

Cepat Mahir Algoritma dalam C

I Putu Gede Darmawan
ipgd_bali@yahoo.com

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

Seluruh dokumen di IlmuKomputer.Com dapat digunakan, dimodifikasi dan disebarkan secara bebas untuk tujuan bukan komersial (nonprofit), dengan syarat tidak menghapus atau merubah atribut penulis dan pernyataan copyright yang disertakan dalam setiap dokumen. Tidak diperbolehkan melakukan penulisan ulang, kecuali mendapatkan ijin terlebih dahulu dari IlmuKomputer.Com.

Konversi Basis Bilangan

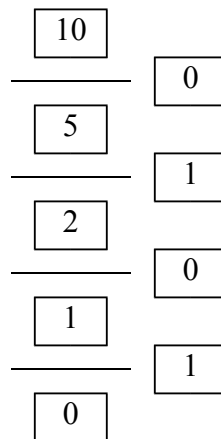
Biasanya dalam berinteraksi dalam dunia nyata, kita menggunakan bilangan dengan basis 10 (Desimal) sebagai perantaranya. Tapi karena satu dan lain hal kita perlu juga menggunakan basis bilangan lain. Dalam dunia digital misalnya. Ukuran terkecil untuk menyatakan satuan data dalam dunia digital ialah bit. 1 Bit terdiri dari 2 buah keadaan yaitu 1 atau 0, low atau high dsbnya. Mengingat pentingnya proses konversi tersebut maka kali ini kita akan membahas cara mengkonversi bilangan dengan basis dasar 10 menjadi bilangan dengan basis 2 sampai 16. Dan dengan algoritma yang sama anda bisa kembangkan sendiri untuk mengkonversi sampai basis ke N. Kita akan menggunakan Metode rekursif untuk pembentukan algoritmanya.

Metode Konvensional

Oke sebelum menginjak kepada metode konvensional, ada beberapa hal yang perlu kita ketahui. Maksud dari basis bilangan 10 ialah karakter bilangan yang tersedia ialah dari '0' sampai dengan '9'. Maksud dari basis bilangan 2, karakter yang tersedia ialah dari '0' sampai '1'. Nah bagaimana dengan basis yang lebih besar dari 10? Jika basis lebih besar dari 10 digunakan karakter abjad. Misalkan basis 16. Maka karakter bilangan yang tersedia yaitu dari '0' .. '9', 'A', 'B', 'C', 'D', 'E', 'F'. Begitu seterusnya untuk bilangan diatas basis 10.

Dibawah ini ialah salah satu metode konvensional untuk merubah bilangan dengan basis dasar 10 ke basis lainnya , yang mana penulis dapat ketika masih duduk di bangku sekolah dasar 😊.

Misalkan kita ingin mencari berapakah nilai bilangan 10 dalam basis 10 jika di konversi ke basis 2 (biner) . Untuk itu perhatikan gambar berikut



Gambar1

Pada gambar 1 didapat bahwa $10_{(10)} = 1010_{(2)}$

Algoritma pada gambar 1 dapat dijelaskan sebagai berikut . Nilai 10 kita bagi dengan basis tujuan yang kita mau (dalam hal ini 2) maka akan dihasilkan 5 , dengan sisa 0 . Kemudian 5 kita kembali bagi dengan basis yang kita tuju , sehingga 5 dibagi dengan 2 . Maka akan didapat 2 dengan sisa 1 begitu seterusnya. Langkah ini diulangi terus sampai hasil dari pembagian ialah 0 . Hasil konversinya didapat dari merunut sisa hasil pembagian dari yang paling bawah ke atas (lihat gambar) Dari gambar diperoleh 1010 sebagai hasil.

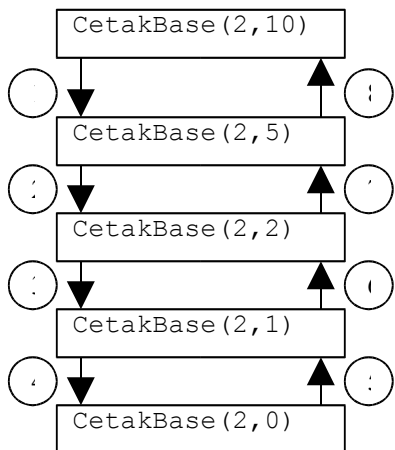
Listing Program

Setelah kita tahu cara metode pengkonversiannya , maka tentunya tidak sulit lagi untuk membuatnya kedalam bahasa pemrograman C . Untuk memudahkan dan sekaligus memahami penggunaan rekursif , maka fungsi akan dibuat dengan menggunakan rekursif

Berikut ialah listing programnya :

```
#include <stdio.h>
char ch[17] = "0123456789ABCDEF";
void CetakBaseN(int base,int nilai)
{
    if (nilai == 0)        //jika nilai sudah 0 maka keluar
        return;
    CetakBaseN(base,nilai / base);
    printf("%c",ch[nilai % base]);
}
void main(void)
{
    CetakBaseN(2,255); printf("\n");
    CetakBaseN(8,255); printf("\n");
    CetakBaseN(10,255); printf("\n");
    CetakBaseN(16,255); printf("\n");
}
```

Cara kerja Fungsi CetakBaseN



Misalkan kita memanggil fungsi CetakBaseN dengan nilai base = 2 dan nilai = 10 . Maka pertama – tama dicek apakah nilai = 0 , jika sama maka langsung keluar . Ternyata nilai tidak sama dengan 0 maka lanjutkan pada baris berikutnya . Yaitu memanggil dirinya sendiri dengan parameter nilai yang kini bernilai nilai / base ($10 / 2 = 5$) (Langkah 1) Kemudian di cek kembali apakah nilai == 0 , dilanjutkan dengan memanggil dirinya sendiri yang kini parameter nilai bernilai nilai / base ($5 / 2 = 2$) (Langkah 2) begitu seterusnya . Ketika nilai == 0 (pada pemanggilan fungsi yang ke 5) maka fungsi akan keluar (Langkah 5) dan akan melanjutkan fungsi pemanggilnya yaitu mencetak sisa dari nilai / base . Kemudian fungsi akan keluar lagi (Langkah 6) dan kembali mencetak sisa dari nilai / base fungsi pemanggilnya dan begitu seterusnya.