Universidad Nacional de Rosario

Facultad de Ciencias Exactas, Ingeniería y Agrimensura

Arquitectura del computador

C y sistemas de numeración posicionales

Alumnos:

Cavagna, Lucas Gastón
Demagistris, Santiago Ignacio

Septiembre 2020

1 Ejercicio 1

A continuación se presentan ciertos números enteros expresados en binario utilizando 32 bits y a su derecha, expresiones en lenguaje C incompletas. Complete estas expresiones de forma que la igualdad sea cierta. Utilice operadores de bits, operadores enteros y constantes de enteros literales según considere necesario.

a) 10000000 00000000 00000000 00000000 == 1 << 31

Sabemos que $(1)_{10} \simeq (00000000\ 00000000\ 00000000\ 00000001)_2$.

Observemos que $1 << 1 \simeq (00000000\ 00000000\ 00000000\ 00000010)_2$.

Es decir el bit menos significativo pasa a ocupar el lugar del bit 1, agregándose un 0 a la izquierda. Por lo tanto para $1 << n$, el bit menos significativo pasa a ocupar el lugar del bit n, en particular para n = 31:

$(00000000\ 00000000\ 00000000\ 00000001)_2 << 31 = (10000000\ 00000000\ 00000000\ 00000000)_2$

b) 10000000 00000000 10000000 00000000 == (1 << 31) | (1 << 15)

Sabemos que:

$(1\ <<\ 31) \simeq (10000000\ 00000000\ 00000000\ 00000000)_2$

$(1\ <<\ 15) \simeq (00000000\ 00000000\ 10000000\ 00000000)_2$

Por lo tanto:

$$\begin{array}{rl} & 10000000\ 00000000\ 00000000\ 00000000 \\ OR & 00000000\ 00000000\ 10000000\ 00000000 \\ \hline & 10000000\ 00000000\ 10000000\ 00000000 \end{array}$$

c) 11111111 11111111 11111111 00000000 == $-1\ \&-256$

Sabemos que:

$C_2^{-1} \quad = (11111111\ 11111111\ 11111111\ 11111111)_2,$

$C_2^{-256} = (11111111\ 11111111\ 11111111\ 00000000)_2$

Por lo tanto:

$$\begin{array}{rl} & 11111111\ 11111111\ 11111111\ 11111111 \\ AND & 11111111\ 11111111\ 11111111\ 00000000 \\ \hline & 11111111\ 11111111\ 11111111\ 00000000 \end{array}$$

d) 10101010 00000000 00000000 10101010 == $0xAA \mid (0xAA\ \ <<\ \ 24)$

$0xAA \rightarrow binary$:

Hexadecimal	A	A
Binario	1010	1010

$$0xAA \simeq (00000000\ 00000000\ 00000000\ 10101010)_2,$$

$$0xAA\ \ <<\ \ 24 \simeq (10101010\ 00000000\ 00000000\ 00000000)_2$$

Por lo tanto

$$\begin{array}{rl} & 00000000\ 00000000\ 00000000\ 10101010 \\ OR & 10101010\ 00000000\ 00000000\ 00000000 \\ \hline & 10101010\ 00000000\ 00000000\ 10101010 \end{array}$$

e) 00000000 00000000 00000101 00000000 == $5\ \ <<\ \ 8$

Sabemos que:

$$(5)_{10} \simeq (00000000\ 00000000\ 00000000\ 00000101)_2$$

Por lo tanto:

$$(5\ \ <<\ \ 8) \simeq (00000000\ 00000000\ 00000101\ 00000000)_2$$

f) 11111111 11111111 11111110 11111111 == $-1\ \&\ (not\ (1 << 8))$

Sabemos que:

$$(-1)_{10} \simeq (11111111\ 11111111\ 11111111\ 11111111)_2$$

$$1\ \ <<\ \ 8 \simeq (00000000\ 00000000\ 00000001\ 00000000)_2$$

Por lo tanto:

$$\begin{array}{rl} NOT & 00000000\ 00000000\ 00000001\ 00000000 \\ \hline & 11111111\ 11111111\ 11111110\ 11111111 \end{array}$$

y así:

$$\begin{array}{rl} & 11111111\ 11111111\ 11111111\ 11111111 \\ AND & 11111111\ 11111111\ 11111110\ 11111111 \\ \hline & 11111111\ 11111111\ 11111110\ 11111111 \end{array}$$

g) 11111111 11111111 11111111 11111111 == 0 − 1

$(0)_{10} - (1)_{10} = (-1)_{10} \simeq (11111111\ 11111111\ 11111111\ 11111111)_2$

h) 00000000 00000000 00000000 00000000 == $0x80000000 + 0x80000000$

$0x80000000 \rightarrow binary$:

Hexadecimal	8	0	0	0	0	0	0	0
Binario	1000	0000	0000	0000	0000	0000	0000	0000

Por lo tanto

$$\begin{array}{rl} & |\ 10000000\ 00000000\ 00000000\ 00000000 \\ \mathrm{A}ND & |\ 10000000\ 00000000\ 00000000\ 00000000 \\ \hline & 1\ 00000000\ 00000000\ 00000000\ 00000000 \end{array}$$

Pero como la representación es de 32 bits obtenemos como resultado = $(00000000\ 00000000\ 00000000\ 00000000)_2$