluan

1.1. Sejarah Bahasa C

Sejarah perkembangan dan latar belakang munculnya bahasa C adalah seperti dalam Gambar 1.



Gambar 1: Sejarah Bahasa C

Boleh dikatakan bahwa akar dari bahasa C adalah bahasa BCPL yang dikembangkan oleh *Martin Richards* pada tahun 1967. Kemudian berdasar pada bahasa BCPL ini *Ken Thompson* yang bekerja di Bell Telephone Laboratories (Bell Labs) mengembangkan bahasa B pada tahun 1970. Saat itu bahasa B telah berhasil diimplementasikan di komputer DEC PDP-7 dengan operating system (OS) UNIX. Pada tahun 1972, peneliti lain di Bell Labs bernama *Dennis Ritchie* menyempurnakannya menjadi bahasa C.



Gambar 2: Tokoh Yang Berjasa Dalam Pengembangan Bahasa C

Pada tahun 1978, *Dennis Ritchie* bersama dengan *Brian Kernighan* mempublikasikan buku yang kemudian menjadi legenda dalam sejarah perkembangan bahasa C, yang berjudul *The C Programming Language*. Buku ini diterbitkan oleh Prentice Hall, dan pada saat ini telah diterjemahkan dalam berbagai bahasa di dunia. Boleh dikatakan bahwa buku ini adalah buku yang paling banyak direfer orang dan dijadikan buku panduan tentang pemrograman bahasa C sampai saat ini. Teknik dan gaya penulisan bahasa C yang merujuk kepada buku ini kemudian terkenal dengan sebutan *K&R C* atau *Classic C* atau *Common C*.

Seiring dengan berkembang pesatnya bahasa C, banyak vendor mengembangkan kompiler C menurut versi masing-masing. Hal ini menggerakkan ANSI (*American National Standards Institute*) pada tahun 1983 untuk membuat suatu komite yang kemudian diberi nama *X3J11*, yang bertujuan untuk membuat definisi standar bahasa C yang lebih modern dan komprehensif, dengan memperbaiki *syntax* dan *grammar* bahasa C. Usaha ini berhasil diselesaikan 5 tahun kemudian, yaitu ditandai dengan lahirnya standard ANSI untuk bahasa C yang kemudian terkenal dengan sebutan *ANSI C* pada tahun 1988.

1.2. Mengapa Memakai Bahasa C

Sampai saat ini, bahasa C telah berhasil digunakan untuk mengembangkan berbagai jenis permasalahan pemrograman, dari level operating system (unix, linux, ms dos, dsb), aplikasi perkantoran (text editor, word processor, spreadsheet, dsb), bahkan sampai pengembangan sistem pakar (*expert system*). Kompiler C juga telah tersedia di semua jenis platform komputer, mulai dari Macintosh, UNIX, PC, Micro PC, sampai super komputer.

C bisa disebut bahasa pemrograman tingkat menengah (*middle level programming language*). Arti tingkat (level) disini adalah kemampuan mengakses fungsi-fungsi dan perintah-perintah dasar bahasa mesin/hardware (*machine basic instruction set*). Semakin tinggi tingkat bahasa pemrograman (misalnya: java), semakin mudahnya bahasa pemrograman dipahami manusia, namun membawa pengaruh semakin berkurang kemampuan untuk mengakses langsung instruksi dasar bahasa mesin. Demikian juga sebaliknya dengan bahasa pemrograman tingkat rendah (misalnya: assembler), yang

semakin sulit dipahami manusia dan hanya berisi perintah untuk mengakses bahasa mesin. Dalam perspektif mudahnya dipahami manusia, C bisa digolongkan dalam bahasa tingkat tinggi, namun C juga menyediakan kemampuan yang ada pada bahasa tingkat rendah, misalnya operasi bit, operasi byte, pengaksesan memori, dsb.

Beberapa alasan mengapa memakai bahasa C adalah terangkum dibawah.

- *C adalah bahasa pemrograman yang paling populer saat ini*
Dengan banyaknya programmer bahasa C, membawa pengaruh semakin mudahnya kita menemukan pemecahan masalah yang kita dapatkan ketika menulis program dalam bahasa C. Pengaruh positif lain adalah semakin banyaknya kompiler yang dikembangkan untuk berbagai platform (berpengaruh ke portabilitas).
- *C adalah bahasa pemrograman yang memiliki portabilitas tinggi*
Program C yang kita tulis untuk satu jenis platform, bisa kita kompilasi dan jalankan di platform lain dengan tanpa ataupun hanya sedikit perubahan. Ini bisa diwujudkan dengan adanya standarisasi ANSI untuk C.
- *C adalah bahasa pemrograman dengan kata kunci (keyword) sedikit*
Kata kunci disini adalah merupakan fungsi ataupun kata dasar yang disediakan oleh kompiler suatu bahasa pemrograman. Hal ini membawa pengaruh semakin mudahnya kita menulis program dengan C. Pengaruh lain dari sedikitnya kata kunci ini adalah proses eksekusi program C yang sangat cepat. C hanya menyediakan 32 kata kunci seperti terangkum dibawah:

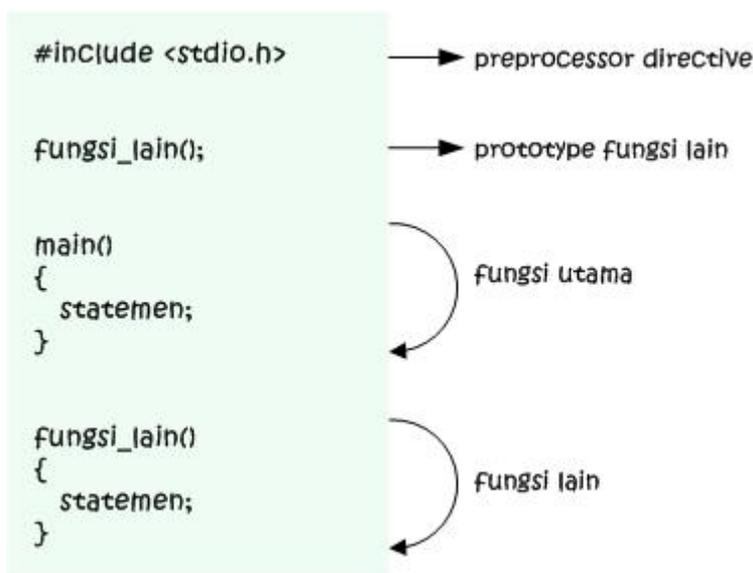
auto	break	case	char	const	continue	default
do	double	else	enum	extern	float	for
goto	if	int	long	register	return	short
signed	sizeof	static	struct	switch	typedef	union
unsigned	void	volatile	while			

- *C adalah bahasa pemrograman yang fleksibel*
Dengan menguasai bahasa C, kita bisa menulis dan mengembangkan berbagai jenis program mulai dari operating system, word processor, graphic processor, spreadsheets, ataupun kompiler untuk suatu bahasa pemrograman.
- *C adalah bahasa pemrograman yang bersifat moduler*
Program C ditulis dalam *routine* yang biasa dipanggil dengan fungsi. Fungsi-fungsi yang telah kita buat, bisa kita gunakan kembali (*reuse*) dalam program ataupun aplikasi lain.

1.3. Struktur Program Bahasa C

Program bahasa C adalah suatu program terdiri dari satu atau lebih fungsi-fungsi. Fungsi utama dan harus ada pada program C yang kita buat adalah fungsi **main()**. Fungsi **main()** ini adalah fungsi pertama yang akan diproses pada saat program di compile dan dijalankan, sehingga bisa disebut sebagai fungsi yang mengontrol fungsi-fungsi lain. Karena struktur program C terdiri dari fungsi-fungsi lain sebagai program bagian (*subroutine*), maka bahasa C biasa disebut sebagai *bahasa pemrograman terstruktur*. Cara penulisan fungsi pada program bahasa C adalah dengan memberi nama fungsi dan kemudian dibuka dengan kurung kurawal buka ({) dan ditutup dengan kurung kurawal tutup (}).

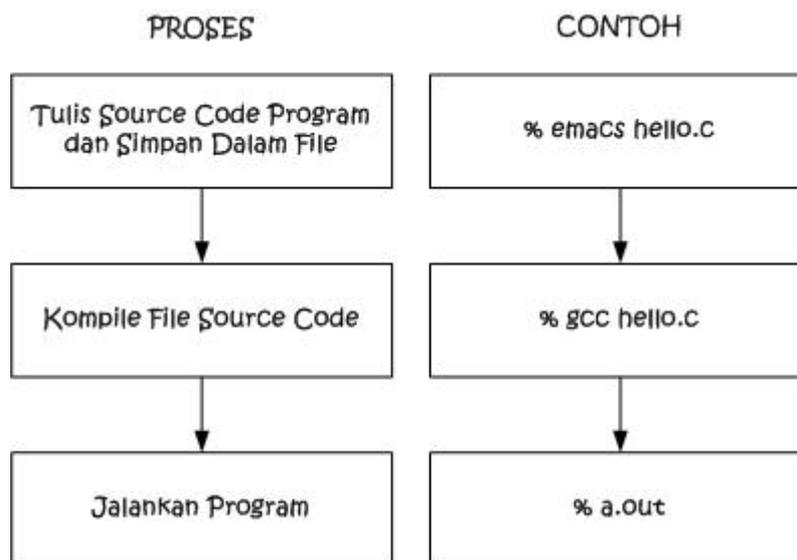
Fungsi-fungsi lain selain fungsi utama bisa dituliskan setelah atau sebelum fungsi utama dengan deskripsi prototype fungsi pada bagian awal program. Bisa juga dituliskan pada file lain yang apabila kita ingin memakai atau memanggil fungsi dalam file lain tersebut, kita harus menuliskan header filenya, dengan *preprocessor directive* **#include**. File ini disebut file pustaka (*library file*). Untuk lebih jelas tentang struktur program bahasa C ini, silakan melihat pada Gambar 3 dibawah.



Gambar 3: Struktur Dasar Program C

1.4. Proses Pembuatan Program C

Proses pembuatan program dengan menggunakan bahasa C adalah seperti dalam gambar 4.



Gambar 4: Proses Pembuatan Program C

- 1. Tulis source code program bahasa C dengan menggunakan text editor, kemudian simpan dalam sebuah file.**
Text editor disini bisa merupakan aplikasi notepad atau editplus pada windows, untuk operating system unix/linux kita bisa menggunakan aplikasi emacs yang cukup terkenal.
- 2. Kompilasi file yang berisi source code program bahasa C.**
Kompilasi atau kompilasi (compile) adalah suatu proses merubah source code ke bahasa mesin sehingga bisa dieksekusi (executable) atau dijalankan. Banyak sekali kompilasi bahasa C yang ada saat ini, baik yang gratis maupun yang kita harus membeli untuk menggunakannya. Untuk memudahkan proses belajar bahasa C, penulis memberikan rekomendasi untuk menginstall Cygwin (www.cygwin.com) bagi yang menggunakan operating system Windows. Cygwin adalah satu set free software yang dikembangkan oleh Redhat, yang berisi koleksi aplikasi dan tools UNIX yang didesain khusus untuk bisa dijalankan di Windows. Kebutuhan akan kompilasi (GCC, GNU C Compiler) dan shell (Bash Shell) untuk membuat program C bisa kita dapatkan dengan menginstall Cygwin ini.
- 3. Jalankan program yang telah dikompilasi.**
Setelah kita kompilasi file yang berisi source code, maka sebagai hasil kompilasi tersebut kita akan mendapatkan suatu file yang bisa dijalankan (*executable file*). Menjalankan program yang kita buat berarti menjalankan file hasil proses kompilasi tersebut.