



Tratamiento de Señales

Version 2022-I

Transformada Discreta de Cosenos (DCT)

[Capítulo 4]

Dr. José Ramón Iglesias

DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingeniería Electronica
Universidad Popular del Cesar

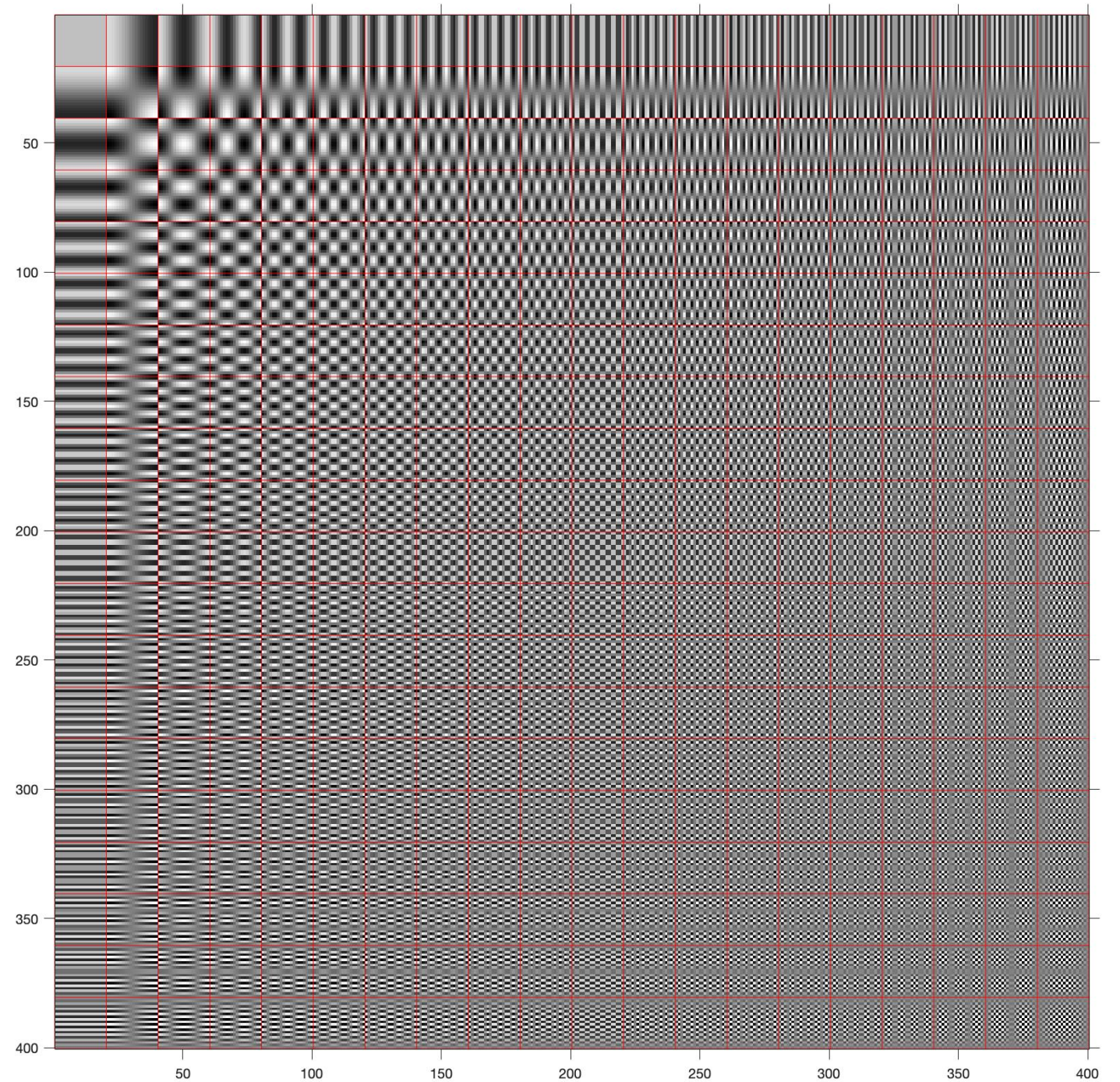
La Transformada Discreta de Cosenos está definida como:

$$D(m, n) = \alpha_m \alpha_n \sum_{i=1}^N \sum_{k=1}^N X(i, k) \cos \left(\frac{\pi(2i-1)(m-1)}{2N} \right) \cos \left(\frac{\pi(2k-1)(n-1)}{2N} \right)$$

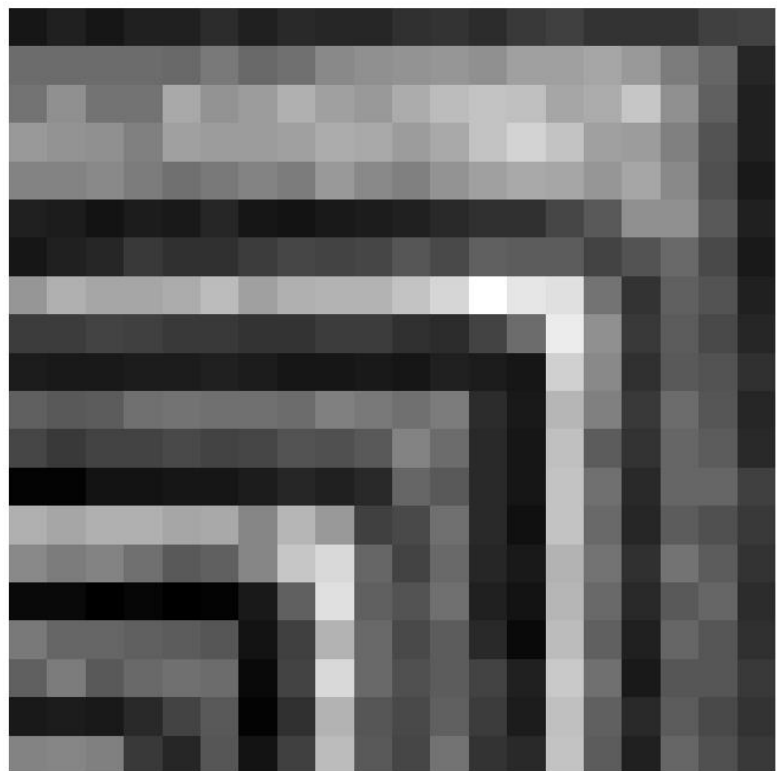
$$\alpha_1 = 1/\sqrt{N} \qquad \alpha_m = \sqrt{2/N} \qquad \text{para } m=2, \dots, N$$

A diferencia de la DFT, esta transformada no presenta números complejos, sólo reales.

Funciones Base:
(para imágenes de 20x20 pixels)



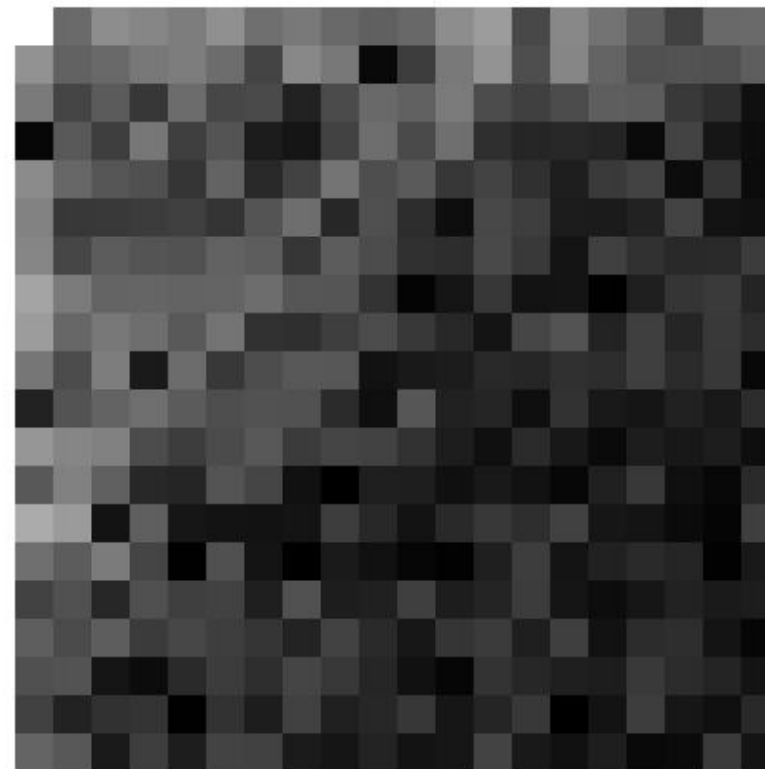
X: Input Image



20 x 20 pixels



$D = \text{DCT2}(X)$



20 x 20 pixels

X: Input Image

40	43	40	43	43	47	43	46	45	45	48	49	47	51	53	49	49	49	53	54
67	67	67	67	66	71	66	68	76	78	79	80	78	83	83	85	81	72	65	45
69	78	69	69	86	79	82	88	83	81	87	92	94	93	85	87	95	78	63	43
81	79	78	73	83	82	82	83	87	86	82	86	94	99	94	83	82	73	59	43
74	74	76	72	68	71	74	72	81	76	73	79	83	86	85	80	85	76	58	41
43	42	39	42	41	45	40	39	41	42	43	46	48	48	55	61	78	78	61	43
40	43	45	51	48	48	52	55	54	55	60	56	63	62	62	54	59	66	56	41
80	88	85	85	87	92	83	88	89	89	94	100	113	105	103	69	49	63	59	43
52	52	54	53	51	51	49	49	52	52	48	47	54	67	107	78	51	62	56	45
42	41	41	42	42	43	42	40	40	41	40	43	42	40	98	76	48	61	59	48
63	61	62	68	69	68	68	67	73	71	68	72	47	41	90	73	51	67	60	45
55	51	54	54	56	54	55	59	58	61	74	67	46	40	93	62	49	65	62	46
34	34	39	39	40	40	42	45	43	46	65	61	46	40	94	68	46	65	65	53
88	85	88	88	85	86	75	90	81	53	56	68	46	38	94	66	45	62	58	51
76	72	74	68	61	63	75	95	101	65	54	66	45	41	89	69	48	69	62	49
36	36	33	35	33	34	41	63	103	63	59	68	43	39	90	66	46	61	65	47
71	65	65	63	62	60	39	53	89	66	56	62	46	36	92	63	43	65	60	48
63	72	61	66	68	67	36	54	101	66	58	62	54	43	96	68	41	60	60	51
41	42	41	46	54	61	34	48	89	60	56	63	52	42	93	63	46	62	56	49
74	75	73	51	45	60	39	53	92	61	55	69	49	46	94	62	43	65	58	51

20 x 20 pixels



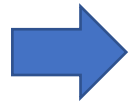
$D = \text{DCT2}(X)$

1245	-20	-53	44	-34	58	-23	-31	19	15	-19	52	-81	7	46	-26	13	-6	-19	20
61	-15	-19	29	-34	-22	6	46	-29	0	-5	-30	63	-9	-43	17	-9	10	9	-14
31	7	-13	4	21	7	-8	-2	-8	19	15	-32	-8	-6	8	14	-13	-4	3	-1
0	12	5	-29	5	10	-1	1	6	-22	7	23	-3	-2	-2	-2	0	6	1	-1
-50	-18	11	9	-4	-16	2	-6	26	-9	-12	4	-6	3	2	-5	6	-1	-4	-1
-39	-4	-4	-5	6	-4	10	-22	2	9	3	1	-7	5	-1	-1	-2	-6	-1	-1
-44	7	13	-10	10	-15	13	4	-13	-8	-3	3	7	-4	1	5	3	-2	-3	-5
-101	-32	17	-17	16	-16	-21	11	11	4	0	-1	4	1	-1	0	-2	-4	-4	2
-81	-18	29	-22	12	26	-3	-3	-5	-8	4	2	1	7	10	-2	-5	2	4	-3
-28	8	34	1	-21	-4	8	-11	11	-1	1	-1	-2	2	3	-3	5	-3	-4	0
2	-10	-16	23	13	-9	10	-9	3	1	-11	2	2	1	3	1	1	2	1	3
68	44	-38	-8	5	-8	12	-5	-7	6	3	-2	-1	-3	-1	0	-1	-1	1	-1
11	38	15	-2	2	-10	7	1	0	-2	2	-1	-1	-1	0	-2	4	1	0	-3
-122	-79	-1	-14	-1	1	-1	1	5	-2	1	2	4	-3	-6	-1	-1	1	0	5
-22	13	32	-6	0	9	1	0	-1	-1	0	0	2	-5	-1	-2	3	2	0	-1
6	9	-2	-9	-5	-6	-2	9	1	2	5	-1	-2	-5	1	-1	-1	2	-1	-1
14	-8	-13	5	7	-5	4	2	-6	-3	1	-4	-5	-2	-5	-1	-3	-3	1	0
9	10	1	1	-3	-5	3	6	-4	-2	1	0	4	-2	-2	-1	4	3	-2	1
-6	2	-3	-4	0	3	-1	-6	-2	-2	4	1	-2	-4	0	1	-5	-1	-1	2
-17	-12	1	-5	-2	6	-6	-1	-1	-2	-1	1	6	1	-1	-2	0	1	5	-1

20 x 20 pixels

Transformada DCT2 de X

X
Input Image



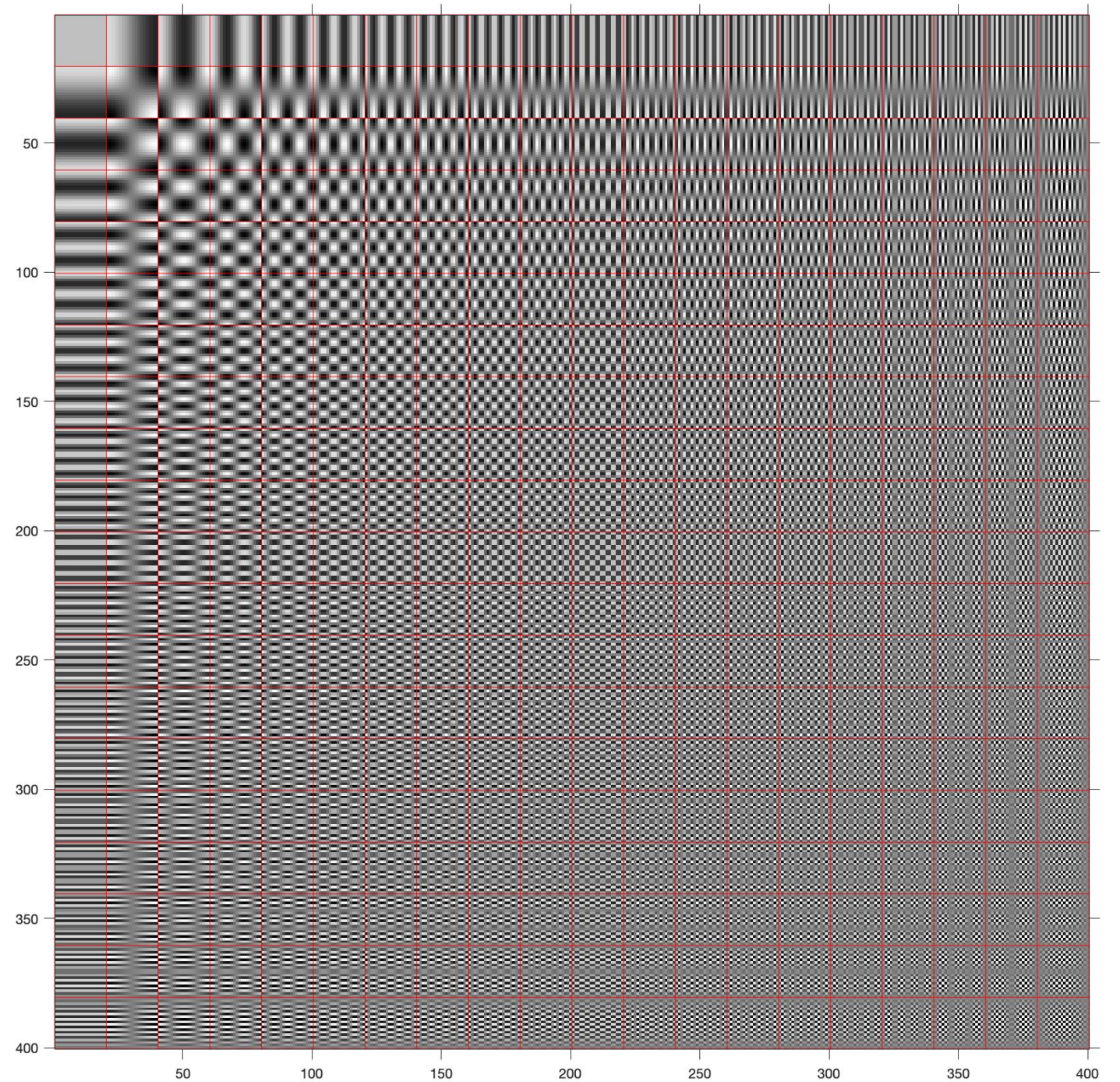
DCT2

$D = \text{DCT2}(X)$



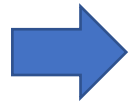
1245	-20	-53	44	-34	58	-23	-31	19	15	-19	52	-81	7	46	-26	13	-6	-19	20
61	-15	-19	29	-34	-22	6	46	-29	0	-5	-30	63	-9	-43	17	-9	10	9	-14
31	7	-13	4	21	7	-8	-2	-8	19	15	-32	-8	-6	8	14	-13	-4	3	-1
0	12	5	-29	5	10	-1	1	6	-22	7	23	-3	-2	-2	-2	0	6	1	-1
-50	-18	11	9	-4	-16	2	-6	26	-9	-12	4	-6	3	2	-5	6	-1	-4	-1
-39	-4	-4	-5	6	-4	10	-22	2	9	3	1	-7	5	-1	-1	-2	-6	-1	-1
-44	7	13	-10	10	-15	13	4	-13	-8	-3	3	7	-4	1	5	3	-2	-3	-5
-101	-32	17	-17	16	-16	-21	11	11	4	0	-1	4	1	-1	0	-2	-4	-4	2
-81	-18	29	-22	12	26	-3	-3	-5	-8	4	2	1	7	10	-2	-5	2	4	-3
-28	8	34	1	-21	-4	8	-11	11	-1	1	-1	-2	2	3	-3	5	-3	-4	0
2	-10	-16	23	13	-9	10	-9	3	1	-11	2	2	1	3	1	1	2	1	3
68	44	-38	-8	5	-8	12	-5	-7	6	3	-2	-1	-3	-1	0	-1	-1	1	-1
11	38	15	-2	2	-10	7	1	0	-2	2	-1	-1	-1	0	-2	4	1	0	-3
-122	-79	-1	-14	-1	1	-1	1	5	-2	1	2	4	-3	-6	-1	-1	1	0	5
-22	13	32	-6	0	9	1	0	-1	-1	0	0	2	-5	-1	-2	3	2	0	-1
6	9	-2	-9	-5	-6	-2	9	1	2	5	-1	-2	-5	1	-1	-1	2	-1	-1
14	-8	-13	5	7	-5	4	2	-6	-3	1	-4	-5	-2	-5	-1	-3	-3	1	0
9	10	1	1	-3	-5	3	6	-4	-2	1	0	4	-2	-2	-1	4	3	-2	1
-6	2	-3	-4	0	3	-1	-6	-2	-2	4	1	-2	-4	0	1	-5	-1	-1	2
-17	-12	1	-5	-2	6	-6	-1	-1	-2	-1	1	6	1	-1	-2	0	1	5	-1

Funciones Base:
(para imágenes de 20x20 pixels)



Transformada DCT2 de X

X
Input Image




DCT2

$D = \text{DCT2}(X)$



1245	-20	-53	44	-34	58	-23	-31	19	15	-19	52	-81	7	46	-26	13	-6	-19	20
61	-15	-19	29	-34	-22	6	46	-29	0	-5	-30	63	-9	-43	17	-9	10	9	-14
31	7	-13	4	21	7	-8	-2	-8	19	15	-32	-8	-6	8	14	-13	-4	3	-1
0	12	5	-29	5	10	-1	1	6	-22	7	23	-3	-2	-2	-2	0	6	1	-1
-50	-18	11	9	-4	-16	2	-6	26	-9	-12	4	-6	3	2	-5	6	-1	-4	-1
-39	-4	-4	-5	6	-4	10	-22	2	9	3	1	-7	5	-1	-1	-2	-6	-1	-1
-44	7	13	-10	10	-15	13	4	-13	-8	-3	3	7	-4	1	5	3	-2	-3	-5
-101	-32	17	-17	16	-16	-21	11	11	4	0	-1	4	1	-1	0	-2	-4	-4	2
-81	-18	29	-22	12	26	-3	-3	-5	-8	4	2	1	7	10	-2	-5	2	4	-3
-28	8	34	1	-21	-4	8	-11	11	-1	1	-1	-2	2	3	-3	5	-3	-4	0
2	-10	-16	23	13	-9	10	-9	3	1	-11	2	2	1	3	1	1	2	1	3
68	44	-38	-8	5	-8	12	-5	-7	6	3	-2	-1	-3	-1	0	-1	-1	1	-1
11	38	15	-2	2	-10	7	1	0	-2	2	-1	-1	-1	0	-2	4	1	0	-3
-122	-79	-1	-14	-1	1	-1	1	5	-2	1	2	4	-3	-6	-1	-1	1	0	5
-22	13	32	-6	0	9	1	0	-1	-1	0	0	2	-5	-1	-2	3	2	0	-1
6	9	-2	-9	-5	-6	-2	9	1	2	5	-1	-2	-5	1	-1	-1	2	-1	-1
14	-8	-13	5	7	-5	4	2	-6	-3	1	-4	-5	-2	-5	-1	-3	-3	1	0
9	10	1	1	-3	-5	3	6	-4	-2	1	0	4	-2	-2	-1	4	3	-2	1
-6	2	-3	-4	0	3	-1	-6	-2	-2	4	1	-2	-4	0	1	-5	-1	-1	2
-17	-12	1	-5	-2	6	-6	-1	-1	-2	-1	1	6	1	-1	-2	0	1	5	-1





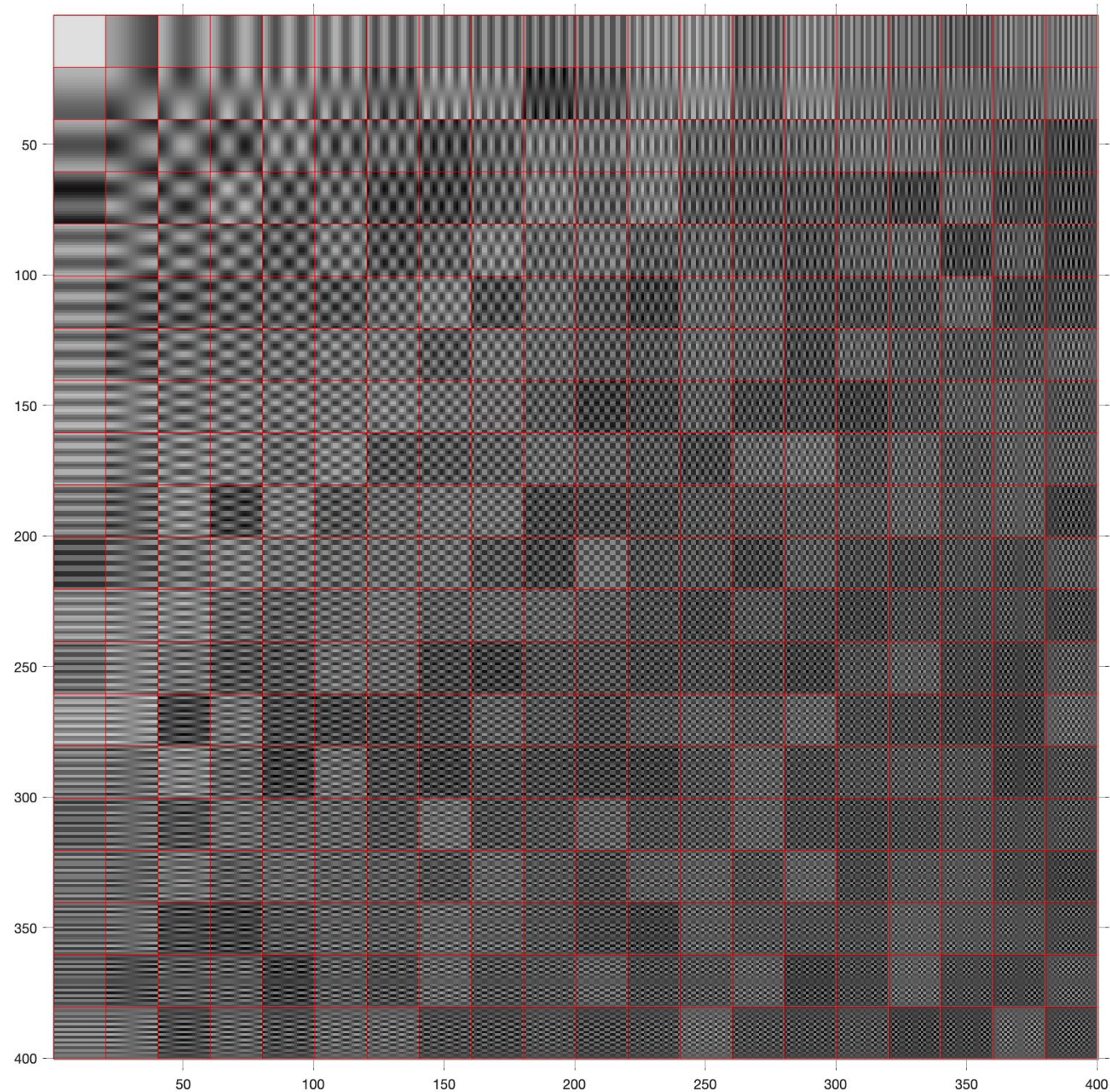
	1245	-20	-53	44	-34	58	-23	-31	19	15	-19	52	-81	7	46	-26	13	-6	-19	20
	61	-15	-19	29	-34	-22	6	46	-29	0	-5	-30	63	-9	-43	17	-9	10	9	-14
50	31	7	-13	4	21	7	-8	-2	-8	19	15	-32	-8	-6	8	14	-13	-4	3	-1
	0	12	5	-29	5	10	-1	1	6	-22	7	23	-3	-2	-2	-2	0	6	1	-1
100	-50	-18	11	9	-4	-16	2	-6	26	-9	-12	4	-6	3	2	-5	6	-1	-4	-1
	-39	-4	-4	-5	6	-4	10	-22	2	9	3	1	-7	5	-1	-1	-2	-6	-1	-1
	-44	7	13	-10	10	-15	13	4	-13	-8	-3	3	7	-4	1	5	3	-2	-3	-5
150	-101	-32	17	-17	16	-16	-21	11	11	4	0	-1	4	1	-1	0	-2	-4	-4	2
	-81	-18	29	-22	12	26	-3	-3	-5	-8	4	2	1	7	10	-2	-5	2	4	-3
	-28	8	34	1	-21	-4	8	-11	11	-1	1	-1	-2	2	3	-3	5	-3	-4	0
200	2	-10	-16	23	13	-9	10	-9	3	1	-11	2	2	1	3	1	1	2	1	3
	68	44	-38	-8	5	-8	12	-5	-7	6	3	-2	-1	-3	-1	0	-1	-1	1	-1
250	11	38	15	-2	2	-10	7	1	0	-2	2	-1	-1	-1	0	-2	4	1	0	-3
	-122	-79	-1	-14	-1	1	-1	1	5	-2	1	2	4	-3	-6	-1	-1	1	0	5
	-22	13	32	-6	0	9	1	0	-1	-1	0	0	2	-5	-1	-2	3	2	0	-1
300	6	9	-2	-9	-5	-6	-2	9	1	2	5	-1	-2	-5	1	-1	-1	2	-1	-1
	14	-8	-13	5	7	-5	4	2	-6	-3	1	-4	-5	-2	-5	-1	-3	-3	1	0
350	9	10	1	1	-3	-5	3	6	-4	-2	1	0	4	-2	-2	-1	4	3	-2	1
	-6	2	-3	-4	0	3	-1	-6	-2	-2	4	1	-2	-4	0	1	-5	-1	-1	2
400	-17	-12	1	-5	-2	6	-6	-1	-1	-2	-1	1	6	1	-1	-2	0	1	5	-1
			50		100		150		200		250		300		350		400			

X
Input Image



=

Σ

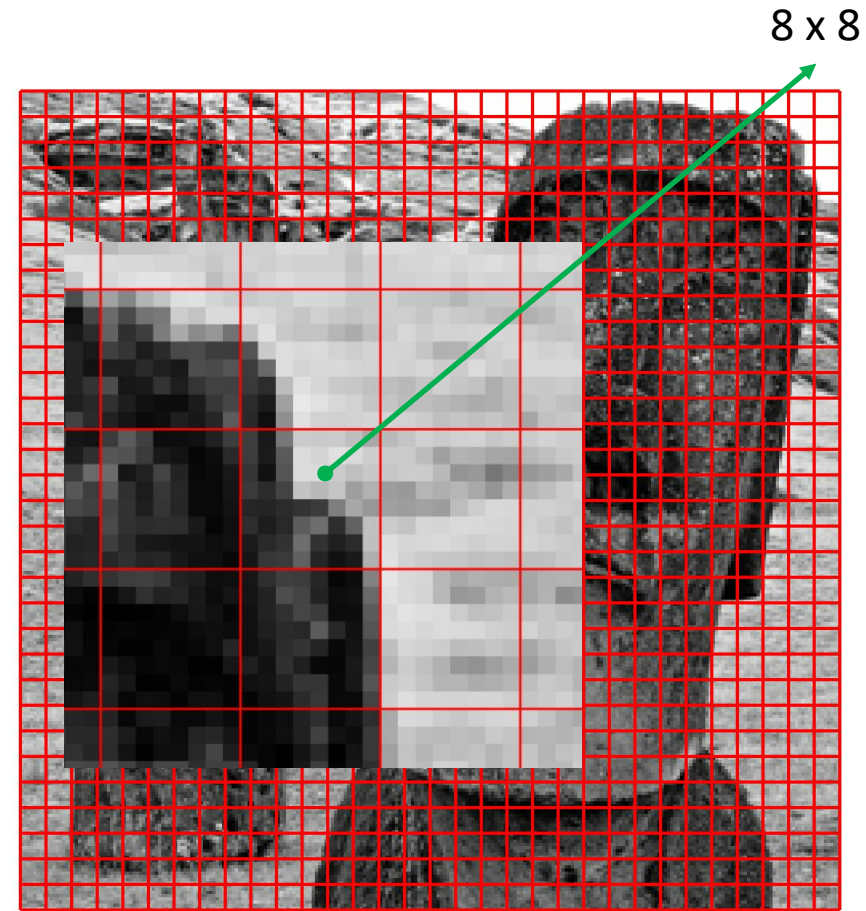


Aplicación: Compresión JPEG

1. Subdivisión en ventanas de 8 x 8

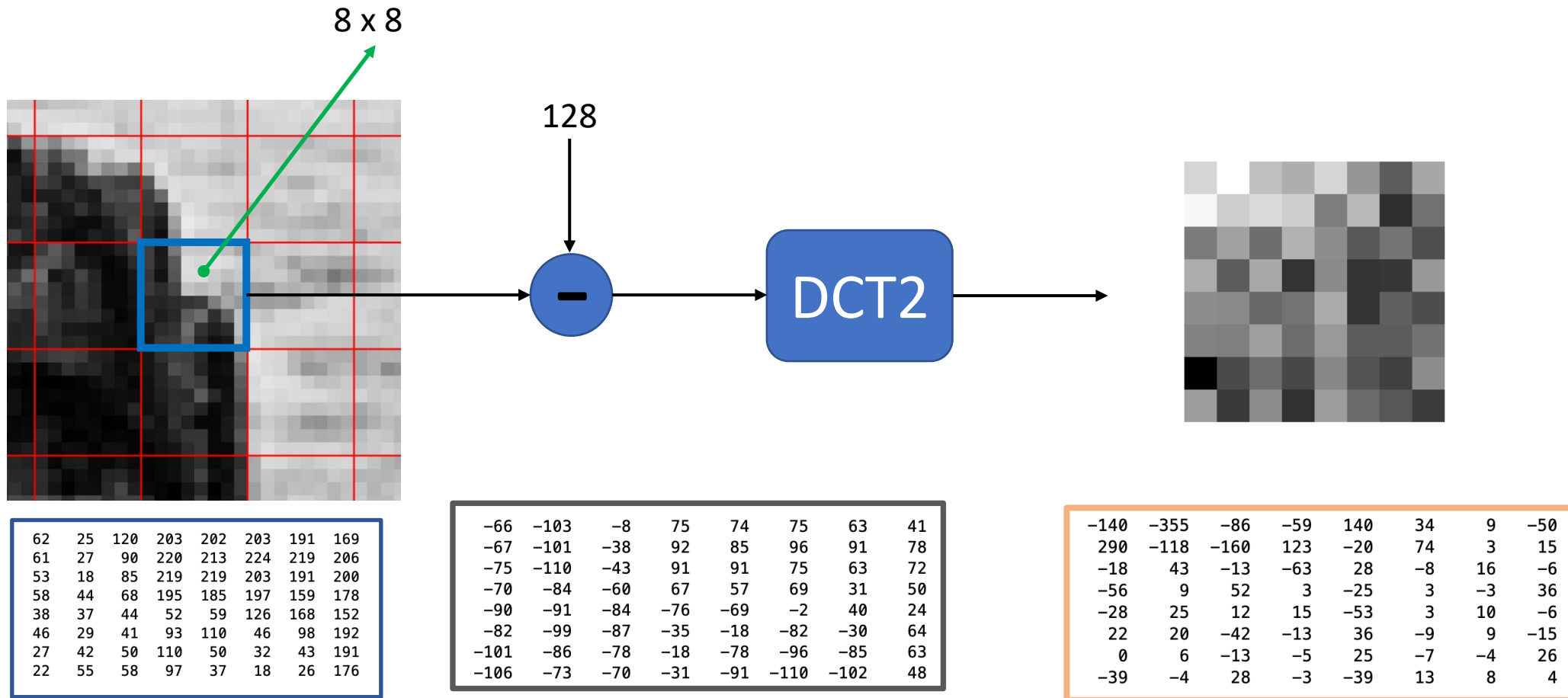


256 x 256



256 x 256

2. DCT2 para cada ventana normalizada

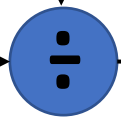


3. División por la 'Tabla de Cuantización'

A Standard
Quantization
Matrix

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

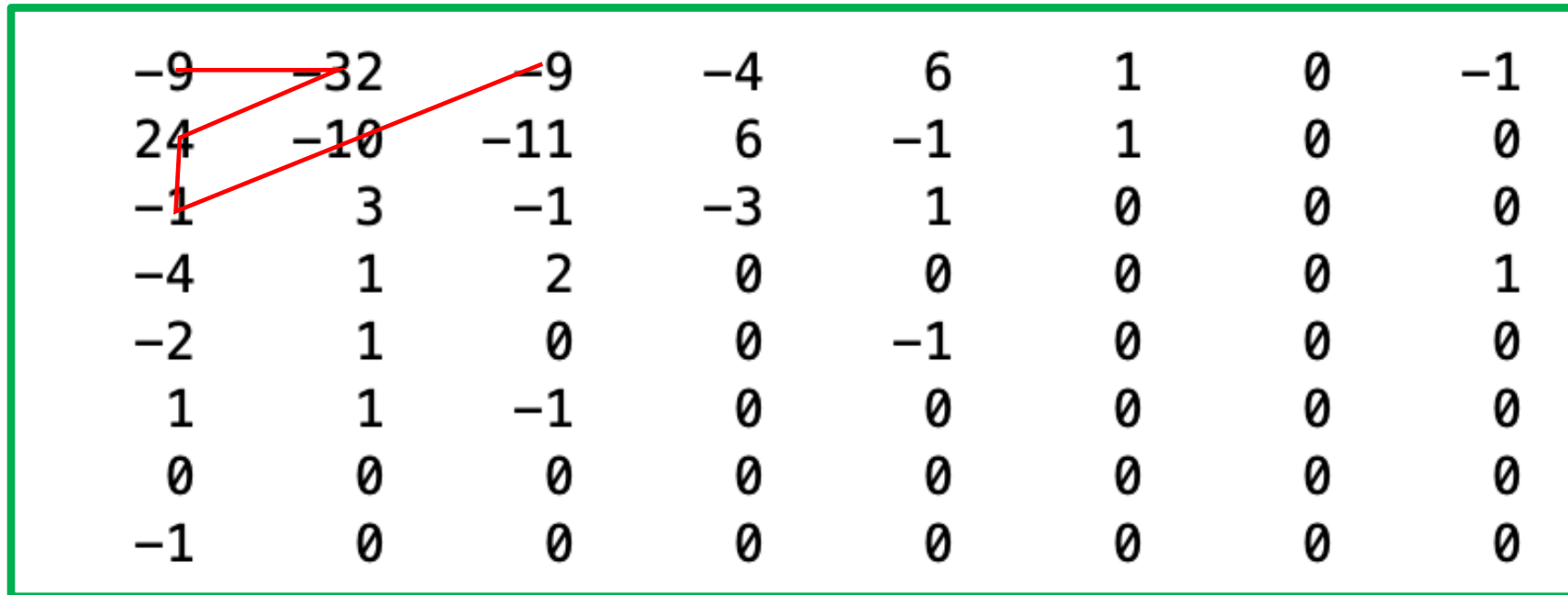
-140	-355	-86	-59	140	34	9	-50
290	-118	-160	123	-20	74	3	15
-18	43	-13	-63	28	-8	16	-6
-56	9	52	3	-25	3	-3	36
-28	25	12	15	-53	3	10	-6
22	20	-42	-13	36	-9	9	-15
0	6	-13	-5	25	-7	-4	26
-39	-4	28	-3	-39	13	8	4



round

-9	-32	-9	-4	6	1	0	-1
24	-10	-11	6	-1	1	0	0
-1	3	-1	-3	1	0	0	0
-4	1	2	0	0	0	0	1
-2	1	0	0	-1	0	0	0
1	1	-1	0	0	0	0	0
0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0

4. Vectorización en zig-zag



-9	-32	-9	-4	6	1	0	-1
24	-10	-11	6	-1	1	0	0
-1	3	-1	-3	1	0	0	0
-4	1	2	0	0	0	0	1
-2	1	0	0	-1	0	0	0
1	1	-1	0	0	0	0	0
0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0

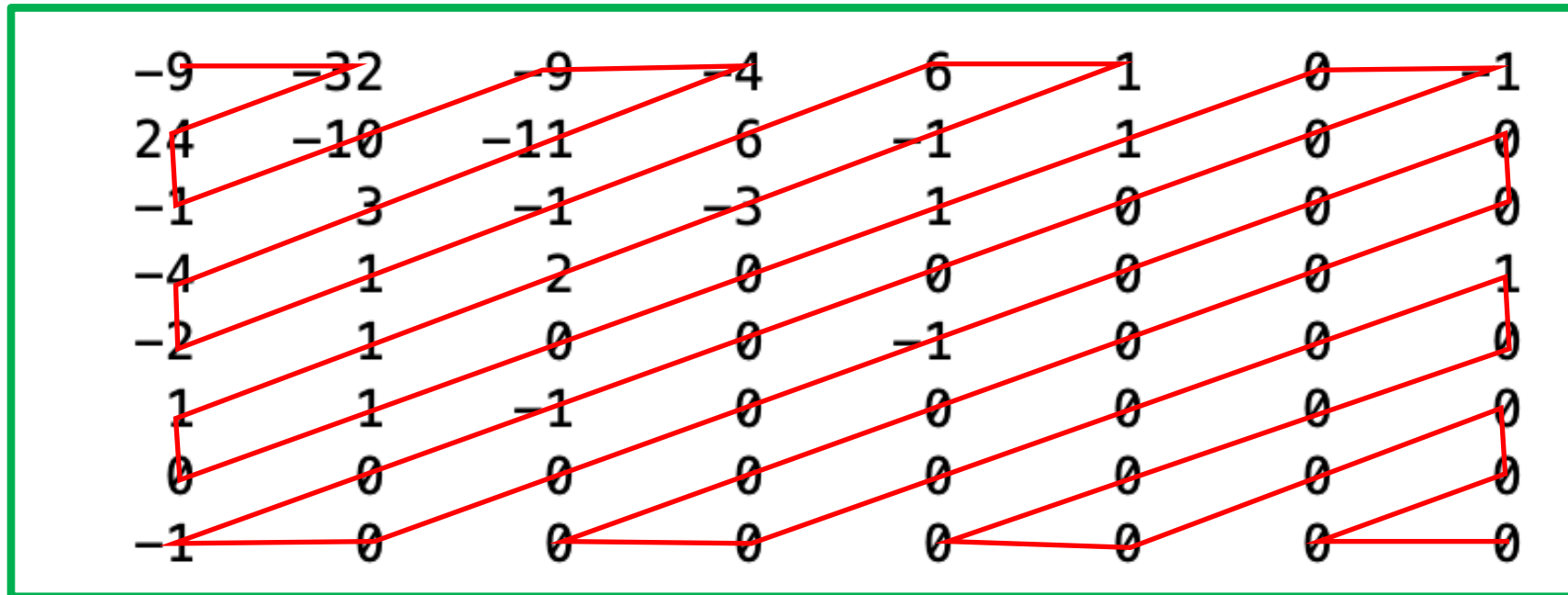
-9, -32, 24, -1, -10, -9,

4. Vectorización en zig-zag

-9	-32	-9	-4	6	1	0	-1
24	-10	-11	6	-1	1	0	0
-1	3	-1	-3	1	0	0	0
-4	1	2	0	0	0	0	1
-2	1	0	0	-1	0	0	0
1	1	-1	0	0	0	0	0
0	0	0	0	0	0	0	0
-1	0	0	0	0	0	0	0

-9, -32, 24, -1, -10, -9, -4, -11, 3, -4, -2, 1, -1, 6, 6,

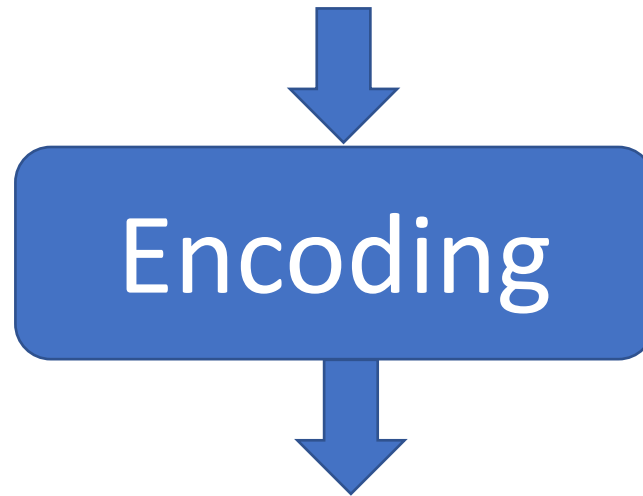
4. Vectorización en zig-zag



-9, -32, 24, -1, -10, -9, -4, -11, 3, -4, -2, 1, -1, 6, 6, 1, -1, -3, 2, 1, 1, 0, 1, 0, 0, 1, 1, 0, -1, 0, 0, 0, 0, 0, -1, 0, -1, 0, 0, 0, ...

5. Huffman encoding

-9, -32, 24, -1, -10, -9, -4, -11, 3, -4, -2, 1, -1, 6, 6, 1, -1, -3, 2, 1, 1, 0, 1, 0, 0, 1, 1, 0, -1, 0, 0, 0, 0, 0, -1, 0, -1, 0, 0, 0, ...



Codificación sin Perdida que asigna un menor número de bits a los términos más frecuentes

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas [ABAABCO-ATAABCO](#)

Frecuencias: A : 6

 B : 3

 C : 2

 O : 2

 - : 1

 T : 1

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

Frecuencias: A : 6

 B : 3

 C : 2

 O : 2

 - : 1 ——— 2

 T : 1 /

Combinar las
dos menos frecuentes

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

Frecuencias: A : 6

 B : 3

 C : 2

 O : 2 ————— 4

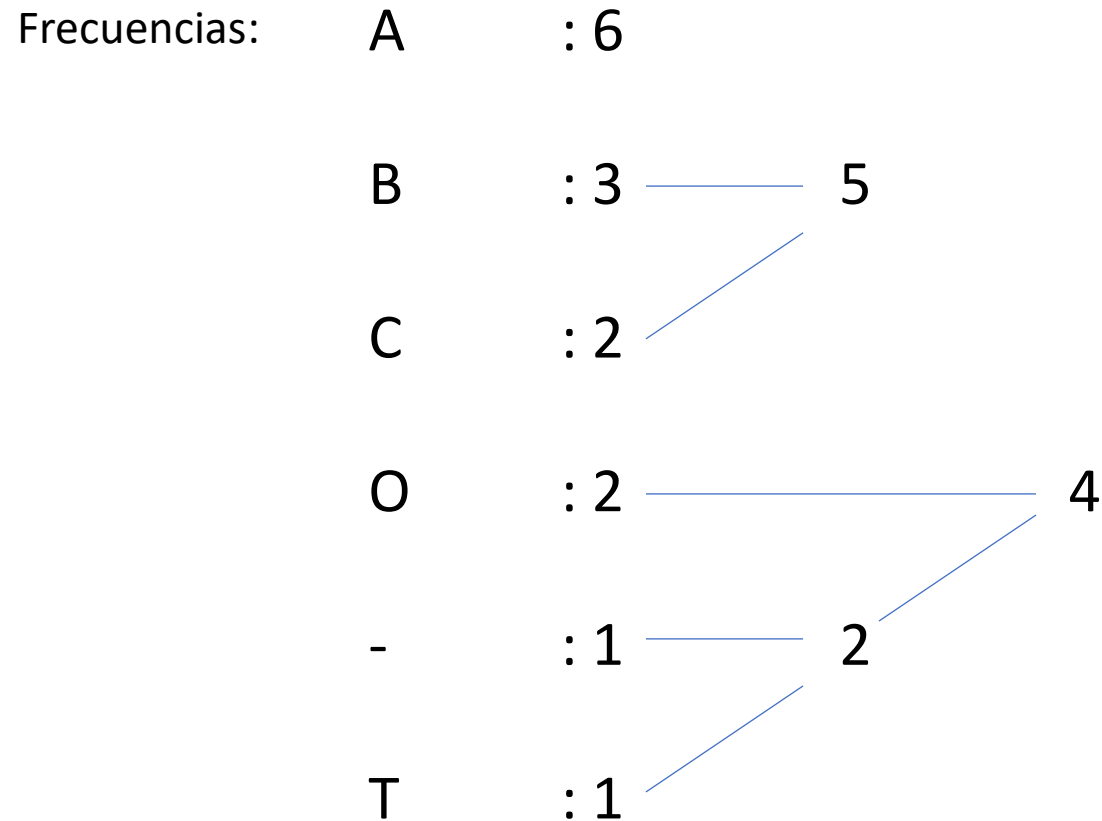
 - : 1 ——— 2

 T : 1 /

Combinar las
dos menos frecuentes

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**



Combinar las
dos menos frecuentes

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

Frecuencias: A : 6

 B : 3 5 9

 C : 2

 O : 2 4

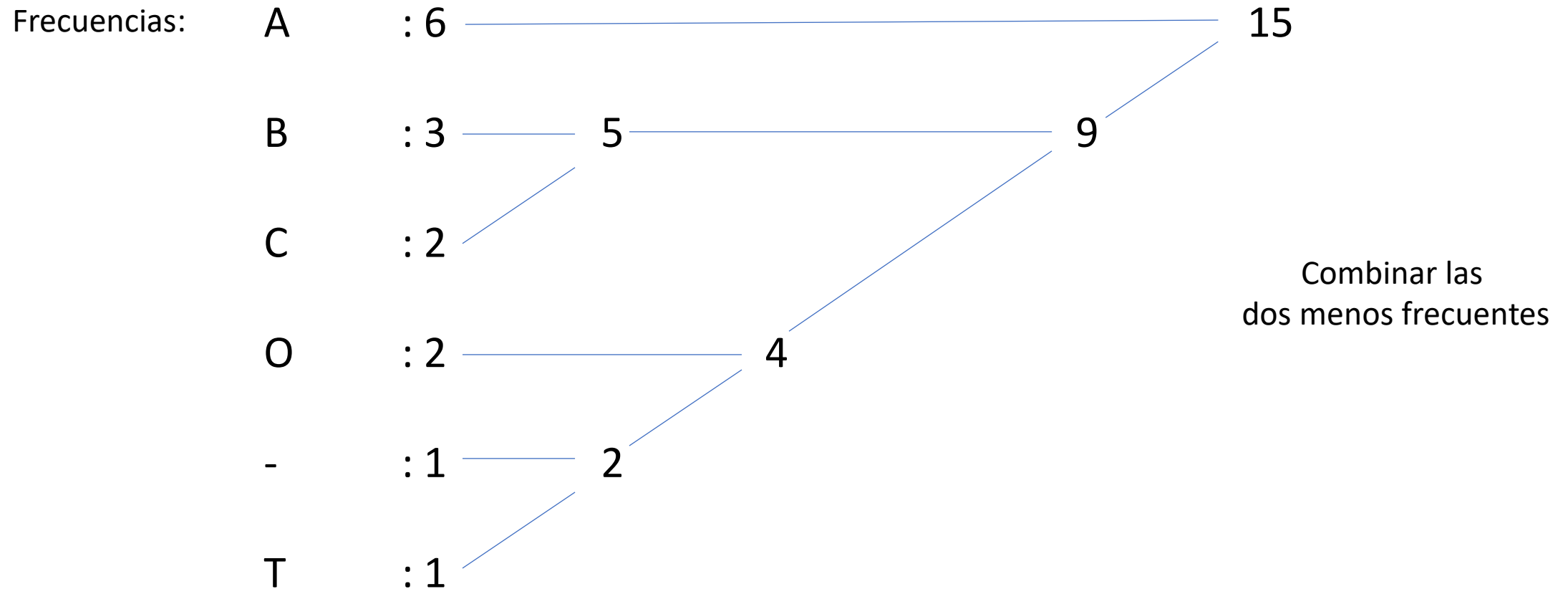
 - : 1 2

 T : 1

Combinar las
dos menos frecuentes

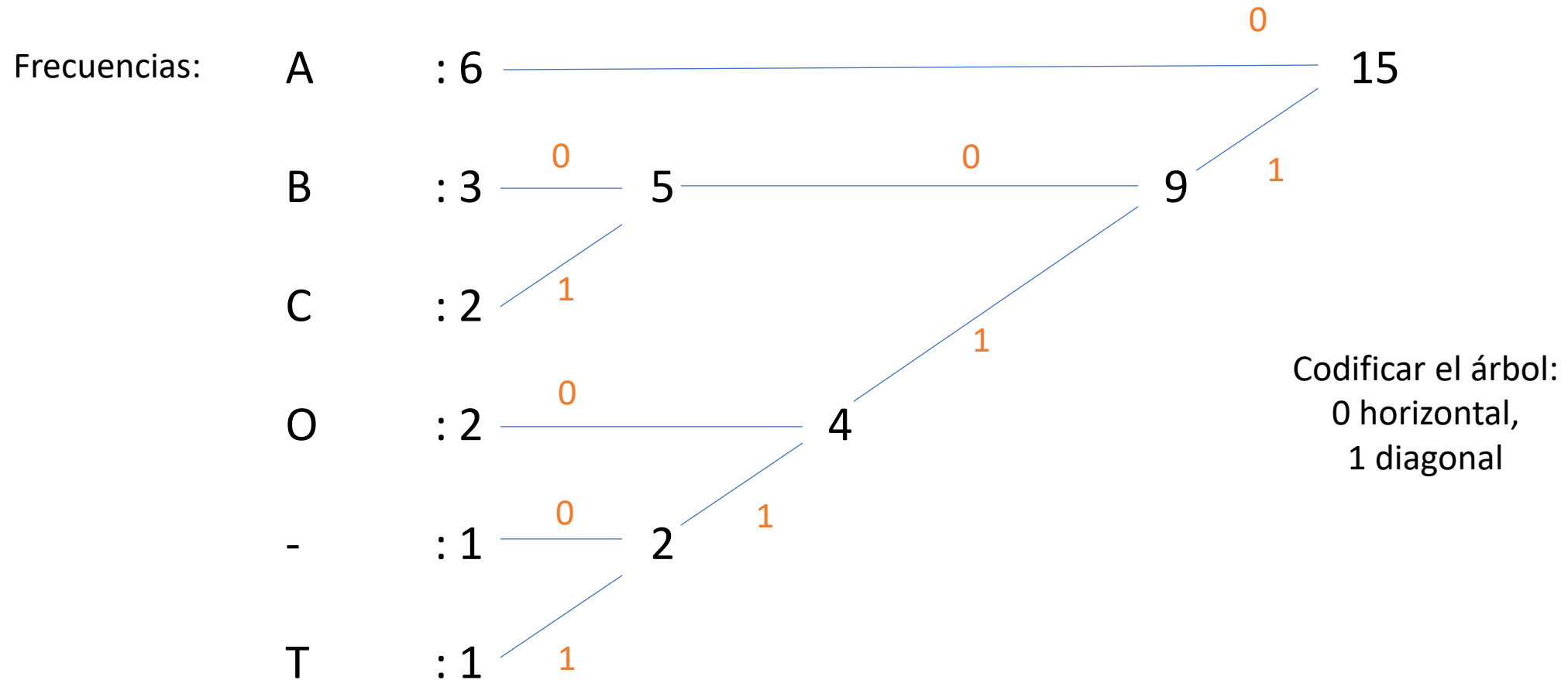
5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**



5. Huffman encoding

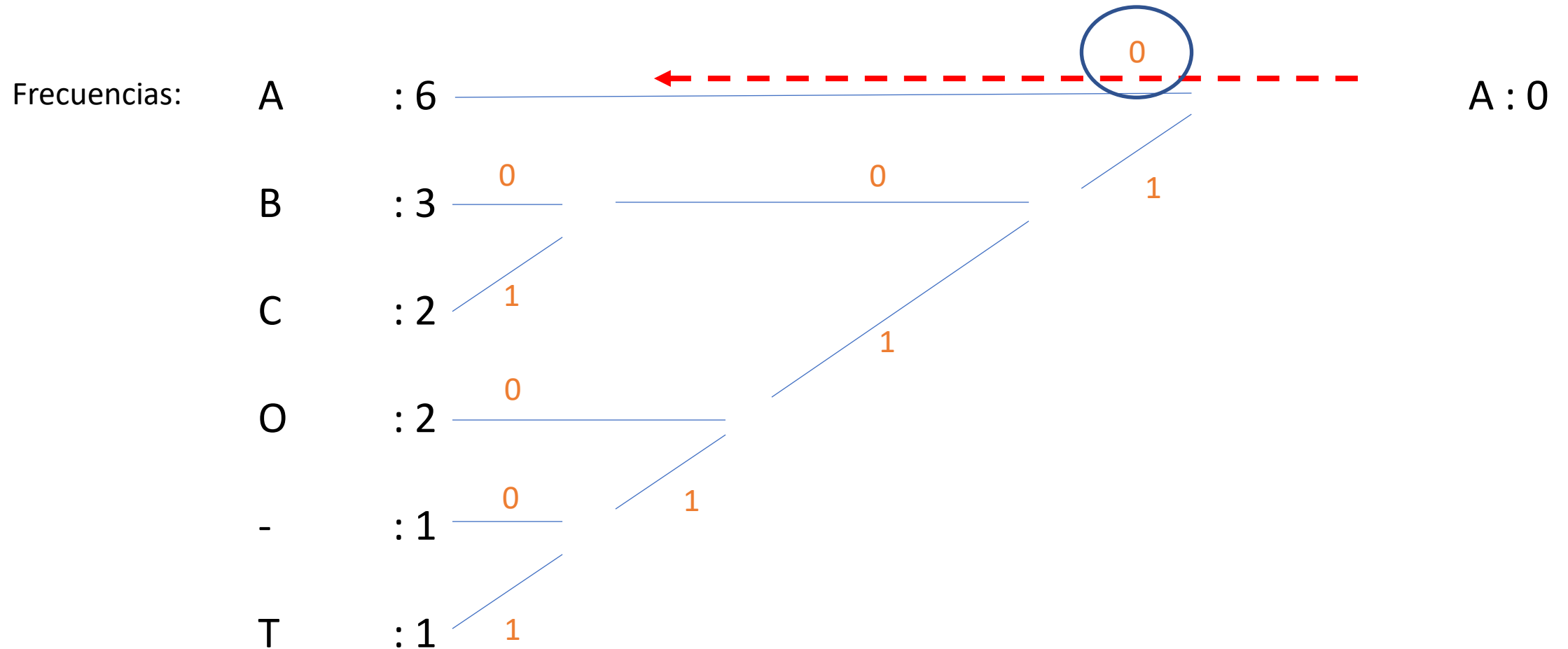
Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

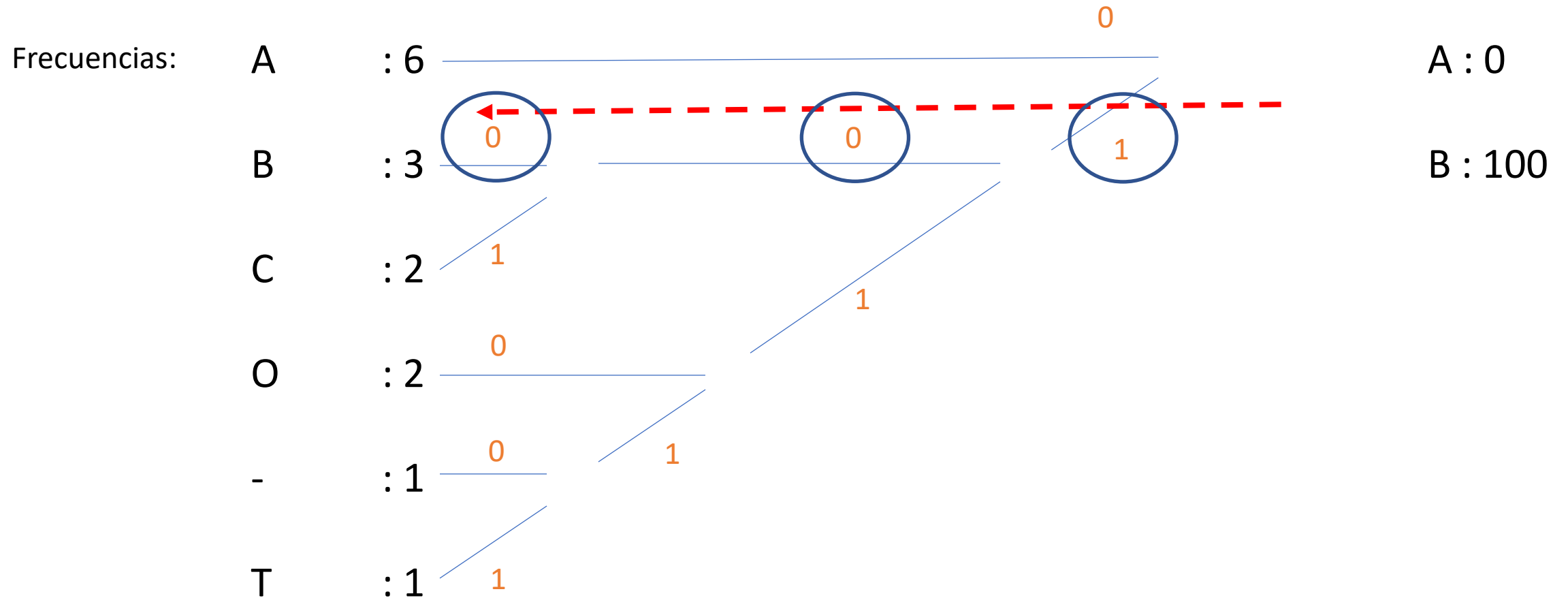
Construir código de cada caracter



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

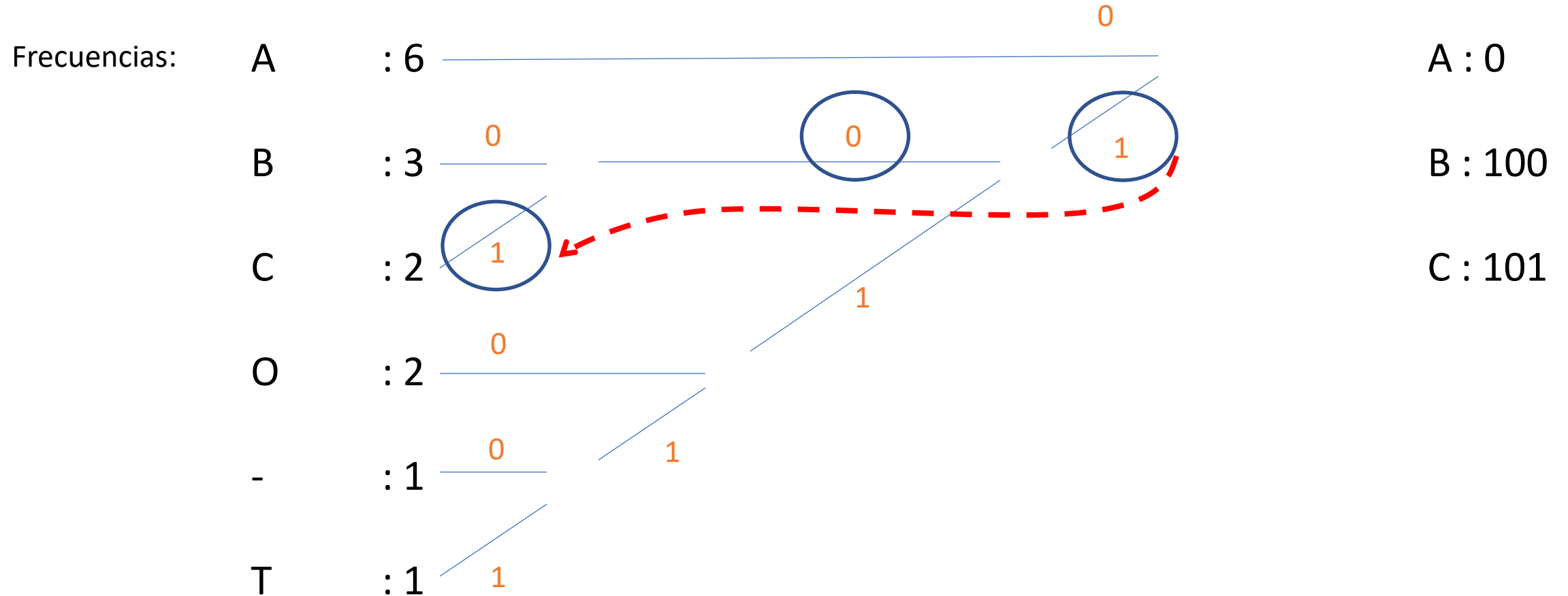
Construir código de cada caracter



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

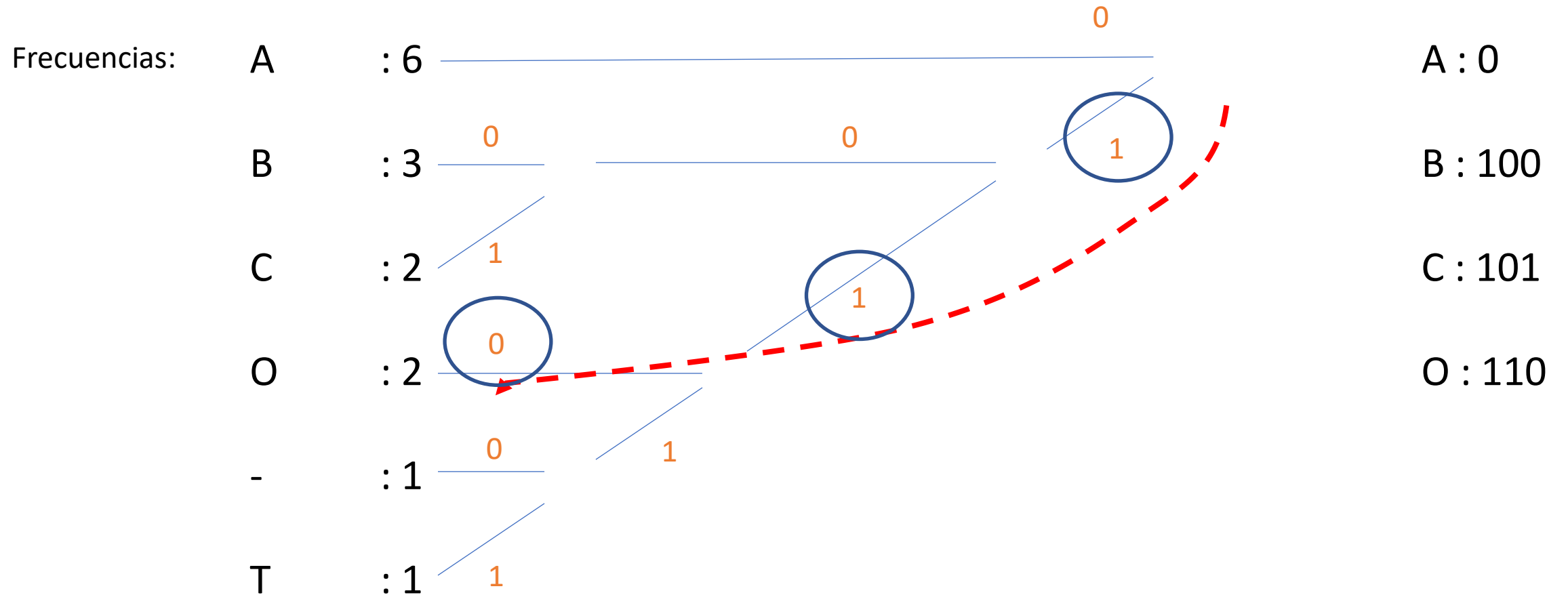
Construir código de cada caracter



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

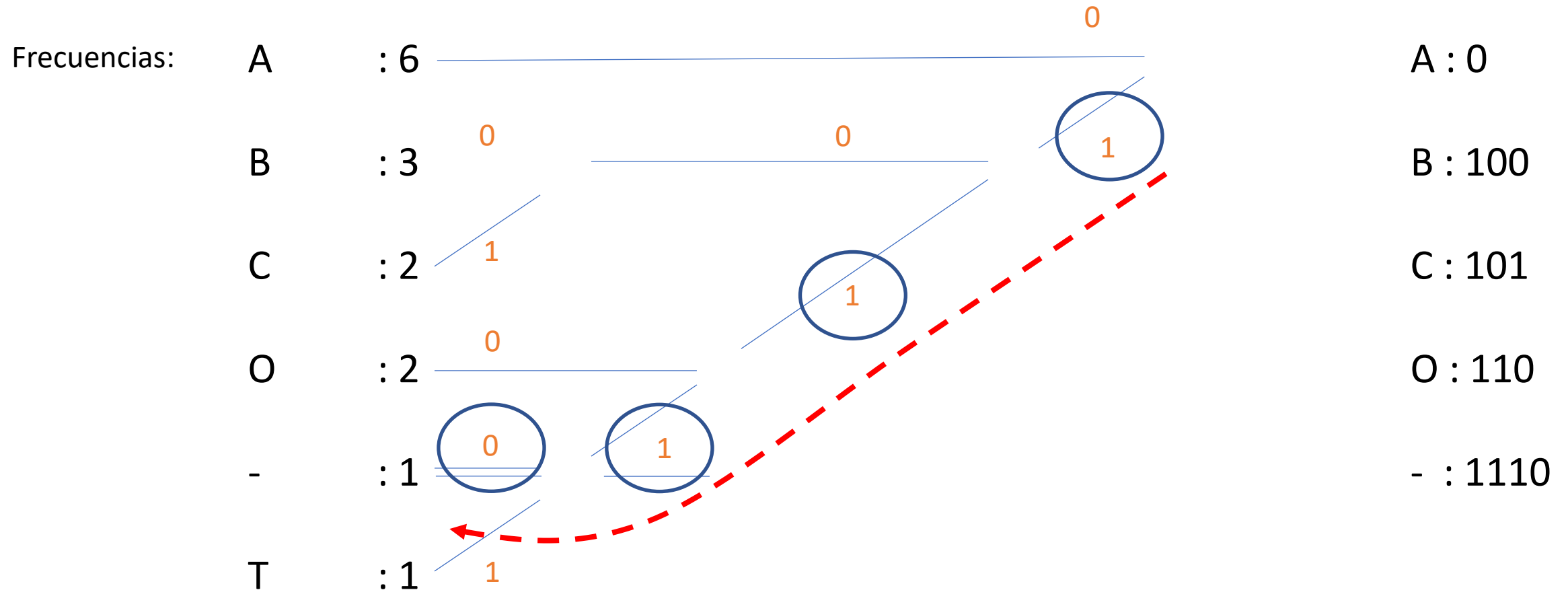
Construir código de cada caracter



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

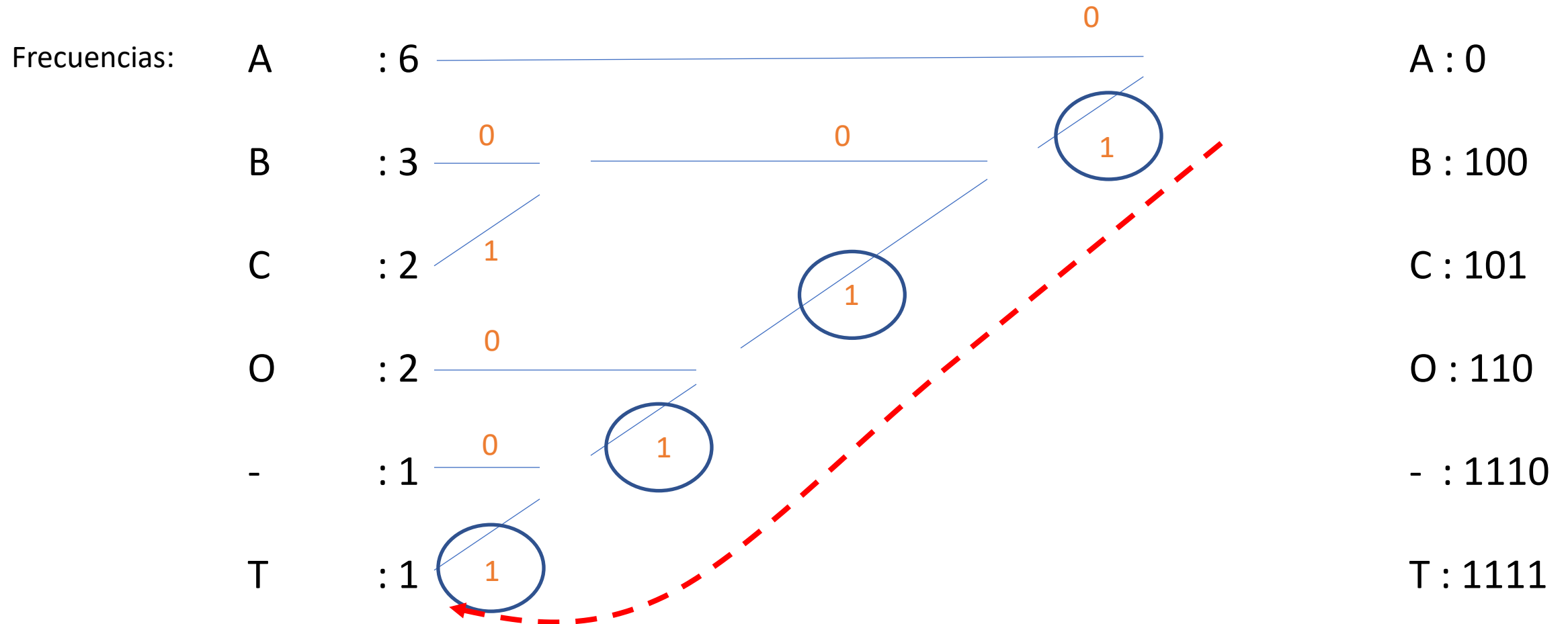
Construir código de cada caracter



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

Construir código de cada caracter



5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **ABAABCO-ATAABCO**

Construir código
de cada caracter

A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

Construir código
de cada caracter

A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

0



Construir código
de cada caracter

A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

0100



Construir código
de cada caracter

A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

010000



Construir código
de cada caracter

A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

Construir código
de cada caracter

010000100



A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

Construir código
de cada caracter

010000100101



A : 0

B : 100

C : 101

O : 110

- : 1110

T : 1111

5. Huffman encoding

Ejemplo: Codificar las palabras mágicas **A****B****A****A****B****C****O**-**A****T****A****A****B****C****O**

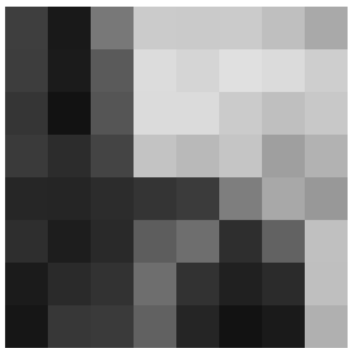
Construir código
de cada caracter

010000010010111011100111100100101110

A : 0
B : 100
C : 101
O : 110
- : 1110
T : 1111

Resultados

Original



8 x 8



Hi-Quality



Med-Quality



Low-Quality

