

Programación de Computadores

Tema 9: Operaciones a nivel de bits



Carrera Ingeniería Civil en Informática
y Ciencias de la Computación
Universidad de Concepción

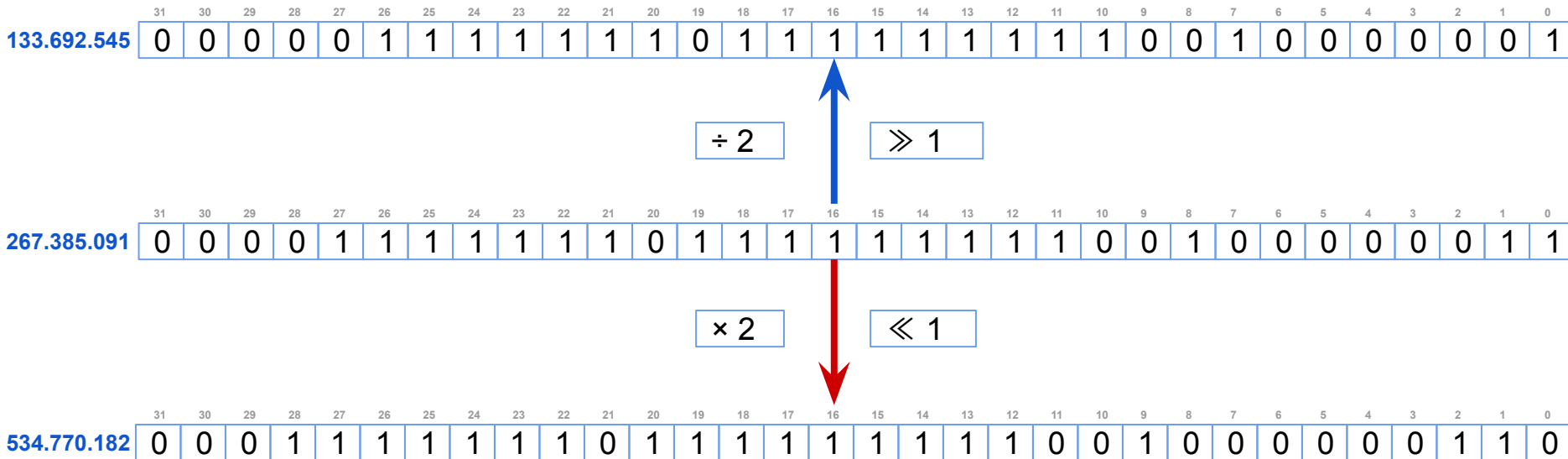
José Fuentes - jfuentess@inf.udec.cl

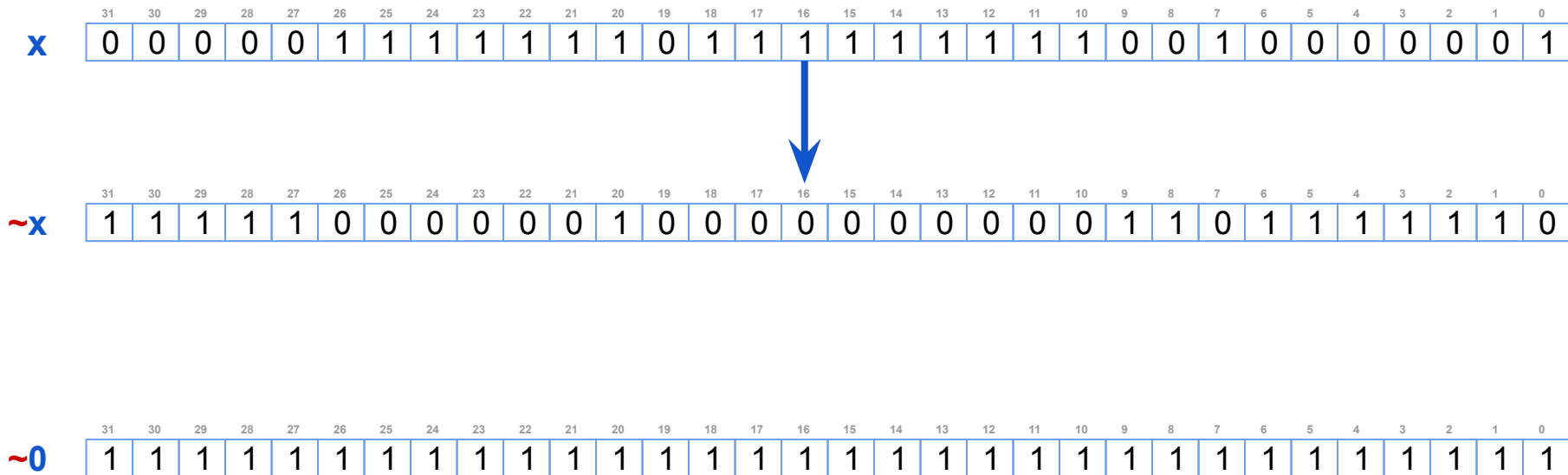
Bit shifting

— — —

$$n \times 2^k \Leftrightarrow n \ll k$$

$$n \div 2^k \Leftrightarrow n \gg k$$



$$1 \rightarrow 0$$


Operador AND (&)

-- --

Operador &

0 **AND** 0 → 0

0 **AND** 1 → 0

1 **AND** 0 → 0

1 **AND** 1 → 1

X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	0	0	1	0	0	0	0	0	0	0	1

Y

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	1	0	1	1	1	0	1	1	0	1	0	1	1	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	1	1

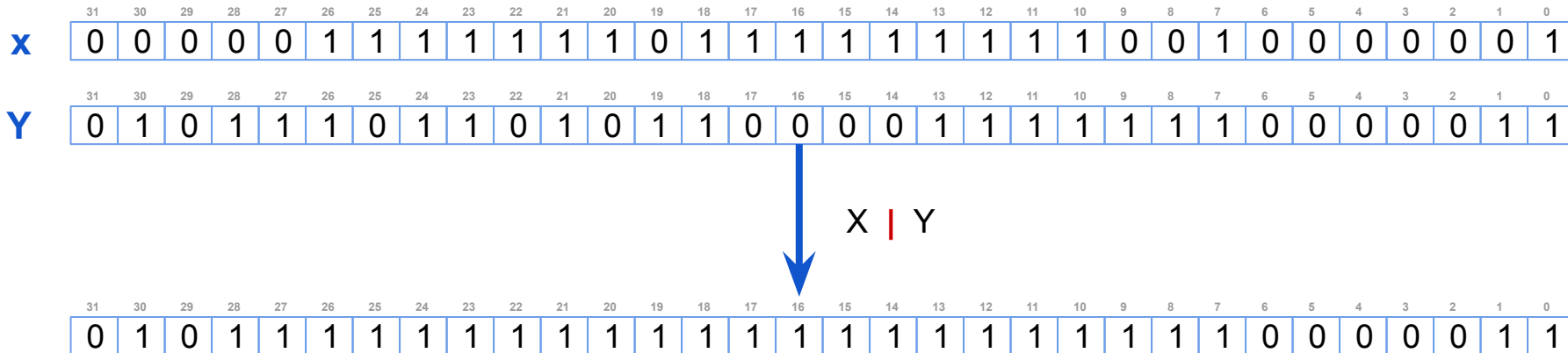
X & Y

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	1	0	1	1	0	1	0	0	1	0	0	0	0	1	1	1	1	0	0	1	0	0	0	0	0	0	1

Operador OR (|)

-- --

Operador		
0	OR 0	→ 0
0	OR 1	→ 1
1	OR 0	→ 1
1	OR 1	→ 1



Operador XOR (^)

— — —

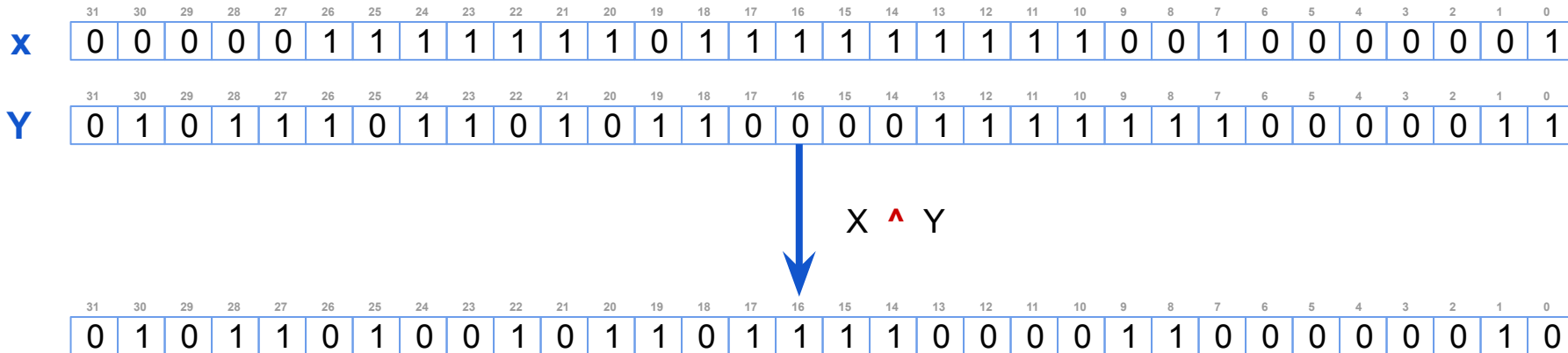
Operador ^

0 **XOR** 0 → 0

0 **XOR** 1 → 1

1 **XOR** 0 → 1

1 **XOR** 1 → 0



Ejercicio: Empaquetar 4 variables char en un entero

— — —

X = 10

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

(int) x

10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

10

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

155

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1	1	0	1	1

230

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	1	1	0

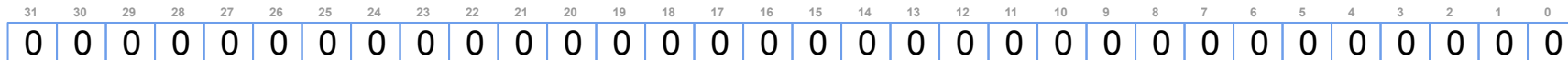
17

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

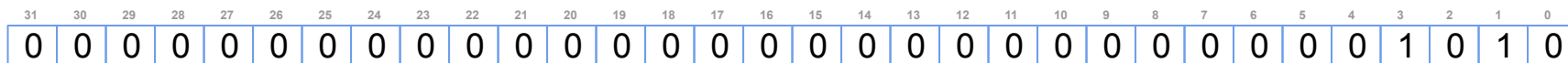
Ejercicio: Empaquetar 4 variables char en un entero (cont.)

-- --

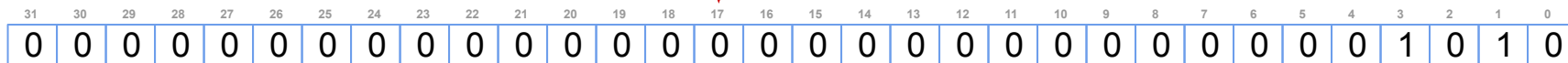
res



10



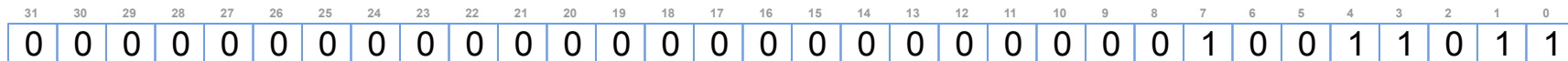
res



Ejercicio: Empaquetar 4 variables char en un entero (cont.)

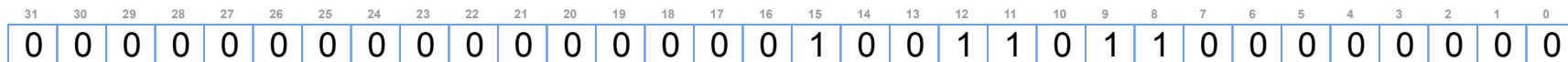
-- --

155

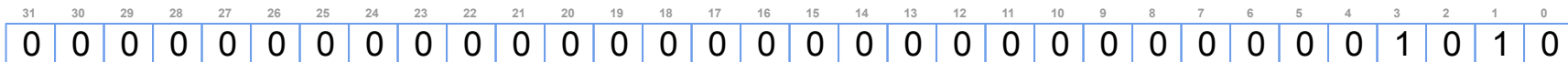


« 8

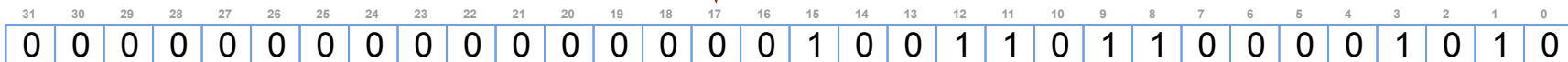
155 « 8



res



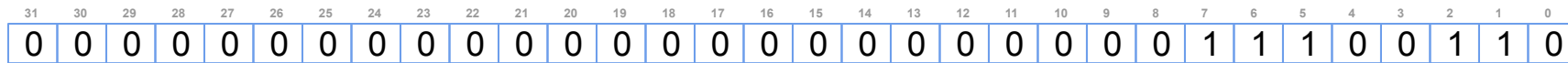
res



Ejercicio: Empaquetar 4 variables char en un entero (cont.)

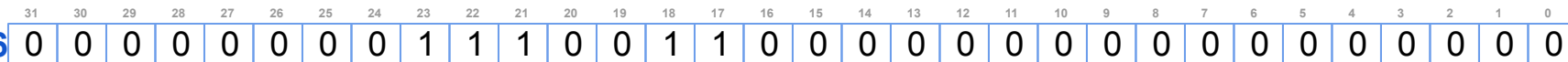
-- --

230

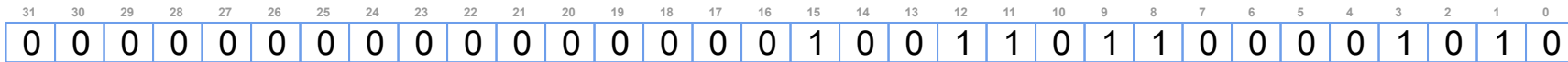


« 16

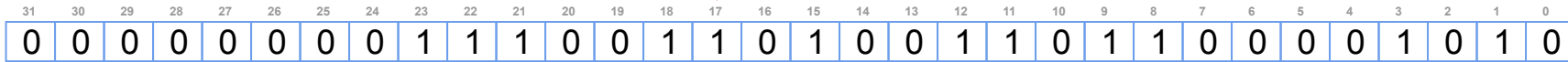
230 « 16



res



res



Ejercicio: Empaquetar 4 variables char en un entero (cont.)

-- --

17

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1

« 24

17 « 24

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

res

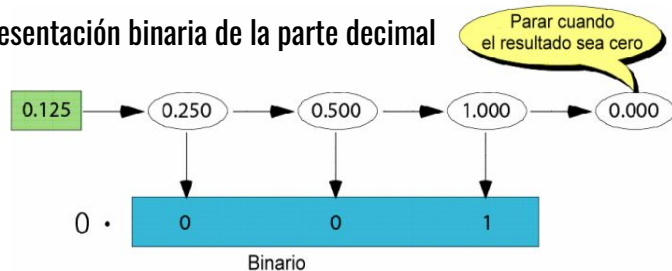
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	1	1	1	0	0	1	1	0	1	0	0	1	1	0	1	1	0	0	0	0	1	0	1	0

res

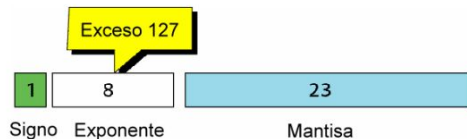
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	0	0	1	0	0	0	1	1	1	1	0	0	1	1	0	1	0	0	1	1	0	1	1	0	0	0	0	1	0	1	0

Recordatorio: Representación de números punto flotante

Representación binaria de la parte decimal



Estándar IEEE (Institute of Electrical and Electronics Engineers)



a. Precisión simple



b. Precisión doble

Representación normalizada

Número original	Desplazamiento	Normalizado
+ 1010001.11001	← 6	$+2^{+6} \times 1.01000111001$
-111.000011	← 2	$-2^{+2} \times 1.11000011$
+0.00000111001	6 →	$+2^{-6} \times 1.11001$
-0.001110011	3 →	$-2^{-3} \times 1.110011$

Ejemplos estándar IEEE

Número	Signo	Exponente	Mantisa
$-2^2 \times 1.11000011$	1	10000001	110000110000000000000000
$+2^{-6} \times 1.11001$	0	01111001	110010000000000000000000
$-2^{-3} \times 1.110011$	1	01111100	110011000000000000000000

¡A practicar!

— — —

Ejemplo 1:
[operadores.c](#)

Ejemplo 2:
[potencias_de_2.c](#)

Ejemplo 3:
[compactar.c](#)

Ejemplo 4:
[diccionario.c](#)