Ejemplos de Conversión entre bases

$243_{10} = ?_2$

$$
\begin{array}{l}
243 \,\lfloor 2 \\
d_0 \quad 1 \quad 121 \,\lfloor 2 \\
\quad d_1 \quad 1 \quad 60 \,\lfloor 2 \\
\quad\quad d_2 \quad 0 \quad 30 \,\lfloor 2 \\
\quad\quad\quad d_3 \quad 0 \quad 15 \,\lfloor 2 \\
\quad\quad\quad\quad d_4 \quad 1 \quad 7 \,\lfloor 2 \\
\quad\quad\quad\quad\quad d_5 \quad 1 \quad 3 \,\lfloor 2 \\
\quad\quad\quad\quad\quad\quad d_6 \quad 1 \quad 1 \,\lfloor 2 \\
\quad\quad\quad\quad\quad\quad\quad d_7 \quad 1 \quad 0
\end{array}
$$

$243_{10} = 1111\ 0011$

$243_{10} = ?_8$

Si tenemos la representación de 243 en binario simplemente agrupamos los dígitos binarios en grupos de 3 $(\lceil \log_2 8 \rceil = \lceil \log_2 2^3 \rceil = 3)$

$243_{10} = \underbrace{11}\underbrace{110}\underbrace{011} = 363_8$

La otra forma es dividiendo entre la base 8

$$
\begin{array}{l}
243 \,\lfloor 8 \\
d_0 \quad 3 \quad 30 \,\lfloor 8 \\
\quad d_1 \quad 6 \quad 3 \,\lfloor 8 \\
\quad\quad d_2 \quad 3 \quad 0
\end{array}
$$

Obtenemos el mismo resultado que nos dio al agrupar los dígitos binarios

$243_{10} = ?_{hex}$

Dado que conocemos su representación en binario, agrupamos los bits (binary digit) de 4 en 4 siempre comenzando del bit menos significativo (derecha)

$243 = \underbrace{1111}\underbrace{0011} = 0XF3$

Si lo resolvemos usando el algoritmo de conversión entre bases sería:

$$
\begin{array}{l}
243 \,\lfloor 16 \\
d_0 \quad 3 \quad 15 \,\lfloor 16 \\
\quad d_1 \quad 15 \quad 0
\end{array}
$$

15 en hexadecimal equivale a F $\quad 243 = 0xF3$

Representación de Números Enteros

Cuál es la representación de 113 en 8 bits si usamos
a) Signo-Magnitud
b) Complemento a Uno
c) Complemento a Dos
d) Exceso de 128

Primero obtenemos la representación de 113 en binario

$$
\begin{array}{l}
113 \,\lfloor 2 \\
d_0 \quad 1 \quad 56 \,\lfloor 2 \\
\quad d_1 \quad 0 \quad 28 \,\lfloor 2
\end{array}
$$

d_2 0 $14 \lfloor 2$

d_3 0 $7 \lfloor 2$

d_4 1 $3 \lfloor 2$

d_5 1 $1 \lfloor 2$

d_6 1 0

113 = 01110001

113 es un número positivo así que se representa de la misma forma para "Signo-Magnitud", "Complemento a Uno" y "Complemento a Dos"

Signo-Magnitud = 01110001 = 0x71
Complemento a Uno = 01110001 = 0x71
Complemento a Dos = 01110001 = 0x71

¿Cómo representamos 113 en exceso de 128?

Para el caso de un número positivo tenemos dos formas para atacar el problema.

1. Obtenemos la representación en binario de (113+ exceso), es decir (113+128) = 241

$241 \lfloor 2$
1 $120 \lfloor 2$
0 60 ... y así sucesivamente

2. La otra forma que sólo aplica si el exceso es una potencia de 2 ($128 = 2^7$) y el número a representar es positivo
En este caso tomamos la representación de 113 y encendemos (pasamos de 0 a 1) la posición que corresponde a la potencia ($128 = 2^7$) posición 7

113_{exceso} = 11110001 = 0xF1

Cuál es la representación de −113 en SM, C1, C2 y exceso 128?

Ya sabemos que 113 en binario es igual a 01110001
En Signo-Magnitud sólo debemos cambiar el bit más significativo

Signo-Magnitud = 11110001 = 0xF1

En complemento a Uno tomamos la representación del equivalente positivo y complementamos cada bit ($0 \rightsquigarrow 1$, $1 \rightsquigarrow 0$)

Complemento a Uno = 10001110 = 0x8E

En complemento a Dos, tomamos la representación del positivo, complementamos cada bit y le sumamos 1

$$\begin{array}{r} 10001110\ + \\ 1 \\ \hline 10001111 \end{array}$$

Complemento a Dos = 10001111 = 0x8F =

Para la representación en exceso debemos calcular la representación en binario de −113+128, es decir, 15

$Exceso_{128}$ = 00001111 = 0x0F

Dado el siguiente patrón binario 10000010 qué número decimal representa si lo interpretamos como Signo-Magnitud, Complemento a Uno, Complemento a Dos, Exceso 128 o Caracter

Signo-Magnitud

Para el caso de Signo-Magnitud, el bit mas significativo representa el signo (1 = Negativo) y el resto corresponde a la Magnitud (000 0010 = 2). Por lo tanto representa −2

Complemento a Uno

Como el bit mas significativo está encendido se trata de un número negativo. Si complementamos el patrón dado obtenemos su complemento (+).

$C1(1000\ 0010) = 0111\ 1101 = 1 + 4 + 8 + 16 + 32 + 64 = 125$

Por lo tanto representa -125

Otra forma es usando la fórmula

$$-(2^{n-1}-1)d_{n-1} + \sum_{i=0}^{n-2} d_i 2^i = -(2^7-1)\cdot 1 + 2 = -127 + 2 = -125$$

Complemento a Dos

Siguiendo un razonamiento similar al de Complemento a Uno tenemos

$C2(1000\ 0010) = 0111\ 1101 + 1 = 0111\ 1110 = 2+4+8+16+32+64$

$C2(1000\ 0010) = 126$

Por lo tanto representa -126

Otra forma es usando la fórmula

$$-(2^{n-1})\cdot d_{n-1} + \sum_{i=0}^{n-2} d_i 2^i = -128\cdot 1 + 2 = -126$$

Exceso 128

Para obtener qué número representa en exceso 128, convertimos el número binario a decimal y le restamos el exceso

$1000\ 0010 = 128 + 2 = 130$

$130 - \text{exceso} = 130 - 128 = 2$

Representa el número 2

Caracter

Necesitamos consultar una tabla ASCII (en este caso extendida).

En la tabla buscamos el hexadecimal 0x82 (1000 0010) o el decimal 130

Representa é