MATEMÁTICA DISCRETA
Teoría de Números

Prof. Sergio Salinas

Facultad de Ingeniería
Universidad Nacional de Cuyo

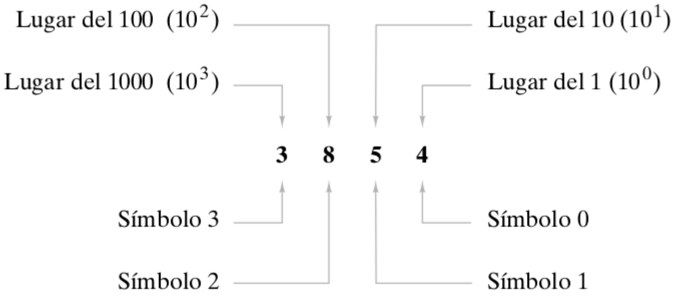
Agosto 2024

Representaciones de enteros

Un bit es un dígito binario, es decir, un 0 o un 1. En una computadora digital, los datos y las instrucciones se codifican como bits. La tecnología determina cómo se representan físicamente los bits dentro del sistema de la computadora. El hardware actual se apoya en el estado de un circuito electrónico para representar un bit. El circuito debe ser capaz de estar en dos estados: uno que representa a 1, y el otro a 0.

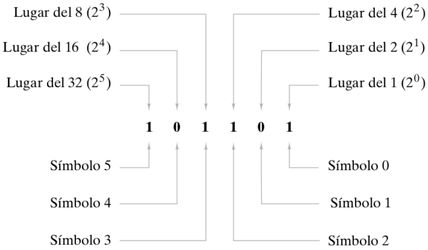
En el sistema numérico decimal, para representar los enteros se usan los 10 símbolos 0, 1, 2, 3, 4, 5, 6, 7, 8 y 9. Al representar un entero, la posición del símbolo es significativa; leyendo desde la derecha, el primer símbolo representa al número de unidades, el siguiente símbolo el número de decenas, el siguiente el número de centenas, y así sucesivamente. Por ejemplo:

$$3854_{10} = 3 \cdot 10^3 + 8 \cdot 10^2 + 5 \cdot 10^1 + 4 \cdot 10^0.$$



En el sistema numérico binario (base 2), para representar enteros se necesitan sólo dos símbolos, 0 y 1. Para representar un entero, leyendo de derecha a izquierda, el primer símbolo representa el número de unos, el siguiente símbolo el número de números dos, el siguiente el número de cuatros, el siguiente el número de ochos, etcétera. Por ejemplo:

$$101101_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0.$$



CONVERSIÓN DE UN NÚMERO EN BASE b A UN NÚMERO DECIMAL

DE BINARIO A DECIMAL

El número binario 101101_2 equivale a un número decimal según la siguiente expresión:

$$\begin{aligned} 101101_2 &= 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 \\ &= 1 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 \\ &= 32 + 8 + 4 + 1 = 45_{10} \end{aligned}$$

DE BASE 3 A DECIMAL

El número 201201_3 en base 3 equivale a un número decimal según la siguiente expresión:

$$\begin{aligned} 201201_3 &= 2 \cdot 3^5 + 0 \cdot 3^4 + 1 \cdot 3^3 + 2 \cdot 3^2 + 0 \cdot 3^1 + 1 \cdot 3^0 \\ &= 2 \cdot 243 + 0 \cdot 81 + 1 \cdot 27 + 2 \cdot 9 + 0 \cdot 3 + 1 \cdot 1 \\ &= 486 + 27 + 18 + 1 = 532_{10} \end{aligned}$$

DE OCTAL A DECIMAL

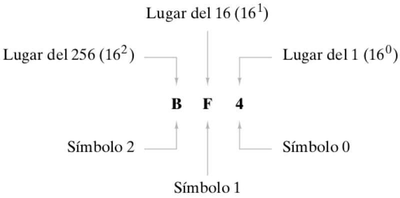
El número en octal 605_8 equivale a un número decimal según la siguiente expresión:

$$\begin{aligned} 605_8 &= 6 \cdot 8^2 + 0 \cdot 8^1 + 5 \cdot 8^0 \\ &= 6 \cdot 64 + 0 \cdot 8 + 5 \cdot 1 \\ &= 384 + 5 = 389_{10} \end{aligned}$$

Sistema Hexadecimal

- En el sistema numérico hexadecimal, para representar enteros se usan los símbolos $0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E$ y F.
- Los símbolos comprendidos entre A y F se interpretan como los decimales 10 al 15.
- En general, en el sistema numérico base N, se requieren N símbolos diferentes que son $(0, 1, 2, \cdots, N-1)$.
- Al representar un entero, si se lee de derecha a izquierda, el primer símbolo representa el número de unos, el siguiente símbolo el número de 16, el siguiente el número de 16^2, y así sucesivamente.

$$B4F_{16} = 11 \cdot 16^2 + 4 \cdot 16^1 + 15 \cdot 16^0.$$


Lugar del 16 (16^1)
Lugar del 256 (16^2)
Lugar del 1 (16^0)
B F 4
Símbolo 2
Símbolo 0
Símbolo 1

DE HEXADECIMAL A DECIMAL

Para convertir el número en hexadecimal $B4F_{16}$ en decimal se deben realizar los siguientes cálculos:

$$\begin{aligned} B4F_{16} &= 11 \cdot 16^2 + 4 \cdot 16^1 + 15 \cdot 16^0 \\ &= 11 \cdot 256 + 4 \cdot 16 + 15 \\ &= 2816 + 64 + 15 = 2895_{10} \end{aligned}$$

Algoritmo de conversión de un entero base b en decimal

Conversión de un entero base b en decimal

Este algoritmo regresa el valor decimal del entero en base b $c_n c_{n-1} \cdots c_1 c_0$.

Entrada: c, n, b

Salida: val_dec

```
base_b_a_dec(c, n, b) {
    val_dec = 0
    potencia = 0
    for i = 0 to n {
        val_dec = val_dec + c_i*potencia
        potencia = potencia*b
    }
    return val_dec
}
```

Importante: la variable potencia se inicializa en 1.

De binario a decimal

Ejercicio

Utilizar el algoritmo presentad previamente para convertir el número 1101_2 en su equivalente en base decimal.
Los parámetros del algoritmo son: $n = 3, b = 2, c_3 = 1, c_2 = 1, c_1 = 0, c_0 = 1$

CONVERSIÓN DE UN NÚMERO DECIMAL A UN NÚMERO EN BASE b

DE DECIMAL A BINARIO

Ejemplo para convertir 130_{10} a su equivalente en binario o base 2.

$$
\begin{array}{cccccl}
n & = & b & q & r & bit \\
130 & = & 2 & 65 & 0 & 2^0 = 1 \\
65 & = & 2 & 32 & 1 & 2^1 = 2 \\
32 & = & 2 & 16 & 0 & 2^2 = 4 \\
16 & = & 2 & 8 & 0 & 2^3 = 8 \\
8 & = & 2 & 4 & 0 & 2^4 = 16 \\
4 & = & 2 & 2 & 0 & 2^5 = 32 \\
2 & = & 2 & 1 & 0 & 2^6 = 64 \\
1 & = & 2 & 0 & 1 & 2^7 = 128
\end{array}
$$

Verificación:

$$130_{10} = 1 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 10000010_2$$

Nota: recordar que $n = b \cdot q + r$.

DE DECIMAL A BASE 3

Ejemplo para convertir 532_{10} a su equivalente en base 3.

$$
\begin{array}{rcrrrl}
n & = & b & q & r & bit \\
532 & = & 3 & 177 & 1 & 3^0 = 1 \\
177 & = & 3 & 59 & 0 & 3^1 = 3 \\
59 & = & 3 & 19 & 2 & 3^2 = 9 \\
19 & = & 3 & 6 & 1 & 3^3 = 27 \\
6 & = & 3 & 2 & 0 & 3^4 = 81 \\
2 & = & 3 & 0 & 2 & 3^5 = 243
\end{array}
$$

Verificación:

$$
\begin{aligned}
201201_3 &= 2 \cdot 3^5 + 0 \cdot 3^4 + 1 \cdot 3^3 + 2 \cdot 3^2 + 0 \cdot 3^1 + 1 \cdot 3^0 \\
&= 2 \cdot 243 + 0 \cdot 81 + 1 \cdot 27 + 2 \cdot 9 + 0 \cdot 3 + 1 \cdot 1 \\
&= 486 + 27 + 18 + 1 = 532_{10}
\end{aligned}
$$

DE DECIMAL A OCTAL

Ejemplo para convertir 513_{10} a su equivalente en octal.

$$\begin{array}{rcrrrl} n & = & b & q & r & bit \\ 513 & = & 8 & 64 & 1 & 8^0 = 1 \\ 64 & = & 8 & 8 & 0 & 8^1 = 8 \\ 8 & = & 8 & 1 & 0 & 8^2 = 64 \\ 1 & = & 8 & 0 & 1 & 8^3 = 512 \end{array}$$

Verificación:

$$\begin{aligned} 1001_8 &= 1 \cdot 8^3 + 0 \cdot 8^2 + 0 \cdot 8^1 + 1 \cdot 8^0 \\ &= 1 \cdot 512 + 0 \cdot 64 + 0 \cdot 8 + 1 \cdot 1 \\ &= 512 + 1 = 513_{10} \end{aligned}$$

DE DECIMAL A HEXADECIMAL

Ejemplo para convertir 20385_{10} a su equivalente en hexadecimal o base 16.

$$\begin{array}{rcrrrl}
n & = & b & q & r & bit \\
20385 & = & 16 & 1274 & 1 & 16^0 = 1 \\
1274 & = & 16 & 79 & 10 & 16^1 = 16 \\
79 & = & 16 & 4 & 15 & 16^2 = 256 \\
4 & = & 16 & 0 & 4 & 16^3 = 4096
\end{array}$$

Verificación:

$$\begin{aligned}
4FA1_{16} &= 4 \cdot 16^3 + 15 \cdot 16^2 + 10 \cdot 16^1 + 1 \cdot 16^0 \\
&= 4 \cdot 4096 + 15 \cdot 256 + 10 \cdot 16 + 1 \cdot 1 \\
&= 16384 + 3840 + 160 + 1 = 20385_{10}
\end{aligned}$$

Conversión de un entero decimal a una base b.

Conversión de un entero decimal a la base b

Este algoritmo convierte el entero positivo m en un entero base b $c_n c_{n-1} \cdots c_1 c_0$. La variable n se usa como índice en la sucesión c. El valor de $m \bmod b$ es el residuo cuando m se divide entre b. El valor de $\lfloor m/b \rfloor$ es el cociente cuando m se divide entre b.

Entrada: m, b

Salida: c, n

```
dec_a_base_b(m, b, c, n) {
    n = -1
    while(m > 0){
        n = n + 1
        c_n = m mod b
        m = ⌊m/b⌋
    }
}
```

Suma de números binarios

Sume los números binarios 10011011 y 1011011.

$$\begin{array}{r} 10011011 \\ +\quad 1011011 \\ \hline 11110110 \end{array}$$

Suma hexadecimal

Sumar los números hexadecimales $84F_{16}$ y $42EA_{16}$.

$$\begin{array}{rr} & 84F \\ + & 42EA \\ \hline & 4B39 \end{array}$$

1. Se comienza por la columna de la derecha sumando A y F. Como F es 15_{10} y A es 10_{10}, $F + A = 15_{10} + 10_{10} = 25_{10} = 19_{16}$. Se escribe 9 y llevamos 1.
2. Después se suman 1, 4 y E, para obtener 13_{16}. Se escribe 3 y llevamos 1.

Fin