

I. PENDAHULUAN

Perkembangan bahasa C erat kaitannya dengan perkembangan sistem operasi UNIX, dimana keduanya dikembangkan di AT&T Bell Lab (USA).

Pertama kali bahasa C diimplementasikan pada komputer DEC PDP-11 yang menggunakan sistem operasi UNIX oleh Dennis Ritchie di Bell Laboratories tahun 1972.

Banyak ide penting dari C diambil dari bahasa BCPL yang dikembangkan oleh Martin Richards melalui bahasa B yang ditulis oleh Ken Thompson. BCPL dan B tidak memiliki data type sedang C memiliki beragam data type.

Pada tahun 1978 Dennis Ritchie dan Brian W Kernighan mengeluarkan buku THE C PROGRAMMING LANGUAGE, buku inilah yang dijadikan acuan dari pembuatan berbagai versi bahasa C yang ada.

Pada tahun 1983 ANSI (American Nasional Standart Institute) membuat standarisasi bahasa C yang kemudian dijadikan sebagai referensi dari berbagai bahasa C yang beredar saat ini. C yang dihasilkan disebut C standard ANSI atau ANSI C. Tahun 1987 Borland menciptakan sebuah compiler C yang dikenal dengan nama TURBO C.

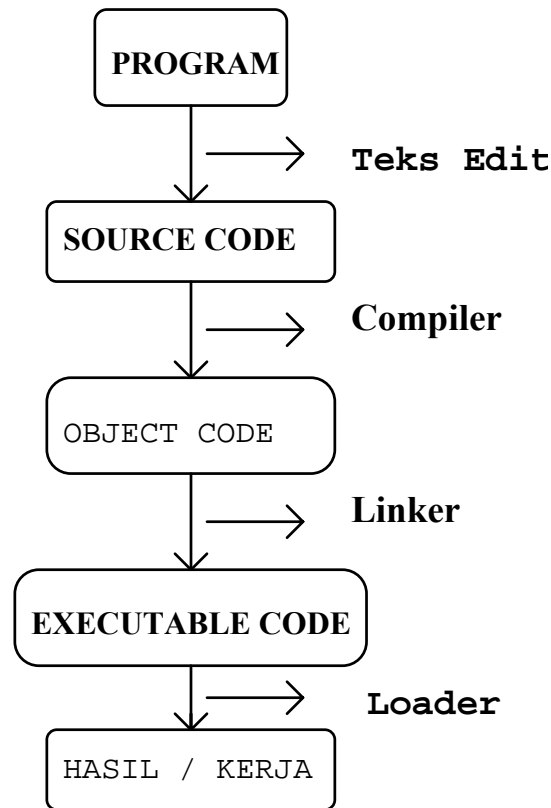
Dalam beberapa literatur bahasa C digolongkan sebagai bahasa tingkat menengah (Medium Level Language). Penggolongan ini tidak berarti bahwa bahasa C lebih sulit dibandingkan dengan bahasa pemrograman tingkat tinggi seperti : PASCAL, BASIC. Ini merupakan suatu penegasan bahwa bahasa C bukan berorientasi pada mesin seperti pada bahasa tingkat rendah (Low Level Language). Dan pada kenyataannya bahasa C mengkombinasikan elemen dalam bahasa tingkat tinggi dan bahasa tingkat rendah. Hampir semua operasi yang dapat

dilakukan oleh bahasa mesin dapat dilakukan oleh C dengan penyusunan program yang lebih sederhana dan mudah.

Bahasa C dalam pemakaiannya memerlukan suatu translator. Jenis translator dalam bahasa C adalah interpreter dan compiler. Interpreter merupakan translator yang menterjemahkan bahasa C kedalam bahasa mesin satu persatu. Contoh interpreter yang beredar saat ini yaitu Run/c. Sedangkan compiler merupakan translator yang menterjemahkan bahasa C ke dalam bahasa mesin secara keseluruhan. Contoh compiler : Turbo C, Microsoft C dan Lattice C.

Bahasa C merupakan bahasa pemrogramman yang sederhana dimana jumlah fungsi dan perintah yang disediakan terbatas, walaupun akhirnya beberapa compiler menyediakan fungsi-fungsi library yang cukup lengkap. Karena sifatnya yang sederhana ini maka bahasa C mudah ditransfer dari satu compiler ke compiler yang lain.

Proses penterjemahan bahasa C ke dalam dalam bahasa mesin dengan menggunakan compiler adalah sebagai berikut.



Pertama kali program ditulis dengan menggunakan editor dan disimpan didalam file program sumber (dengan ekstension .c). Kemudian kode file program sumber dikompilasi oleh compiler menjadi object code dan disimpan pada sebuah file object (dengan ekstension .obj). Object code sudah berbentuk kode mesin, tapi kode ini belum bisa dimengerti oleh komputer. Agar dapat dimengerti oleh komputer maka object code bersama dengan object code yang lain serta file library (file yang berisi rutin untuk tugas tertentu) perlu dikaitkan (linking) dengan menggunakan linker. Sehingga terbentuk suatu file yang executable (program yang dapat dijalankan secara langsung dalam lingkungan sistem operasi). Program hasil linker ini disimpan dalam sebuah file executable dengan ciri mempunyai ekstension .EXE

☹ Kelebihan dan kelemahan bahasa C

👍 Kelebihan :

- ⇒ C mempunyai banyak jenis data yang dikenal & operator untuk manipulasi data.
- ⇒ C menyediakan berbagai struktur data dan pengendalian proses. Sehingga memungkinkan untuk membuat program yang terstruktur (program yang mudah dipakai dan dikembangkan).
- ⇒ C mudah dipahami dibandingkan dengan bahasa mesin, karena bahasa C berorientasi pada permasalahan bukan pada mesin..
- ⇒ C mempunyai kecepatan eksekusi yang mendekati kecepatan eksekusi bahasa mesin.
- ⇒ C memungkinkan manipulasi data dalam bentuk bit maupun byte secara efisien dan dapat memanipulasi alamat dari suatu data.
- ⇒ C memakai memori yang sedikit
- ⇒ C merupakan salah satu bahasa pemrograman yang terstruktur.

👎 Kelemahan :

- ⇒ Banyaknya operator serta fleksibilitas penulisan program yang terkadang membingungkan pemakai.
- ⇒ Programmer C tingkat pemula pada umumnya kurang mengenal pointer dan tak terbiasa menggunakannya, padahal kelebihan C ada pada pointer ini.

MENGAPA C ???

- ⚙ Semakin banyak program dan software yang ditulis dalam C
- ⚙ C adalah bahasa yang modern, efisien, portable, powerful, flexible, friendly menghasilkan program yang kompak dan cepat
- ⚙ C adalah bahasa yang mendominasi pemakaian komputer dengan sistem operasi UNIX.

C adalah bahasa yang portable:

program yang ditulis pada satu sistem dapat dijalankan pada sistem lain tanpa atau dengan sedikit perubahan

C adalah bahasa yang efisien :

menghasilkan program yang kompak/ringkas dan cepat

C adalah bahasa modern :

mempunyai fasilitas - fasilitas kontrol yang diperlukan oleh teori - teori komputer science maupun aplikasi

C adalah bahasa yang friendly :

keterikatannya/ restriksinya tidak ketat

C adalah bahasa yang powerful dan fleksible

- ♦UNIX sebagian besar ditulis dalam C
- ♦C compiler ditulis dalam C
- ♦Interpreter lain (FORTRAN,APL,Pascal,LISP,BASIC) bisa ditulis dlm C
- ♦Digunakan baik dlm memecahkan masalah fisik maupun engineering
- ♦Digunakan untuk animasi
- ♦Mempunyai kontrol yang biasanya hanya dimiliki oleh bahasa assembly.

II. Pengenalan Program C

STRUKTUR PEMROGRAMAN C

Program terdiri dari koleksi satu / lebih fungsi, salah satu diantaranya harus main(). Fungsi terdiri dari sebuah header dan sebuah badan. Header berisi preprocessor statement seperti #include dan nama fungsi Badan ditandai oleh {} dan berisi seri dari statement masing diakhiri tanda ;
Adapun bentuk umum dari program bahasa C dengan Turbo C adalah

```
# include file
# define var konstan
    deklarasi fungsi
    main()
    {
        deklarasi variabel global
        :
        <pernyataan - pernyataan>
        :
    }
    nama fungsi (arg1, arg2,...)
    {
        deklarasi variabel lokal
        :
        < pernyataan - pernyataan>
        :
    }
}
```

CONTOH :

- ```
#include <stdio.h>
main()
{
 printf("hello\n");
}
```

berisi informasi tentang standar library  
mendefinisikan fungsi main.  
fungsi main tidak ada argumen  
awal dari badan fungsi  
fungsi main memanggil library function  
printf yang mencetak sejumlah karakter  
dalam tanda "..."  
akhir dari badan fungsi

OUTPUT :

hello

```
2. #include <stdio.h>
 main() /* program sederhana*/
 {
 int num;
 num = 4;
 printf ("I am a student");
 printf ("in STMIK GUNADARMA. \n");
 printf ("I am in the %d nd year.\n",num);
 }
```

OUTPUT : I am a student in STMIK GUNADARMA.  
I am in the 2 nd year.

### **PENJELASAN SINGKAT PROGRAM**

`#include <stdio.h>`

- menyertakan sebuah file lain yaitu file `stdio.h`
- berisi informasi ttg Input & Output , bagian dari paket C Compiler
- `#` merupakan instruksi PREPROCESSOR, diproses oleh C-PREPROCESSOR.
- baris ini bukan sebuah statement dalam bahasa C
- baris ini tidak selalu diperlukan

`main()`

- sebuah nama fungsi , tanda kurung mencakup informasi yang merupakan argumen.
- dalam hal ini fungsi `main` tidak mempunyai argumen
- Harus ada dalam program , karena dari fungsi ini program akan mulai dieksekusi, tanpa fungsi ini program tidak akan dapat dilaksanakan

`/* program sederhana*/`

- sebuah komentar;memudahkan orang lain dalam mengerti program kita
- Simbol `/*` dan `*/` dipakai untuk menandai komentar segala sesuatu di dalam tanda `/*` dan `*/` diabaikan oleh C compiler

{

- menandai awal dari badan fungsi

}

- tanda untuk mengakhiri definisi fungsi

;(titik koma)

- untuk mengakhiri tiap pernyataan/perintah
- antara pernyataan/perintah yang satu dengan yang lain hanya dapat dipisahkan dengan tanda titik koma (;) bukan dengan baris yang berbeda atau spasi atau karakter yang lain

int num;

- sebuah deklarasi statement yaitu mendefinisikan variabel bernama num sebagai tipe integer.

deklarasi statement sangat penting, karena mendeklarasikan 2 hal :

1. dalam fungsi tersebut akan ada variabel yang bernama num
2. variabel num mempunyai tipe integer.

dalam C, semua variabel harus dideklarasikan

num = 4;

- sebuah assignment statement yi pemberian harga 4 pd variabel num
- assignmenet statement adalah operasi paling dasar .

printf ("I am a student");

- sebuah print statement yaitu mencetak kalimat dalam tanda petik

printf ("in STMIK GUNADARMA.\n");

- \n merepresentasikan 1 karakter tunggal yang adalah menginstruksikan komputer untuk mengawali baris baru.
- \n adalah salah satu dari escape sequence.

printf (" I am in the %d nd year.\n",num);

- mencetak kalimat dalam tanda petik dengan meletakkan harga variabel num (yaitu 4) di posisi %d.



- harga dari num mensubstitusi simbol %d yaitu
  1. menempatkannya pada posisi %d.
  2. menampilkannya sebagai digit.

kalimat dalam tanda kurung adalah informasi yang diteruskan oleh fungsi main(), ke fungsi printf(); informasi tersebut disebut Argumen. Jadi pada baris pertama "I am a student" adalah argumen.

fungsi printf() mengambil isi dari tanda petik dan menampilkannya di layar.

```
#include <stdio.h>
/*mencetak konversi 200 C ke dalam Fahrenheit */
main()
{
 int fahr, celcius;
 celcius = 20;
 fahr = (9*celcius/5)+32;
 printf ("%d derajat Celcius = %d derajat Fahrenheit.\n",
 Celcius, fahr);
}
```

OUTPUT :

20 derajat Celcius = 68 derajat Fahrenheit

Catatan :     deklarasi 2 variabel  
               kalkulasi  
               2 substitusi pada fungsi printf()

```
#include <stdio.h>

main()
{
 printf ("saya akan memanggil fungsi pelayan.\n");
 pelayan();
 print ("Ya. Bawakan saya segelas kopi dan disket.\n");
}
```

```
pelayan()
{
 printf ("anda memanggil pak ?\n");
}
```

OUTPUT :

saya akan memanggil fungsi pelayan.  
anda memanggil pak ?  
Ya. Bawakan saya segelas kopi dan disket.

Catatan : menyertakan fungsi yang bukan fungsi standard dari C compiler  
yaitu : pelayan()

```
#include <stdio.h>
main()
{
 printf ("hari ini,");
 panggil();
 printf ("tanggal 1 Oktober 1994.\n");
}
panggil()
{
 printf ("Sabtu."\n);
}
```