

Matematika Biner

Abe Poetra

abe@ilmukomputer.com

abe_poetra@yahoo.com

YM! = abe_poetra

Lisensi Dokumen:

Copyright © 2003 IlmuKomputer.Com

*Seluruh dokumen di **IlmuKomputer.Com** dapat digunakan, dimodifikasi dan disebarakan 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**.*

1. Kata-Kata Introduksi

Teman-teman sekalian pada tulisan kali ini penulis akan mengajak anda untuk membahas tentang *Matematika Biner* (kali ini? Wah, serasa banyak saja tulisan yang udah dibuat 😊), yaitu sebuah konsep penghitungan binary berbasis dua, dimana hal ini nantinya akan berkaitan dengan proses komputasi logika sebagai prinsip kerja mesin komputer.

Matematika biner selalu disajikan sebagai mata kuliah pengantar bagi anda yang berdisiplin ilmu komputer, biasanya diajarkan dalam mata kuliah *Data Processing (Pemrosesan data)*, *Pengantar Organisasi Komputer (POK)* dan mata kuliah *Matematika Diskret (Madis)*. Dalam pengajaran beberapa mata kuliah juga mengandung unsur-unsur penghitungan biner ini, akan tetapi tidak terlalu mutlak.

Jika anda berminat untuk mempelajari konsep jaringan, setidaknya anda harus mengerti konsep *IP Address* dimana anda diwajibkan tahu untuk

pengubahan dari biner ke desimal ataupun dari desimal menjadi biner. Hal ini berkaitan langsung dengan *kelas jaringan* dan *subnetting*.

Beberapa contoh pada tutorial ini disajikan sangat sederhana, dengan tujuan agar lebih mudah memahaminya. Dan penulis juga menyinggung sedikit tentang sejarah kata menghitung dan berhitung. *Membosankan?* Tentu saja ☺. Jadi disarankan kalau memang anda malas untuk membaca sejarahnya, silahkan langsung pada praktiknya.

Selamat membaca, dan nikmatilah kepusingan anda! Ops.. Hampir lupa, tulisan ini ditujukan untuk kamu yang *Newbie* (baca: pemula!)

2. Konsep Menghitung dan Berhitung

Berhitung adalah cabang dari matematika. Tetapi sekalipun sebagai cabang, berhitung telah menelusuri seluruh tubuh matematika. Anda bisa lihat metoda berhitung ini ada pada aljabar, dalam ilmu ukur (geometri), di teori kemungkinan (probabilitas), statistika, analisis, teori fungsi, topologi, dan hampir keseluruhan batang tubuh matematika (hihihi.. seperti UU aja yah ☺)

Jika anda lirik kamus *Webster's New Third International Dictionary*, kata *berhitung* dirumuskan sebagai "cabang matematika yang berkenaan dengan sifat dan hubungan bilangan-bilangan nyata dan dengan perhitungan diantaranya, terutama berkenaan dengan penjumlahan, pengurangan, perkalian dan pembagian". Sementara pada kamus *Concise Oxford English Dictionary* dan *American Encyclopedia* dengan singkat mengartikan berhitung sebagai "ilmu tentang bilangan".

Mari kita kembali ke zaman Yunani Kuno, yang telah mengenal kata berhitung sejak zaman tarikh masehi. Mereka menamakan berhitung dengan kata *arithmetike*, suatu istilah yang diturunkan dari kata *arithmos* yang berarti "bilangan" dan *techne* yang berarti "ilmu pengetahuan". Bahkan mereka

merumuskan lebih jauh lagi, bahwa berhitung dan menghitung adalah dua hal yang berbeda sekalipun keduanya bersumber pada satu induk yang sama.

Dari sinilah timbul istilah *logistica*, yang kalau kita tarik kembali ke sejarah Arthur Schopenhauer ketika dia menggambarkan berhitung sebagai “kejiwaan yang terendah karena ternyata dapat dilakukan oleh mesin”.

Lantas apa hubungannya dengan komputer? Walah.. belum jelas juga? Baiklah. Pemikiran dari Schopenhauer itu benar adanya, anda lihat pada masa sekarang telah diciptakan alat bantu menghitung, contohnya alat penjumlah tangan (swipoa), kalkulator elektronik, dan komputer dewasa ini diciptakan untuk melaksanakan *logistica* yang sederhana maupun sampai tingkat yang rumit. Oke, sampai saat ini sudah jelas bukan? Belum? Hehehe.. *no coment* deh! Ini berarti penulis yang kurang jelas atau anda yang..? Silahkan di resapi lagi kata-katanya.

3. Matematika Biner

Pada system bilangan desimal, anda pasti akan sangat mudah menandai dan menghitung, karena memang hal ini berkenaan dengan perhitungan sehari-hari. Penulis tidak menklaim bahwa perhitungan biner itu susah, akan tetapi lebih terutama karena faktor kebiasaan saja. Sebagai contoh dari bilangan desimal, untuk angka 157:

$$157_{(10)} = (1 \times 100) + (5 \times 10) + (7 \times 1)$$

Perhatikan! Sekarang anda tahu mengapa bilangan desimal ini sering juga disebut basis 10 bukan? Benar. Hal ini dikarenakan perpangkatan 10 yang didapat dari 10^0 , 10^1 , 10^2 , dlsb (dan lainnya saya bingung.. ☺).

Untuk lebih jelasnya, penulis akan ajak anda mengetahui bagaimana penjumlahan, pengurangan, dan perkalian bilangan biner. Penulis juga akan mengajak anda semakin pusing dengan menerapkan gagasan komplemen pada

konsep pengurangan biner. Dalam setiap kasus, supaya anda tidak semakin bingung, penulis juga akan menyertakan bilangan desimal diantaranya.

3.1. Mengenal Konsep Bilangan Biner dan Desimal

Perbedaan mendasar dari metoda biner dan desimal adalah berkenaan dengan basis. Jika desimal berbasis 10 (X_{10}) berpangkatkan 10^x , maka untuk bilangan biner berbasiskan 2 (X_2) menggunakan perpangkatan 2^x . Sederhananya anda perhatikan contoh di bawah ini!

Untuk Desimal:

$$\begin{aligned} 14_{(10)} &= (1 \times 10^1) + (4 \times 10^0) \\ &= 10 + 4 \\ &= 14 \end{aligned}$$

Untuk Biner:

$$\begin{aligned} 1110_{(2)} &= (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0) \\ &= 8 + 4 + 2 + 0 \\ &= 14 \end{aligned}$$

Tentu saja anda masih bingung dengan konsep tersebut, akan tetapi jika anda melihat susunan dibawah ini bingung anda akan sirna. Diharapkan pada sub bagian ini anda benar-benar memahami bagaimana konsep pengubahan dari biner ke desimal sehingga nantinya anda tidak akan mengalami kesulitan pada materi selanjutnya yaitu pada proses penambahan, pengurangan, maupun perkalian.

Bentuk umum dari bilangan biner dan bilangan desimal adalah :

* Biner	1	1	1	1	1	1	1	1	11111111
* Desimal	128	64	32	16	8	4	2	1	255
* Pangkat	2^6	2^5	2^5	2^4	2^3	2^2	2^1	2^0	X^{1-6}

Sekarang kita balik lagi ke contoh soal di atas! Darimana kita dapatkan angka desimal $14_{(10)}$ menjadi angka biner $1110_{(2)}$? Mari kita lihat lagi pada bentuk umumnya!

* Biner	0	0	0	0	1	1	1	0	00001110
* Desimal	0	0	0	0	8	4	2	0	14
* Pangkat	2^6	2^5	2^5	2^4	2^3	2^2	2^1	2^0	X^{1-6}

Mari kita telusuri perlahan-lahan!

- Pertama sekali, kita jumlahkan angka pada desimal sehingga menjadi 14. anda lihat angka-angka yang menghasilkan angka 14 adalah 8, 4, dan 2!
- Untuk angka-angka yang membentuk angka 14 (lihat angka yang diarsir), diberi sign biner "1", sebaliknya diberi sign "0".
- Sehingga kalau dibaca dari kanan, angka desimal 14 akan menjadi 00001110 (terkadang dibaca 1110) pada angka biner nya.

Mungkin untuk lebih jelas, saya akan ajak anda membahas soal-soal pengkonversian dari biner ke desimal, atau sebaliknya dari desimal ke biner.

3.2. Mengubah Angka Biner ke Desimal

Saya akan berikan 4 soal, silahkan dipelajari sehingga anda benar-benar familiar dengan bentuk dan otomatis mampu untuk mempelajari tahapan berikutnya. Perhatikan contoh!

1. 11001101₍₂₎

* Biner	1	1	0	0	1	1	0	1	11001101
* Desimal	128	64	0	0	8	4	0	1	205
* Pangkat	2^6	2^5	2^5	2^4	2^3	2^2	2^1	2^0	X^{1-6}

Note:

- Angka desimal 205 didapat dari penjumlahan angka yang di arsir (128+64+8+4+1)
- Setiap biner yang bertanda "1" akan dihitung, sementara biner yang bertanda "0" tidak dihitung, alias "0" juga.

2. $00111100_{(2)}$

* Biner	0	0	1	1	1	1	0	0	00111100
* Desimal	0	0	32	16	8	4	0	0	60
* Pangkat	2^6	2^5	2^5	2^4	2^3	2^2	2^1	2^0	X^{1-6}

3. $11111111_{(2)}$

* Biner	1	1	1	1	1	1	1	1	11111111
* Desimal	128	64	32	16	8	4	2	1	255
* Pangkat	2^6	2^5	2^5	2^4	2^3	2^2	2^1	2^0	X^{1-6}

4. $11000000_{(2)}$

* Biner	1	1	0	0	0	0	0	0	11000000
* Desimal	128	64	0	0	0	0	0	0	192
* Pangkat	2^6	2^5	2^5	2^4	2^3	2^2	2^1	2^0	X^{1-6}

3.3. Mengubah Angka Desimal ke Biner

Untuk mengubah angka desimal menjadi angka biner digunakan metode pembagian dengan angka 2 sambil memperhatikan sisanya. Mari kita perhatikan contohnya!

1. $205_{(10)}$

205	: 2	= 102	sisa 1	↑
102	: 2	= 51	sisa 0	
51	: 2	= 25	sisa 1	
25	: 2	= 12	sisa 1	
12	: 2	= 6	sisa 0	
6	: 2	= 3	sisa 0	
3	: 2	= 1	sisa 1	
1	→ sebagai sisa akhir "1"			

Note:

Untuk menuliskan notasi binernya, pembacaan dilakukan dari bawah yang berarti $11001101_{(2)}$

2. $60_{(10)}$

$$60 : 2 = 30 \text{ sisa } 0$$

$$30 : 2 = 15 \text{ sisa } 0$$

$$15 : 2 = 7 \text{ sisa } 1$$

$$7 : 2 = 3 \text{ sisa } 1$$

$$3 : 2 = 1 \text{ sisa } 1$$

1 → sebagai sisa akhir "1"

Note:

Dibaca dari bawah menjadi $111100_{(2)}$ atau lazimnya dituliskan dengan $00111100_{(2)}$. Ingat bentuk umumnya mengacu untuk 8 digit! Kalau 111100 (ini 6 digit) menjadi 00111100 (ini sudah 8 digit).

3. $14_{(10)}$

$$14 : 2 = 7 \text{ sisa } 0$$

$$7 : 2 = 3 \text{ sisa } 1$$

$$3 : 2 = 1 \text{ sisa } 1$$

1 → sebagai sisa akhir "1"

Note:

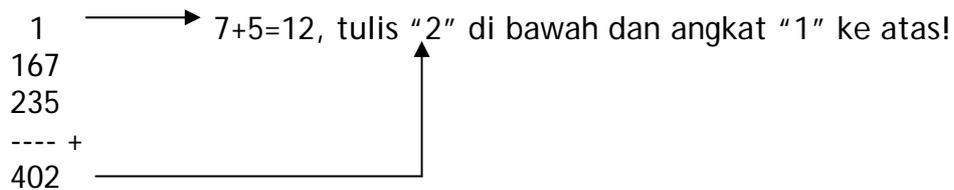
Dibaca dari bawah $1110_{(2)}$ atau dituliskan $00001110_{(2)}$ dengan 8 digit.

4. Aritmatika Biner

Pada bagian ini kita akan membahas penjumlahan dan pengurangan biner. Perkalian biner adalah pengulangan dari penjumlahan; dan kita juga akan membahas pengurangan biner berdasarkan ide atau gagasan komplemen.

4.1. Penjumlahan Biner

Penjumlahan biner tidak begitu beda jauh dengan penjumlahan desimal. Perhatikan contoh penjumlahan desimal antara 167 dan 235!



Penghitungan desimal diatas sangat sederhana sekali konsepnya, wah.. kalau anda tidak tau kebangetan deh! Sejak Sekolah Dasar (SD) perhitungan ini sudah diajarkan.

Seperti bilangan desimal, bilangan biner juga dijumlahkan dengan cara yang sama. Pertama-tama yang harus dicermati adalah aturan pasangan digit biner berikut:

$$0 + 0 = 0$$

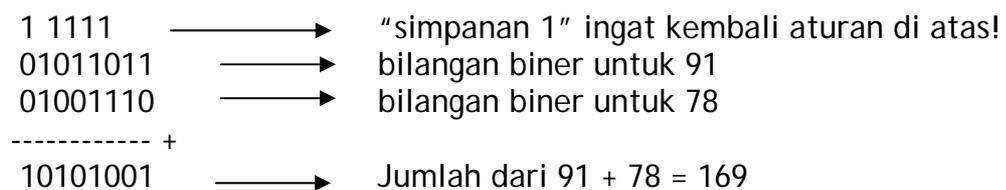
$$0 + 1 = 1$$

$$1 + 1 = 0 \rightarrow \text{dan menyimpan 1}$$

sebagai catatan bahwa jumlah dua yang terakhir adalah :

$$1 + 1 + 1 = 1 \rightarrow \text{dengan menyimpan 1}$$

Dengan hanya menggunakan penjumlahan-penjumlahan di atas, kita dapat melakukan penjumlahan biner seperti ditunjukkan di bawah ini:



Silahkan pelajari aturan-aturan pasangan digit biner yang telah disebutkan di atas! Untuk memudahkan anda, angka desimalnya juga penulis sertakan, walaupun sekarang tanpa disertai angka desimal tersebut, penulis asumsikan anda sudah paham.

Ok, sekarang penulis ingin coba kemampuan anda untuk memahami soal berikut. Kita akan menghitung penjumlahan biner yang terdiri dari 5 bilangan!

```

11101   bilangan 1)
10110   bilangan 2)
  1100   bilangan 3)
11011   bilangan 4)
  1001   bilangan 5)
----- +
    
```

untuk menjumlahkannya, kita hitung berdasarkan aturan yang berlaku, dan untuk lebih mudahnya perhitungan dilakukan bertahap!

```

11101   bilangan 1)
10110   bilangan 2)
----- +
110011
  1100   bilangan 3)
----- +
111111
  11011   bilangan 4)
----- +
1011010
  1001   bilangan 5)
----- +
1100011 → Jumlah Akhir ☺
    
```

} Berapakah bilangan desimal untuk bilangan 1,2,3,4 dan 5 !!

Nah, sekarang coba kamu tentukan berapakah bilangan 1,2,3,4 dan 5! Apakah memang perhitungan di atas sudah benar? Kalau memang sudah benar, ya sudah tidak usah dihitung lagi. Tapi kalau kamu memang ragu, ada baiknya kamu terjemahkan ke desimal terlebih dahulu satu demi satu.

4.2. Pengurangan Biner

Untuk memahami konsep pengurangan biner, kita harus mengingat kembali perhitungan desimal (angka biasa), kita mengurangkan digit desimal dengan digit desimal yang lebih kecil. Jika digit desimal yang dikurangkan lebih kecil daripada digit desimal yang akan dikurangi, maka terjadi "konsep peminjaman". Digit tersebut akan meminjam 1 dari digit sebelah kirinya.

Hihihi.. bingung? Baiklah mari kita lihat contoh saja! Pengurangan bilangan desimal 73426 - 9185 akan menghasilkan:

```

  6 3  —————> angka yang telah di pinjam menjadi 6 dan 3!
713426 —————> lihat! Angka 7 dan angka 4 dikurangi dengan 1
    
```

$$\begin{array}{r}
 91\ 85 \quad \longrightarrow \text{ digit desimal pengurang.} \\
 \hline
 6\ 42\ 41 \quad \longrightarrow \text{ Hasil pengurangan akhir 😊}
 \end{array}$$

Pengurangan biner dapat dilaksanakan dengan cara yang sama. Tapi untuk menghindari kebingungan silahkan lihat Bentuk Umum pengurangan berikut:

$$\begin{array}{l}
 0 - 0 = 0 \\
 1 - 0 = 0 \\
 1 - 1 = 0 \\
 0 - 1 = 1 \rightarrow \text{ dengan meminjam '1' dari digit disebelah kirinya!}
 \end{array}$$

Untuk penurangan biner dapat dilakukan dengan cara yang sama. Coba perhatikan bentuk pengurangan berikut:

$$\begin{array}{r}
 1111011 \quad \text{desimal 123} \\
 101001 \quad \text{desimal 41} \\
 \hline
 1010010 \quad \text{desimal 82}
 \end{array}$$

Pada contoh di atas tidak terjadi "konsep peminjaman". Perhatikan contoh berikut!

$$\begin{array}{r}
 0 \quad \longrightarrow \text{ kolom ke-3 sudah menjadi '0', sudah dipinjam!} \\
 111101 \quad \longrightarrow \text{ desimal 61} \\
 10010 \quad \longrightarrow \text{ desimal 18} \\
 \hline
 101011 \quad \longrightarrow \text{ Hasil pengurangan akhir 43 😊}
 \end{array}$$

Pada soal yang kedua ini kita pinjam '1' dari kolom 3, karena ada selisih 0-1 pada kolom ke-2. Lihat Bentuk Umum!

Jika anda gokil dan cermat pasti anda akan tanya "Bagaimana jika saya tidak dapat meminjam 1 dari kolom berikutnya karena kolom tersebut berupa bilangan '0'?"

Wah, berarti anda orang yang cermat. Untuk membahasa hal itu mari kita beri bandingkan jika hal ini terjadi pada bilangan desimal. Mari kita hitung desimal 800046 - 397261!

$$\begin{array}{r} 7999 \\ 8000^{1}46 \\ 3972\ 61 \\ \hline 4027\ 05 \end{array}$$

Perhatikan bahwa kita meminjam 1 dari kolom keenam untuk kolom kedua, karena kolom ketiga, keempat dan kolom kelima adalah nol. Setelah meminjam, kolom ketiga, keempat, dan kelima menjadi: $10 - 9 = 1$

Hal ini juga berlaku dalam pengurangan biner, kecuali bahwa setelah meminjam kolom nol akan mengandung: $10 - 1 = 1$

Sebagai contoh pengurangan bilangan biner 110001-1010 akan diperoleh hasil sebagai berikut:

$$\begin{array}{r} 1100^{1}01 \\ 10\ 10 \\ \hline 1001\ 11 \end{array}$$

4.3. Komplemen

Salah satu metoda yang dipergunakan dalam pengurangan pada komputer yang ditransformasikan menjadi penjumlahan dengan menggunakan *minus-radiks-komplemen satu* atau *komplemen radiks*. Pertama-tama marilah kita bahas komplemen di dalam sistem desimal, dimana komplemen-komplemen tersebut secara berurutan disebut dengan *komplemen sembilan* dan *komplemen sepuluh* (komplemen di dalam system biner disebut dengan *komplemen satu* dan *komplemen dua*). Sekarang yang paling penting adalah menanamkan prinsip ini:

“Komplemen sembilan dari bilangan desimal diperoleh dengan mengurangkan masing-masing digit desimal tersebut ke bilangan 9, sedangkan komplemen sepuluh adalah komplemen sembilan ditambah 1”

Lihat contoh nyatanya!

Bilangan Desimal	123	651	914	}
Komplemen Sembilan	876	348	085	
Komplemen Sepuluh	877	349	086	→ ditambah dengan 1!

Perhatikan hubungan diantara bilangan dan komplemennya adalah simetris. Jadi, dengan memperhatikan contoh di atas, komplemen 9 dari 123 adalah 876 dengan simple menjadikan jumlahnya=9 (1+8=9, 2+7=9, 3+6=9)! Sementara komplemen 10 didapat dengan menambahkan 1 pada komplemen 9, berarti 876+1=877!

Pengurangan desimal dapat dilaksanakan dengan penjumlahan komplemen sembilan plus satu, atau penjumlahan dari komplemen sepuluh!

893	→	893	(komp. 9)	→	893
321		678			679 (komp. 10)
---- -		---- +			---- +
572		1571			1572
		1			
		---- +			
		572	→ angka 1 dihilangkan!		

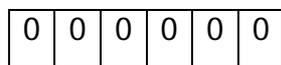
Analogi yang bisa diambil dari perhitungan komplemen di atas adalah, komplemen satu dari bilangan biner diperoleh dengan jalan mengurangkan masing-masing digit biner tersebut ke bilangan 1, atau dengan bahasa sederhananya mengubah masing-masing 0 menjadi 1 atau sebaliknya mengubah masing-masing 1 menjadi 0. Sedangkan komplemen dua adalah satu plus satu. Perhatikan Contoh ☺!

Bilangan Biner	110011	101010	011100
Komplemen Satu	001100	010101	100011
Komplemen Dua	001101	010110	100100

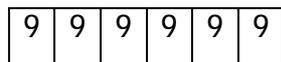
Pengurangan biner 110001 - 1010 akan kita telaah pada contoh di bawah ini!

110001 001010 ----- - 100111	110001 110101 ----- + 100111	110001 110110 ----- + 1100111 └───────────▶ dihilangkan!
---------------------------------------	---------------------------------------	--

Alasan teoritis mengapa cara komplemen ini dilakukan, dapat dijelaskan dengan memperhatikan sebuah *speedometer* mobil/motor dengan empat digit sedang membaca nol!



Jika sekarang kita tambahkan -1 pada pembacaan tersebut; yakni jika *speedometer* kita putar kembali 1 mil, maka pembacaan akan berubah menjadi!

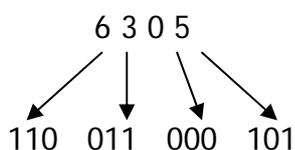


5. Sistem Oktal dan Heksa Desimal

Bilangan oktal adalah bilangan dasar delapan, sedangkan bilangan heksadesimal atau sering disingkat menjadi heks ini adalah bilangan berbasis enam belas. Karena oktal dan heks ini merupakan pangkat dari dua, maka mereka memiliki hubungan yang sangat erat. Katakanlah hubungan antara anak dengan bapaknya, atau cucu dengan neneknya ☺. Tapi yang pasti octal dan heksadesimal berkaitan dengan prinsip biner!

Untuk menepis kebingungan anda, silahkan dipelajari contoh-contoh yang saya diberikan!

- Ubahlah *bilangan oktal* 6305₈ menjadi *bilangan biner*!



Note:

- Masing-masing digit oktal diganti dengan ekivalens 3 bit (biner)
- Untuk lebih jelasnya lihat tabel Digit Oktal di bawah!

2. Ubahlah *bilangan heks* $5D93_{16}$ menjadi *bilangan biner*!

5	→	0101
D	→	1101
9	→	1001
3	→	0011

Note:

- Jadi bilangan biner untuk heks $5D93_{16}$ adalah 0101110110010011
- Untuk lebih jelasnya lihat tabel Digit Heksadesimal di bawah!

3. Ubahlah *bilangan biner* 1010100001101 menjadi *bilangan oktal*!

001	010	100	001	101
↓	↓	↓	↓	↓
3	2	4	1	5

Note:

- Kelompokkan bilangan biner yang bersangkutan menjadi 3-bit mulai dari kanan!

4. Ubahlah *bilangan biner* 101101011011001011 menjadi *bilangan heks*!

0010	1101	0110	1100	1011
2	D	6	C	B

Tabel Digit Oktal

Digit Oktal	Ekivalens 3-Bit
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Tabel Digit Heksadesimal

Digit Desimal	Ekivalens 4-Bit

0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A (10)	1010
B (11)	1011
C (12)	1100
D (13)	1101
E (14)	1110
F (15)	1111

Kalau boleh saya berikan soal tambahan untuk anda! Ubahlah *bilangan heksadesimal* ABE_{16} menjadi *bilangan biner*! Silahkan, dan ini bukan karena nama saya ABE lantas anda saya suruh mengerjakan ini. Katakanlah ini sekedar uji coba untuk anda, apakah anda sudah memahami konsep *heks* dan *biner* yang telah diuraikan diatas ☺ ! jawabanya adalah 101010111110! Lah, tau darimana? Hihi.. silahkan dilihat pada tabel Digit Heksadesimal di atas deh.

6. Penjelasan (Sangat) Singkat Tentang IP Address

Pada kata-kata introduksi saya sudah jelaskan kalau konsep dari matematika biner ini juga diterapkan pada penentuan IP Address dan konsep subnetting! Tapi untuk belajar lebih lanjut tentang IP Address ini adalah di luar cakupan tulisan ini. Jadi silahkan menambah pemahaman anda dengan membuka-buka buku jaringan anda, atau silahkan di *googling* aja sendiri!

6.1. Penulisan IP Address Dengan Bilangan Biner

IP Address terdiri dari atas 32 bit angka biner, yang dapat dituliskan kedalam empat kelompok 8 bit (oktet) dan dipisahkan oleh tanda titik. Perhatikan contoh di bawah:

11000000.10101000.00000000.00000001 → *Bilangan Biner*

dapat ditulis!

192.168.0.1 —————> Penulisan dalam *Bilangan Desimal*

Anda perhatikan contoh di atas, penulisan IP dalam biner biasanya dipergunakan untuk mempermudah anda untuk melakukan subnetting dan penentuan IP Address pada jaringan.

Tak bosan penulis bilang, untuk pemahaman yang lebih lanjut tentang pembahasan penulisan IP Address, broadcast, bahkan konsep untuk melakukan subnetting silahkan dicari!

OK, sampai jumpa di tulisan atau artikel selanjutnya! ☺

Source:

- 1) Mengetahui Matematika Biner, Marihat S. MSc, USU Press.
- 2) Catatan Kuliah Organisasi Komputer dan Matematika Diskret.

Setelah Beberapa kali ditunda, Phiuh.. akhirnya selesai juga!
Medan, Go-Internet 17 November 2003, setelah berbuka puasa!
All Regards,

Abe Poetra
(Just Like Another Noobie.. ☺)