Systemy liczbowe: przeliczanie i działania

Są 4 podstawowe systemy liczbowe:

-Dziesiętny: operujemy nim na codziennie. Składa sie z 10 cyfr: 0123456789. Myślę, ze każdy umie liczyć, wiec nie będę sie rozpisywał,

-Dwójkowy: [Binarny] Podstawowy zbiór liczb, którymi posługuje sie komputer. System składający sie z 2 cyfr. 0 i 1. No więc, jak wiadomo komputer to elektronika. Wymiana informacji polega na odpowiednim przesyłaniu sygnałów prądem elektrycznym, który albo płynie albo nie. Aby łatwiej było komputerowi rozpoznawać sygnały, interpretuje on płynący prąd jako 1, a jego brak jako 0. Komputer operuje tymi sygnałami i przekształca je na czytelne dla nas znaki: obraz, dźwięk itp. Również na płytach są wypalone laserem minimalne zagłębienia, które odpowiadają 1.

-Ósemkowy: [Bitowy] składa sie z liczb od 0 do 7. Bajt to 8 bitów.

-Szesnastkowy: [Heksadecymalny] System 10 liczb i 6 liter. Z zapisem heksadecymalnym można spotkać się miedzy innymi w opisach sprzętu komputerowego [adresie MAC], a także przy analizie i tworzeniu dokumentów HTML, gdzie stanowi sposób zapisu definiowanych kolorów RGB (red/green/blue).

Zaczynamy przeliczanie:

Dziesiętny -> Binarny

Aby przeliczyć z systemu dziesiętnego na binarny należy dzielić nasza liczbę przez 2. Jeżeli otrzymamy liczbę parzysta otrzymujemy 0, nieparzysta 1. Przykład:

321:2 1 [1, bo 321 nie dzieli sie przez 2, zaokrąglamy w dół]
160:2 0 [0, bo 160 dzieli sie przez 2]
80:2 0
40:2 0
20:2 0
10:2 0
5:2 1
2:2 0
1:2 1

Wynik przepisujemy od dołu, końca:

321(10) -> 101000001(2)

Binarny -> Dziesiętny

Teraz mamy odwrotna sytuacje. Aby cokolwiek zrozumieć trzeba by było sie zapoznać z poniższym wypisem. Dla niewiedzących znak ^ znaczy do potęgi:

2^0=1 | 2^1=2 | 2^2=4 | 2^3=8 | 2^4=16 | 2^5=32 | 2^6=64 | 2^7=128 | 2^8=256 | 2^9=512 | 2^10=1024

Pamiętajcie, ze 2^0 jest zawsze równe 1! Przykład:

100101(2) -> (1*2^0)+(0*2^1)+(1*2^2)+(0*2^3)+(0*2^4)+(1*2^5) =
=1 + 4 + 32 = 39

Teraz tak, ostatnia cyfrę z systemu binarnego mnożymy przez 2^0, przedostatnia 2^1, wcześniejsza 2^2 itd... Zawsze 0*2^X równe jest 0. Spróbuje pokazać trochę bliżej:

1 0 0 1 0 1
* * * * * *
2^5 2^4 2^3 2^2 2^1 2^0
= = = = = =
32 0 0 4 0 1 = 39

Jeszcze 1 przykład:

10111(2) -> (1*2^0)+(1*2^1)+(1*2^2)+(0*2^3)+(1*2^4) =
= 1 + 2 + 4 + 16 = 23

Dodawanie i mnożenie liczb binarnych

Działania na liczbach binarnych są naprawdę proste. We wszystkich systemach rachuje sie tak samo. W systemie (10) mamy 100 pozycji, każda liczba z każdą. Tu tylko 4, bo sa 2 liczby .

Tabliczka dodawania
0+0=0
0+1=1
1+0=1
1+1=(1)0

Przy dodawaniu 1+1 mamy dwie możliwości, ponieważ w aktualnej kolumnie zapisujemy 0, a do następnej przenosimy 1. Przykład:

1011 = 11
+0100 = 4

1111 = 15

101 = 5
+101 = 5

1010 = 10

Mnożenie jest dużo prostsze niż w systemie dziesiętnym, ale opiera sie na tych samych zasadach:

Tabliczka mnożenia
0*0=0
0*1=0
1*0=0
1*1=1

Przykład:

0101 = 5
x 0110 = 6

0000
0101
0101
+0000

0011110 = 30

<u>Binarny -> Bitowy</u>

Ósemkowy system liczenia jest 3 krotnie krótszy od binarnego. Zapis binarny dzielimy na 3 cyfry od końca, możemy sobie teraz wstawić linie pomocnicza. Przykład:

101011(2) -> 101|011(2)

Czemu to służy? Teraz przedstawiam tabelkę:

2^2	2^1	2^0	BITY
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

Jak wcześniej tłumaczyłem 2^0 = 1 itd.... aby otrzymać np. 3(8) w systemie dwójkowym należy napisać 011. Porównajcie z tabelka dlaczego. Tym razem nie liczymy od końca. Przykład:

101|111|001(2) -> 571(8)

Jeżeli mamy ilość liczb nie dzieląca sie przez trzy, możemy dla siebie dopisać zera:

011|111|001|101(2) -> 3715(8)