

Sistemas de numeración

Arquitectura de Computadoras - Práctico 0

1

Conversión de base de números enteros

Tenemos un número entero N representado en una base B :

$$N = A_n B^n + \dots + A_0 B^0$$

y queremos hallar su expresión en la base b , es decir encontrar los valores a_m, a_{m-1}, \dots, a_0 tal que:

$$N = a_m b^m + \dots + a_0 b^0$$

(B → b) Usando la aritmética de la base b

muy útil para pasar de cualquier base a base 10

expreso símbolos en base b (10):

- $A_{16} \rightarrow 10_{10}$
- $2_{16} \rightarrow 2_{10}$
- $F_{16} \rightarrow 15_{10}$

a través del polinomio característico, expresando los símbolos y la base B en la base b, y usando la aritmética de la base b

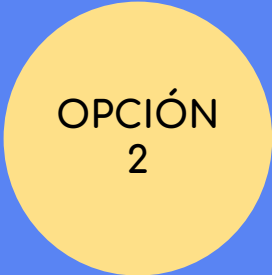
- Convertir A2Fh a decimal

$$A2Fh = 10 \times 16^2 + 2 \times 16^1 + 15 \times 16^0 = 2607$$

expreso base B (10_{16}) en la base b (16_{10})

OPCIÓN
1

Usando la aritmética de la base B



muy útil para pasar de base 10 a cualquier base

los valores a_0, \dots, a_n son los restos de las divisiones de N entre b realizadas en la aritmética de la base B .

- Convertir 653 a base 5

$653 \mid 5$
 $130 \mid 5$
 $26 \mid 5$
 $5 \mid 5$
 1

Restos: 3, 0, 1, 0, 1

$a_0 \leftarrow 3$

\rightarrow cociente $< b$

$\rightarrow a_n$

$\Rightarrow 653 = 10103_5$

Conversión de números con parte fraccionaria

$$\text{Sea } N = N_e + N_f = a_n b^n + \dots + a_1 b^1 + a_0 + a_{-1} b^{-1} + \dots$$

parte
entera

parte
fraccionaria

la parte entera puede convertirse igual que antes y la parte fraccionaria se convierte por separado



Usando la aritmética de la base b

OPCIÓN
1

muy útil para pasar de cualquier base a base 10

tenemos $N_f = A_{-1}B^{-1} + A_{-2}B^{-2} + \dots + A_{-m}B^{-m}$ y desarrollamos el polinomio equivalente, $P(x)$, obteniendo su valor numérico

- Convertir $0,213_8$ a decimal

$$\text{tenemos } N = 2x8^{-1} + 1x8^{-2} + 3x8^{-3} \Rightarrow N = 2x(8^{-1})^1 + 1x(8^{-1})^2 + 3x(8^{-1})^3$$

El valor numérico para $8^{-1} = \frac{1}{8} = 0,125$

$$\Rightarrow P(x) = 2.(0,125) + (0,125)^2 + 3.(0,125)^3 = 0,27148..$$

Usando la aritmética de la base B

OPCIÓN
2

muy útil para pasar de base 10 a cualquier base

tenemos $N_f = a_{-1}b^{-1} + a_{-2}b^{-2} + \dots + a_{-m}b^{-m}$ y multiplicamos por b

$\Rightarrow N_f \cdot b = a_{-1} + a_{-2}b^{-1} + \dots$ donde a_{-1} es la parte entera de $N_f \cdot b$

- Convertir 653.61 a base 2

$$2(0,61) = 1,22 \Rightarrow a_{-1} = 1 \quad 2(0,88) = 1,76 \Rightarrow a_{-4} = 1$$

$$2(0,22) = 0,44 \Rightarrow a_{-2} = 0 \quad 2(0,76) = 1,52 \Rightarrow a_{-5} = 1$$

$$2(0,44) = 0,88 \Rightarrow a_{-3} = 0$$

$$653 = 1010001101b$$

$$\Rightarrow 653.61 =$$

$$1010001101.10011\dots b$$

Representación exacta y aproximada

El proceso de conversión de números entre bases no necesariamente es exacto

Al convertir de una base a la otra puede ocurrir que se requiera un número infinito de dígitos para representar el número en la nueva base

En estos casos es necesario definir un **criterio de parada**

Sistemas de numeración y el lenguaje C

Los lenguajes de programación posibilitan definir constantes numéricas en distintas bases

Hexadecimal

- prefijo 0x
- Ejemplo: 0x10

Decimal

- no requiere prefijo
- Ejemplo: 10000

Octal

- prefijo 0
- Ejemplo: 020

Ejercicio 2 d) y 2 e)

- Convertir a base 10 los siguientes números:

d) DB_{16}

e) 111110_2

Ejercicios 1 b) y 5 a)

- Realizar las siguientes conversiones:

1 b) 63_{10} a base 2

5 a) 100110111011_2 a hexadecimal

Ejercicios de práctico

- Ejercicios 2 d) y 2 e):
 - d) DB_{16} a base 10
 - e) 111110_2 a base 10
- Ejercicio 1 b):
 - b) 63_{10} a base 2
- Ejercicio 5 a):
 - a) 100110111011_2 a hexadecimal