Practico 6 Matchings

I) Costo de asignar trabajos I, II, ... de una X a trabajadores A, B, ...
Hallar matching que minimize el mayor costo que deba pagar un trabajador.

a)

	I	II	III	IV	V	VI	VII	VIII
A	1	2	5	5	3	8	2	9
B	9	8	8	9	8	8	1	3
C	3	1	5	8	9	6	5	8
D	9	1	7	9	3	8	8	5
E	8	9	2	4	8	5	9	9
F	9	8	3	8	8	9	8	1
G	5	4	8	9	1	8	9	8
H	8	8	9	1	8	3	1	1

Elem. matriz: 1, 2, 3, 4, 5, 6, 7, 8, 9

Umbral 5 ⇒ "1" si ≤ 5 ó "0" si no. Aplico Búsqueda Binaria

	I	II	III	IV	V	VI	VII	VIII	
A	1	1	1	1	1	0	1	0	-I
B	0	0	0	0	0	0	1	1	VII
C	1	1	1	0	0	0	1	0	-II
D	0	1	0	0	1	0	0	1	-V
E	0	0	1	1	0	1	0	0	III
F	0	0	1	0	0	0	0	1	VIII
G	1	1	0	0	1	0	0	0	✗
H	0	0	0	1	0	1	1	1	IV
	~~G~~	~~G~~	A	A	~~G~~	E	A	D	

Hay matching perfecto ≤ 5 !!

A	III	: 5
B	VII	: 1
C	II	: 1
D	V	3
E	VI	: 5
F	VIII	: 1
G	I	: 5
H	IV	: 1

Costo max = 5

Ahora si tomamos umbral 2, no hay matching pues la columna VI tiene todos sus pesos mayores a 2

Umbral 3 ≤ 3

	I	II	III	IV	V	VI	VII	VIII	
A	1	1	0	0	1	0	1	0	I
B	0	0	0	0	0	0	1	1	VII
C	1	1	0	0	0	0	0	0	II
D	0	1	0	0	1	0	0	0	V
E	0	0	1	0	0	0	0	0	III
F	0	0	1	0	0	0	0	1	VIII
G	0	0	0	0	1	0	0	0	✗
H	0	0	0	1	0	1	1	1	
	~~C~~	~~D~~	~~F~~		~~G~~		A	B	

no puedo extender el matching
∄ matching perfecto de costo ≤ 3
(me marigue)

- S = filas etiqueteadas al final = {A, B, C, D, E, F, G}
- T = {columnas etiqueteadas} = {I, II, III, V, VII, VIII}

$\boxed{\Gamma(S) = T}$

$|S| = 7$

$|\Gamma(S)| = 6$ ← Γ

$6 < 7$

Umbral 4

	I	II	III	IV	V	VI	VII	VIII	
A	1	1	0	0	1	0	1	0	I
B	0	0	0	0	0	0	1	1	VII
C	1	1	0	0	0	0	0	0	II
D	0	1	0	0	1	0	0	0	V
E	0	0	1	1	0	0	0	0	III
F	0	0	1	0	0	0	0	1	VIII
G	0	1	0	0	1	0	0	0	✗
H	0	0	0	1	0	1	1	1	IV
	~~C~~	~~G~~	~~E~~	~~E~~	~~G~~	~~H~~	A	B	

Hay matching perfecto ≤ 4

A	VII	: 2
B	VIII	: 3
C	I	: 3
D	V	: 3
E	IV	: 4
F	III	: 3
G	II	: 4
H	VI	: 3

Costo max = 4

Elementos Matriz: 2, 3, 4, 5, 6, 7, 8, 9, 10

B)

Umbral ≤6

	I	II	III	IV	V	VI	VII	VIII	IX	
A	0	0	0	1	1	1	0	1	0	IV
B	1	1	0	1	1	1	1	1	1	I
C	1	1	1	1	1	1	1	1	0	II
D	0	0	1	1	1	1	1	1	1	III
E	0	1	0	1	1	1	0	0	0	V
F	0	0	0	1	1	1	0	1	0	VI
G	0	0	0	0	0	1	0	0	0	*
H	0	0	0	1	1	1	0	1	0	VII
I	0	1	0	1	1	1	0	0	0	*
	C	I	C	I	I	G	C	A		
		E		F	F	G		F		

IV

No puedo extender el matching ∄ matching perfecto de costo ≤6

S = filas etiqueteadas = {A, E, F, G, H, I}

t = columnas etiqueteadas {II, IV, V, VI, VIII}

$\Gamma(s) = T$

$|s| = 6 \quad |\Gamma(s)| = 5$ y $5 < 6$.

Umbral ≤8

	I	II	III	IV	V	VI	VII	VIII	IX
A	0	1	1	1	1	1	1	1	1
B	1	1	1	1	1	1	1	1	1
C	1	1	1	1	1	1	1	1	0
D	1	1	1	1	1	1	1	1	1
E	1	1	1	1	1	1	1	1	1
F	1	1	1	1	1	1	1	1	1
G	0	1	0	1	1	1	1	1	1
H	1	1	1	1	1	1	1	1	1
I	1	1	1	1	1	1	1	1	1

Hay matching perfecto

A II : 8
B I : 5
C III : 5
D IV : 5
E V : 5
F VI : 5
G VII : 8
H VIII : 5
I IX : 7

costo max: 5

Umbral ≤7

	I	II	III	IV	V	VI	VII	VIII	IX	
A	0	0	0	1	1	1	0	1	0	IV
B	1	1	1	1	1	1	1	1	1	I
C	1	1	1	1	1	1	1	1	0	II
D	0	1	1	1	1	1	1	1	1	III
E	0	1	0	1	1	1	1	1	1	V
F	0	1	1	1	1	1	1	1	1	VI
G	0	1	0	1	1	1	0	0	0	*
H	1	1	1	1	1	1	1	1	1	VII
I	0	1	1	1	1	1	1	1	1	VIII
	C	G	C	G	G	G	C	A	E	

Hay matching perfecto

A IV : 5
B I : 5
C II : 5
D III : 4
E IX : 7
F VI : 5
G V : 7
H VII : 7
I VIII : 7

Costo max: 7

② trabajos en paralelo no dependen uno de otro
encontrar matching que minimize el tiempo para completar el proyecto y ~~de~~ este tiempo

-Debemos minimizar el maximo

	I	II	III	IV	V	VI	VII
A	31	42	7	4	11	2	7
B	2	5	31	3	10	768	768
C	6	10	31	768	11	5	1
D	2	2	3	9	10	4	99
E	4	4	2	6	3	10	7
F	3	10	8	4	5	99	31
G	31	42	10	768	6	2	3

elementos de la matriz

{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 31, 42, 99, 768}

Umbral ≥ 8

	I	II	III	IV	V	VI	VII
A	0	0	$\boxed{1}$	1	0	1	1
B	$\boxed{1}$	1	0	1	0	0	0
C	1	0	0	0	0	$\boxed{1}$	1
D	1	$\boxed{1}$	1	0	0	1	0
E	1	1	1	$\boxed{1}$	1	0	1
F	1	0	1	1	$\boxed{1}$	0	0
G	0	0	0	0	1	1	$\boxed{1}$

Hay matching perfecto!

A III : $\boxed{7}$
B I : 2
C VI : 5
D II : 2
E IV : 6
F V : 5
G VII : 3

Costo máx: $\boxed{7}$

Umbral ≥ 4

	I	II	III	IV	V	VI	VII
A	0	0	0	$\boxed{1}$	0	1	0
B	$\boxed{1}$	0	0	1	0	0	0
C	0	0	0	0	0	0	$\boxed{1}$
D	1	$\boxed{1}$	1	0	0	1	0
E	1	1	$\boxed{1}$	0	1	0	0
F	1	0	0	1	0	0	0
G	0	0	0	0	0	$\boxed{1}$	1
→	~~F~~			~~F~~		A	G

↓ IV, I, VII, ~~F~~, VI

No fue posible extender el matching
No hay matching perfecto de costo ≥ 4

S = filas etiqueteadas
{A, B, C, F, G}

T = {columnas etiqueteadas}
{I, IV, VI, VII}

$\Gamma(S) = T$ $|S| = 5$ $|\Gamma(S)| = 4$
$|\Gamma(S)| < |S|$ $4 < 5$ ✓

Umbral ≥ 6

	I	II	III	IV	V	VI	VII
A	0	0	0	$\boxed{1}$	0	1	0
B	$\boxed{1}$	1	0	1	0	0	0
C	1	0	0	0	0	$\boxed{1}$	1
D	1	$\boxed{1}$	1	0	0	1	0
E	1	1	$\boxed{1}$	1	1	0	0
F	1	0	0	1	$\boxed{1}$	0	0
G	0	0	0	0	0	1	$\boxed{1}$

Hay matching perfecto

A IV : 4
B I : 2
C VI : 5
D II : 2
E III : 2
F V : 5
G VII : 3

Costo máx: $\boxed{5}$

Umbral ≥ 5

	I	II	III	IV	V	VI	VII
A	0	0	0	$\boxed{1}$	0	1	0
B	$\boxed{1}$	1	0	1	0	0	0
C	0	0	0	0	0	$\boxed{1}$	1
D	1	$\boxed{1}$	1	0	0	1	
E	1	1	$\boxed{1}$	1	1	0	0
F	1	0	0	1	$\boxed{1}$	0	0
G	0	0	0	0	0	1	$\boxed{1}$

Hay matching perfecto

A IV : 4
B I : 2
C VI : 5
D II : 2
E III : 2
F V : 5
G VII : 3

Costo máx: $\boxed{5}$

La mínima cantidad de días que toma completar todo el proyecto es 5.

1º restarle a c/fila el min de esa fila
2º " el min de c/columna a c/columna
3º Hallar un matching de "0"

③ Trabajos I, II, ..., etc
antes de hacer I hay que hacer II antes de II hay que hacer III

1 trabajo 1 trabajador
Matriz → tiempo en días de asignar los trabajos I, II, ... a trabajadores A, B, etc

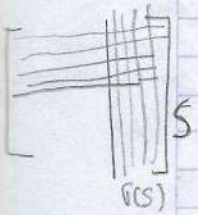
- Encontrar matching que minimize el tiempo para completar el proyecto y dar ese tiempo
↳ minimizar la suma de los costos

	I	II	III	IV	V	VI	VII	VIII
A	1	2	5	5	3	8	2	9
B	9	8	8	9	8	8	1	9
C	3	1	5	8	9	6	5	9
D	1	9	7	9	3	8	8	9
E	8	9	2	4	8	5	8	9
F	9	8	3	8	8	9	8	9
G	5	4	8	9	1	8	9	9
H	8	8	4	8	8	8	3	9

①→

	I	II	III	IV	V	VI	VII	VIII
A	0	1	4	4	2	7	1	8
B	8	7	7	8	7	7	0	8
C	2	0	4	7	8	5	4	8
D	0	8	6	8	2	7	7	8
E	6	7	0	2	6	3	7	7
F	6	5	0	5	5	6	5	6
G	4	3	7	8	0	7	8	8
H	5	5	1	5	5	5	0	6

②



No pude encontrar un matching de ceros sólo restando filas y columnas
∴ ∄ matching perfecto de "0" en esta matriz.

	I	II	III	IV	V	VI	VII	VIII	
A	0	1	4	2	2	4	1	2	I
B	8	7	7	6	7	4	0	2	VII
C	2	0	4	5	8	2	4	2	
D	0	8	6	6	2	4	7	2	*
E	6	7	0	0	6	0	7	1	III
F	6	5	0	3	5	3	5	0	VIII
G	4	3	7	6	0	4	8	2	
H	5	5	1	3	5	2	0	0	*
	D		F	E		E	H	H	
	D								

tachado 2 veces: sumar
tachado 1 vez: nada
tachado 0 vs 3: restar

⇒ Por Teorema de Hall ∃ S ⊆ Filas:
$|S| = |\Gamma(S)|$ donde $\Gamma(S) = T$

∴ S = Filas etiqueteadas
= {A, D} $|S| = 2$
T = columnas etiqueteadas.
{I} $|\Gamma(S)| = 1$

y $1 < 2$ ∴ No puede haber un matching perfecto

∴ Ahora tengo que tachar $\bar{S}$ y $\Gamma(S)$ y obtener $\min S \times \overline{\Gamma(S)}$ = min no tachado
$S \times \Gamma(S)$ $S \times \overline{\Gamma(S)}$

⇒ $\min S \times \overline{\Gamma(S)} = 1$ restar 1 a las filas de S y sumarle 1 a las columnas de $\Gamma(S)$

	I	II	III	IV	V	VI	VII	VIII	
A	0	0	3	1	1	3	0	1	I
B	9	1	7	6	7	4	0	2	VII
C	3	0	4	5	8	2	4	2	II
D	0	7	5	5	1	3	6	1	*
E	1	7	0	0	6	0	7	1	
F	7	5	0	3	5	3	5	0	
G	5	3	7	6	0	4	8	2	
H	6	5	1	3	5	2	0	0	
	D	A					A		

No puede encontrar matching de ceros
∴ ∄ matching perfecto de "0" en esta matriz.

⇒ Por teorema Hall ∃ S ⊆ Filas
S = filas etiqueteadas {A, B, C, D}
T = {I, II, VII} columnas etiqueteadas

$|S| = 4$ $\Gamma(S) = T \Rightarrow |\Gamma(S)| = 3$

y $3 < 4$ ∴ no puede haber matching perfecto de "0" en esta matriz.

tacho $\bar{S}$ y $\Gamma(S)$

Luego $\min S \times \overline{\Gamma(S)} = 1$

Ya no se agregan más ceros

luego sumar 1 vez, nada 0 veces restar

	I	II	III	IV	V	VI	VII	VIII	
A	0	0	2	$\boxed{0}$	0	2	0	0	~~I~~
B	9	1	6	5	6	3	$\boxed{0}$	1	~~VII~~
C	3	$\boxed{0}$	3	4	1	1	4	1	~~II~~
D	$\boxed{0}$	1	4	4	0	2	6	0	~~*~~
E	8	8	0	0	6	$\boxed{0}$	8	1	~~IV~~
F	3	6	$\boxed{0}$	3	5	3	6	0	~~III~~
G	6	4	7	6	$\boxed{0}$	4	9	2	~~V~~
H	1	6	1	3	5	2	1	$\boxed{0}$	~~VIII~~
	~~D~~	~~A~~	~~E~~	~~A~~	~~D~~	~~E~~	~~A~~	~~D~~	

pongo □ en D

me queda libre pongo □ en A
tacho el otro

¡¡Hay matching perfecto!!

			Σ de suma mínima
A	IV	:	5
B	VII	:	1+5=6
C	II	:	1+6=7
D	I	:	1+7=8
E	VI	:	5+8=13
F	III	:	3+13=16
G	V	:	1+16=17
H	VIII	:	9+17=$\boxed{26}$

Este matching minimiza el tiempo para completar el proyecto haciendo que se demore $\boxed{26 \text{ días}}$ en realizarlo.

4) Costo que la empresa X deberá incurrir al asignar los trabajos I, II, ..., etc a los trabajadores A, B, etc. Asignar todos los trabajos en forma tal que el costo total de la empresa X sea mínimo.

MINIMIZAR LA SUMA EN TODOS.

a) (resto min de c/ fila)

	I	II	III	IV	V	VI	VII	
A	1	1	0	2	4	2	0	
B	0	0	1	2	0	2	0	~~I~~
C	2	0	3	3	2	3	2	~~II~~
D	0	0	4	3	4	4	1	~~*~~
E	0	0	4	5	0	0	3	
F	3	4	4	0	3	4	0	
G	3	0	3	4	4	1	0	~~VII~~
	~~D~~	~~D~~					~~B~~	

luego tendría que restar el min de c/ columna sólo me cambia la última columna (borro y cambio) p/ ahorrar espacio

→ No pude encontrar matching de cero solo restando filas y columnas

¡¡∃ matching perfecto de 0 en esta matriz

$S = \{B, C, D, G\}$ $|S| = 4$
$\Gamma = \{I, II, VII\}$ $|\Gamma(S)| = 3$

así no puedo tener MP.

Tengo que tachar S y Γ(S)

$\min S \times \overline{\Gamma(S)} = 1$

	I	II	III	IV	V	VI	VII	
A	5	2	0	2	4	2	1	
B	0	0	1	1	3	1	0	~~I~~
C	2	0	2	2	1	2	2	~~II~~
D	0	0	3	2	3	3	2	~~*~~
E	1	1	4	5	0	0	4	
F	4	5	4	0	3	4	1	
G	3	0	2	3	3	0	0	~~VII~~
	~~D~~	~~D~~				~~G~~	~~B~~	

¡Hay matching perfecto! de suma mínima

A	III	:	4
B	VII	:	5
C	II	:	6
D	I	:	5
E	V	:	4
F	IV	:	5
G	VI	:	6

$\Sigma = \boxed{35}$

resto 3 en esta columna

resta 1 en esta columna

b)

	I	II	III	IV	V	VI	VII	
A	3	3	4	8	0	8	3	V
B	3	3	6	0	6	3	2	
C	2	0	0	1	5	1	4	
D	0	3	4	0	2	1	6	
E	1	4	5	2	4	4	0	
F	3	5	4	3	0	2	8	*
G	2	2	4	4	1	0	7	
					F			

No puedo encontrar matching de ceros sólo restando filas y columnas.

∴ ∄ MP de "0" en esta matriz

Por t. hall ∃ S ⊆ Filas,

S = {A, F} |S| = 2 1 < 2 ✓

T = {V} |Γ(S)| = 1 no puede haber MP

tacho S y Γ(S)

min S × Γ(S) = 2

	I	II	III	IV	V	VI	VII	
A	1	1	2	6	0	6	1	V
B	3	3	6	0	6	3	2	
C	2	0	0	1	5	1	4	
D	0	3	4	0	3	1	6	
E	1	4	5	2	3	4	0	
F	1	3	2	1	0	0	6	*
G	2	2	4	4	2	0	7	VI
					F	F		

2 sumar
0 restar
1 NADA

No encuentro M de 0

Por t. Hall ∃ S ⊆ Filas

S = {A, F, G} |S| = 3

T = {V, VI} |Γ(S)| = 2

2 < 3 ✓ no puede haber un matching perfecto de 0's.

Tacho S y Γ(S) ⇒ min S × Γ(S) = 1

	I	II	III	IV	V	VI	VII	
A	0	0	1	5	0	6	0	V
B	3	3	6	0	7	4	2	IV
C	2	0	0	1	7	2	4	V
D	0	3	4	0	4	2	6	I
E	1	4	5	2	6	5	0	VII
F	0	2	1	0	0	0	5	*
G	1	1	3	3	2	0	6	VI
	F	A	C	F	F	F	A	

¡Hay matching Perfecto!

A II : 4
B IV : 2
C III : 3
D I : 4
E VII : 1
F V : 1
G VI : 2

Σ 17

c)

resto 1 resta 3 resta 2

	I	II	III	IV	
A	0	0	0	0	I
B	0	3	4	1	*
C	0	10	8	5	*
D	0	9	8	6	*
	B	A	A	A	
	C				

No hay matching perfecto

Por t. Hall ∃ S ⊆ Filas y

S = {B, C, D} |S| = 3 1 < 3 ✓ ∴ no puede haber un matching perfecto de 0's

T = {I} |T| = 1

tacho S y Γ(S)

min S × Γ(S) = 1

(2)

	I	II	III	IV	
A	1	0	0	0	
B	0	2	3	0	I
C	0	9	7	4	* I
D	0	8	7	5	* *
	C			B	
	D				

No hay matching perfecto

Por T. Hall ∃ S ⊆ Filas

S = {C, D} |S| = 2 1 < 2 ✓ ∴ No puede haber matching perfecto de 0's

T = {I} |T| = 1

tacho S y Γ(S) min S × Γ(S) = 4

(3)

Tacho S y Γ(S)

min S × Γ(S) = 2

	I	II	III	IV	
A	5	0	0	0	
B	4	2	3	0	IV
C	0	5	3	0	V
D	0	4	3	1	*
	D			C	

No hay MP

T Hall

S = {B, C, D}

T = {I, IV}

|S| = 3 1 < 3 ✓ ∴ no puede haber MP.

|Γ(S)| = 1

(4)

	I	II	III	IV	
A	7	0	0	2	II
B	4	0	1	0	IV
C	0	3	1	0	I
D	0	2	1	1	*
	D	B	A	C	

¡Hay MP!

A II : 10
B II : 6
C IV : 15
D I : 1

Σ : 32

D)

	I	II	III	IV	V	VI	VII	VIII	
A	0	1	2	1	0	6	3	1	I
B	0	3	5	1	8	1	6	0	II
C	2	3	2	0	0	0	1	5	IV
D	1	0	1	1	6	3	0	5	V
E	2	2	0	0	1	1	3	2	III
F	0	5	5	6	4	0	3	6	VI
G	0	7	0	4	1	5	3	5	*
H	4	4	6	4	6	2	7	0	*
	G		G	E	A			H	
	B		H	E	C	C		H	

IV
VIII
IV
III
VI
I
*

No hay matching perfecto de 0 sólo restando filas y columnas
por T. Hall $\exists$ $S \subseteq$ filas y

$S = \{A, B, C, E, F, G, H\}$

$T = \{I, II, IV, V, VI, VIII\}$

$|S| = 7$ $|\Gamma(S)| = 6$

$6 < 7$ $\therefore$ no puede haber matching perfecto en esta matriz.

tacho $\overline{S}$ y $\Gamma(S)$ min $S \times \overline{\Gamma(S)} = 1$

	I	II	III	IV	V	VI	VII	VIII	
A	0	0	2	1	0	6	2	1	V
B	0	2	5	1	8	1	5	0	VIII
C	2	2	2	0	0	0	0	5	IV
D	2	0	3	2	7	4	0	6	
E	2	1	0	0	2	1	2	2	III
F	0	4	5	6	4	0	2	6	VI
G	0	6	0	4	1	5	2	5	I
H	4	3	0	4	6	2	6	0	*
	B		H	E	C	C	C	H	

¡Hay matching perfecto!

A V : 1
B VIII : 1
C VII : 2
D II : 2
E IV : 2
F VI : 2
G I : 1
H III : 2

Σ $\boxed{13}$

e)

	I	II	III	IV	V	VI	VII	VIII	IX	
A	0	0	1	3	4	2	2	8	6	I
B	0	0	0	1	2	4	2	3	0	II
C	0	0	2	2	3	0	2	2	2	VI
D	0	0	2	3	5	7	0	7	0	
E	5	5	3	2	0	2	2	0	3	V
F	0	0	4	8	3	0	2	2	2	*
G	0	2	2	3	0	4	5	0	5	VIII
H	0	0	3	0	3	3	0	4	4	
I	0	7	8	7	0	8	6	0	7	*
	F	E	B		I	F		I	B	
	I	A			I	F		I		

I
VI
V
*
II
VIII
*

No hay matching de "0" perfecto sólo restando filas y columnas.
Por T. Hall $\exists$ $S \subseteq$ filas

$S = \{A, C, E, F, G, I\}$

$T = \{I, II, V, VI, VIII\}$

$|S| = 6$ $|\Gamma(S)| = 5$

$5 < 6$ $\therefore$ no puede haber matching perfecto

Tacho $\overline{S}$ y $\Gamma(S)$ min $S \times \overline{\Gamma(S)} = 1$

	I	II	III	IV	V	VI	VII	VIII	IX	
A	0	0	0	2	4	2	1	8	5	I
B	1	1	0	1	3	5	2	4	0	VII
C	0	0	1	1	3	0	1	2	1	VI
D	1	1	2	3	6	8	0	8	0	
E	5	5	2	1	0	2	1	0	2	V
F	0	0	3	7	3	0	1	2	1	II
G	0	2	1	2	0	4	5	0	4	VIII
H	1	1	3	0	4	4	0	5	4	
I	0	7	7	6	0	8	5	0	6	*
	I	A	A		I	F		I	B	

¡Hay matching perfecto!

A III : 2
B IX : 6
C VI : 4
D VII : 2
E V : 5
F II : 2
G VIII : 3
H IV : 4
I I : 1

Σ $\boxed{29}$

2 sumo
1 NADA
0 resto

Elementos de matriz: {2, 3, 4, 5, 6, 7, 8, 9, 10, 11}

minimizar el max costo

5 Costo representa tanto el tiempo en horas que se tarda en hacer el trabajo como lo que cuesta el trabajo. trabajos en paralelo, no dependen uno de otro

	I	II	III	IV	V	VI	VII	VIII	IX
A	9	8	8	5	6	6	8	6	8
B	5	6	7	4	2	3	6	5	6
C	5	5	5	2	2	3	4	3	9
D	8	7	4	5	5	5	4	6	4
E	8	5	8	6	5	3	7	7	7
F	8	7	7	4	5	5	7	5	7
G	11	8	10	8	8	6	9	9	9
H	7	7	7	4	5	5	7	5	7
I	9	6	8	6	6	4	8	8	8

Umbral ≤ 6

	I	II	III	IV	V	VI	VII	VIII	IX
A	0	0	0	1	1	1	0	1	0
B	1	1	0	1	1	1	1	1	1
C	1	1	1	1	1	1	1	1	0
D	0	0	1	1	1	1	1	1	1
E	0	1	0	1	1	1	0	0	0
F	0	0	0	1	1	1	0	1	0
G	0	0	0	0	0	1	0	0	0
H	0	0	0	1	1	1	0	1	0
I	0	1	0	1	1	1	0	0	0

No hay matching perfecto

$S = \{A, E, F, G, H, I\}$

$T = \{II, IV, V, VI, VIII\}$

$|S| = 6 \quad |\Gamma(S)| = 5$

$5 < 6$ ✓ ∴ no puede haber MP. con costo ≤ 6

Umbral ≤ 9

	I	II	III	IV	V	VI	VII	VIII	IX
A	1	1	1	1	1	1	1	1	1
B	1	1	1	1	1	1	1	1	1
C	1	1	1	1	1	1	1	1	1
D	1	1	1	1	1	1	1	1	1
E	1	1	1	1	1	1	1	1	1
F	1	1	1	1	1	1	1	1	1
G	0	1	0	1	1	1	1	1	1
H	1	1	1	1	1	1	1	1	1
I	1	1	1	1	1	1	1	1	1

¡Hay matching perfecto!

AI: 9
BII: 5
CIII: 5
DIV: 5
EV: 5
FVI: 5
GVII: 9
HVIII: 5
IIX: 8

Costo máx: 9

Umbral ≤ 7

	I	II	III	IV	V	VI	VII	VIII	IX
A	0	0	0	1	1	1	0	1	0
B	1	1	1	1	1	1	1	1	1
C	1	1	1	1	1	1	1	1	0
D	0	1	1	1	1	1	1	1	1
E	0	1	0	1	1	1	1	1	1
F	0	1	1	1	1	1	1	1	1
G	0	0	0	0	0	1	0	0	0
H	1	1	1	1	1	1	1	1	1
I	0	1	0	1	1	1	0	0	0

¡Hay matching perfecto!

AVIII: 6
BI: 5
CII: 5
DIII: 4
EV: 5
FIX: 7
GVI: 6
HVII: 7
IIV: 6

Costo máx: 7

mayor a 7 Poner ∞

(*)

	I	II	III	IV	V	VI	VII	VIII	IX
A	∞	∞	∞	5	6	6	∞	6	∞
B	5	6	7	4	2	3	6	5	6
C	5	5	5	2	2	3	4	3	∞
D	∞	7	4	5	5	5	4	6	4
E	∞	5	∞	6	5	3	7	7	7
F	∞	7	7	4	5	5	7	5	7
G	∞	∞	∞	∞	∞	6	∞	∞	∞
H	7	7	7	4	5	5	7	5	7
I	∞	6	∞	6	6	4	∞	∞	∞

Sabemos que podemos terminar todos los trabajos con 7 horas. Como quiero gastarlo menos posible, voy a ignorar todos los trabajos que demoran más de 7 horas Poniendoles ∞ en la matriz original en donde están estos valores y aplicaré el algoritmo húngaro (minimizar la suma sobre está nueva matriz (*))

$(x+y) = 121$

$z = x+y-\epsilon$

$|z| = 10$

↓ resto min de la f.la y min columna

	I	II	III	IV	V	VI	VII	VIII	IX	
A	∞	∞	∞	0	1	1	∞	0	∞	IV
B	0	2	5	2	0	1	4	2	4	I
C	0	1	3	0	0	1	2	0	∞	V
D	∞	1	0	1	1	1	0	1	0	
E	∞	0	∞	3	1	0	4	3	4	VII
F	∞	1	3	0	1	1	3	0	3	VIII
G	∞	∞	∞	∞	∞	0	∞	∞	∞	*
H	0	1	3	0	1	1	3	0	3	*
I	∞	0	∞	2	2	0	∞	∞	∞	II
	~~H~~	~~E~~		~~H~~	~~B~~	~~G~~		~~H~~		

restes

resto

No hay matching perfecto en esta matriz sólo restando filas y columnas.

Por Hall $\exists\, S \subseteq$ filas.

$S = \{A, B, C, E, F, G, H, I\}$

$T = \{I, II, IV, V, VI, VIII\}$

$|S| = 8 \quad |\Gamma(S)| = |T| = 6$

y $6 < 8$ ∴ no puede haber matching perfecto de 0 en esta matriz.

Tacho S y $\Gamma(S)$ luego $\min S \times \overline{\Gamma(S)} = 2$

	I	II	III	IV	V	VI	VII	VIII	IX	
A	∞	∞	∞	0	1	1	∞	0	∞	IV
B	0	1	3	2	0	1	2	2	2	I
C	0	1	1	0	0	1	0	0	∞	V
D	∞	3	0	3	3	3	0	3	0	VII
E	∞	0	∞	3	2	0	2	3	2	VI
F	∞	1	1	0	1	1	1	0	1	VIII
G	∞	∞	∞	∞	∞	0	∞	∞	∞	*
H	0	1	1	0	1	1	1	0	1	*
I	∞	0	∞	2	2	0	∞	∞	∞	II
	~~H~~	~~E~~	D	~~H~~	~~B~~	~~G~~	~~C~~	~~H~~	D	
		~~E~~				~~G~~				

2 sumar
1 no hago nada
0 restar

No hay matching perfecto

Por T. Hall $\exists\, S \subseteq$ filas

$S = \{E, G, I\}$

$T = \{II, VI\}$

$|S| = 3 \quad |\Gamma(S)| = |T| = 2$

y $2 < 3$ ∴ no puede haber matching perfecto de 0 en esta matriz

Tacho S y $\Gamma(S)$ luego $\min S \times \overline{\Gamma(S)} = 2$.

	I	II	III	IV	V	VI	VII	VIII	IX	
A	∞	∞	∞	0	1	1	∞	0	∞	
B	0	4	3	2	0	3	2	2	2	V
C	0	3	1	0	0	3	0	0	∞	VII
D	∞	5	0	3	3	5	0	3	0	
E	∞	0	∞	1	0	0	0	1	0	VI
F	∞	3	1	0	1	3	1	0	1	
G	∞	∞	∞	∞	∞	0	∞	∞	∞	*
H	0	3	1	0	1	3	1	0	1	
I	∞	0	∞	0	0	0	∞	∞	∞	II
		~~E~~			~~E~~	~~G~~	~~E~~		~~E~~	

¡Hay matching perfecto!

A IV : 5
B V : 2
C VII : 4
D III : 4
E IX : 7
F VIII : 5
G VI : 6
H I : 7
I II : 6

Σ : 46

El trabajo se terminará en 7 días todos los trabajos y el costo total en pesos es de 46

Pregunta? Es válido ver el mínimo de columna III por ejemplo que es 9

⇒ umbral ≥ 9 y los posibles 9, 10, 11, 12, 15, 19?

↑ Probar con 11 si hay probar con 9 o 10

VI)

	I	II	III	IV
A	7	8	10	9
B	2	6	9	5
C	8	19	19	15
D	1	11	12	9

elementos = {1, 2, 5, 6, 7, 8, 9, 10, 11, 12, 15, 19}

umbral ≥ 8

	I	II	III	IV	
A	1	1	0	0	~~I~~
B	1	1	0	1	~~II~~
C	1	0	0	0	* ~~I~~
D	1	0	0	0	* ~~II~~
	~~C~~	~~A~~		B	
	~~D~~				

No hay matching perfecto

S = {C, D} t = {I} |S| = 2 |Γ(S)| = 1 1 < 2

∴ no puede haber matching perfecto

umbral ≥ 11

	I	II	III	IV	
A	1	1	1	1	~~I~~
B	1	1	1	1	~~II~~
C	1	0	0	0	*
D	1	1	0	1	
	~~C~~	~~A~~	A	A	

Hay matching perfecto

A III : 10
B II : 6
C I : 8
D IV : 9

Costo máx 10

No puede ser

umbral ≥ 9

	I	II	III	IV	
A	1	1	0	1	~~I~~
B	1	1	1	1	~~II~~
C	1	0	0	0	*
D	1	0	0	1	IV
	~~C~~	~~A~~	B	~~A~~	

Hay matching perfecto

A II : 8
B III : 9
C I : 8
D IV : 9

Costo máx 9

↳ mínimo umbral posible.

Ahora minimizo la suma con umbral = 9

	I	II	III	IV
A	7	8	∞	9
B	2	6	9	5
C	8	∞	∞	∞
D	1	∞	∞	9

→ resto min filas

	I	II	III	IV
A	0	1	∞	2
B	0	4	7	3
C	0	∞	∞	∞
D	0	∞	∞	8

→ resto min columna

	I	II	III	IV	
A	0	0	∞	0	~~I~~
B	0	3	0	1	
C	0	∞	∞	∞	* ~~I~~
D	0	∞	∞	6	* ~~II~~
	~~C~~	~~A~~		A.	
	~~D~~				

No hay MP por T. Hall ∃ S ⊆ filas S = {C, D} Γ(S) = {I}

y |S| = 2 y |Γ(S)| = 1 y 1 < 2 ∴ no puede haber M.P.

⇒ tacho S y Γ(S) min S × Γ(S) = 6

	I	II	III	IV	
A	6	0	∞	0	
B	6	3	0	1	
C	0	∞	∞	∞	~~I~~
D	0	∞	∞	0	*
	~~D~~			D	

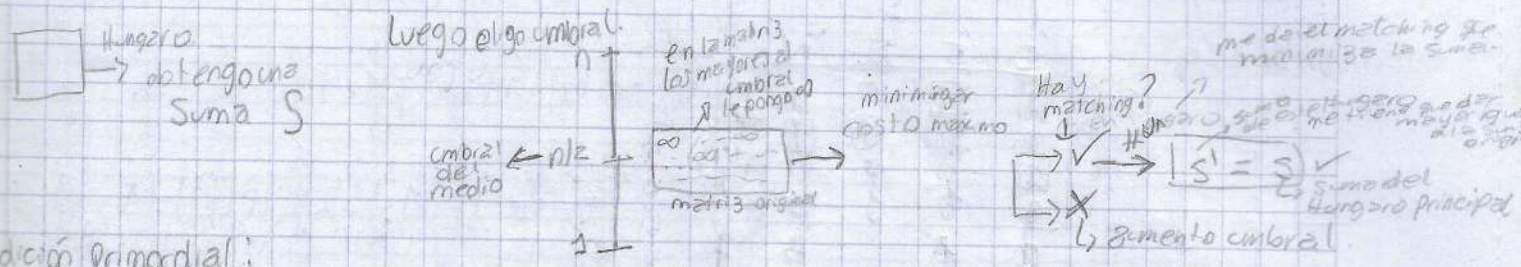
Hay matching perfecto

A II : 8
B III : 9
C I : 8
D IV : 9

$\boxed{\Sigma = 34}$

Ahora la suma total es 34 es mayor a la mínima suma total usando todos los pesos esto debíalo a la limitación de sólo usar pesos menores a 10. En el matching del ej IV c) están A III y C IV que son mayores a 9.

VII) Primero corremos el algoritmo húngaro para encontrar el valor **S** de la suma minimal y luego para encontrar el matching que minimice el máximo se escoge un umbral y se modifica la matriz inicial para reemplazar los valores mayores al umbral por ∞ y se encuentra el matching que minimice la suma total de esa matriz, si el valor de la suma (S') es igual a **S** entonces es el matching que buscamos, si no aumentemos el umbral y repetimos el proceso.



Condición primordial:
de los que minimiza la suma yo quiero el máximo más chico

- voy a subir o bajar el umbral dependiendo lo que me da el hungaro después de correr el algoritmo de costo teniendo en cuenta la suma, esto es si la suma obtenida acá es igual a la primera suma del algoritmo hungaro primero entonces busco un umbral más chico.

Cuando agregamos más lados la suma mínima va a bajar, si en algún umbral no vale $S'_{suma} = S^{suma}$ ⇒ todos los umbriales por debajo de él no va a valer. (es monotono) porque si agrego lados y me dio no ⇒ con menos lados no voy a poder encontrar Matching Perfecto. Cuando logro $S' = S$ en un umbral quiere decir que obtuve la suma mínima y la minimice al máximo

Algoritmo sería algo así

Sea M matriz

```
Costos = [X1, X2, ..., Xn] // donde están ordenados crecientemente.
  S = valor suma con Algoritmo Hungaro(M);
Contador = 0;
  Umbral = costos[0];
  while (true):
      if (∃ matching perfecto con costo[X1, X2, ..., Xn]):
          S' = valor suma algoritmo Hungaro(M)
          if (S' = S)
              return ¡matching perfecto!;
      else contador ++;
           umbral = Costos[contador];
      fi  fi
od
```

Aplico Algoritmo Húngaro

M=

	I	II	III	IV	V	VI	VII	VIII	IX
A	9	8	8	5	6	6	8	6	8
B	5	6	7	4	2	3	6	5	6
C	5	5	5	2	2	3	4	3	9
D	8	7	4	5	5	5	4	6	4
E	8	5	8	6	5	3	7	7	7
F	8	7	7	4	5	5	7	5	7
G	11	8	10	8	8	6	9	9	9
H	7	7	7	4	5	5	7	5	7
I	9	6	8	6	6	4	8	8	8

resto min de filas.

resto columna 3 · resto 2 · resto 1

	I	II	III	IV	V	VI	VII	VIII	IX	
A	1	1	3	$\boxed{0}$	1	1	3	0	3	IV
B	$\boxed{0}$	2	5	2	0	1	4	2	4	I
C	0	1	3	0	$\boxed{0}$	1	2	0	7	V
D	1	1	$\boxed{0}$	1	1	1	0	1	0	
E	2	$\boxed{0}$	5	3	2	0	4	3	4	II
F	1	1	3	0	1	1	3	$\boxed{0}$	3	VIII
G	2	0	4	2	2	$\boxed{0}$	3	2	3	VI
H	0	1	3	0	1	1	3	0	3	*
I	2	0	4	1	1	0	4	3	4	*
	II	I		II	III	I		II		

No hay matching perfecto sólo restando filas y columnas

Por T. Hall $\exists\, S \subseteq$ filas.

$S =$ filas etiqueteadas $= \{A, B, C, E, F, G, H, I\}$

$T =$ columnas etiqueteadas $= \{I, II, IV, V, VI, VIII\}$

$|S| = 8$ y $|\Gamma(S)| = 6$ y $6 < 8$ ∴ No puede haber matching de ceros perfecto en esta matriz.

Ahora tacho $\bar{S}$ y $\Gamma(S)$ y entonces

$\min S \times \overline{\Gamma(S)} = 2$

	I	II	III	IV	V	VI	VII	VIII	IX		
A	1	1	1	$\boxed{0}$	1	1	1	0	1	IV	
B	$\boxed{0}$	2	3	2	$\boxed{0}$	1	2	2	2	I	
C	0	1	1	0	$\boxed{0}$	1	$\boxed{0}$	0	5	V	
D	3	3	$\boxed{0}$	3	3	3	0	3	0		
E	2	$\boxed{0}$	3	3	2	0	2	3	2	II	II
F	1	1	1	0	1	1	1	$\boxed{0}$	1	VIII	
G	2	0	2	2	2	$\boxed{0}$	1	2	1	VI	VI
H	$\boxed{0}$	1	1	0	1	1	1	0	1	*	
I	2	0	2	2	2	0	2	3	2	*	*
	II	I		II	III	I	C	II			
		I				I					

No hay MP por T. Hall $\exists\, S \subseteq$ filas

$S = \{E, G, I\}$

$T = \{II, VI\}$

$|S| = 3$ $|\Gamma(S)| = 2$

y $2 < 3$ ∴ No puede haber matching de ceros perfecto en esta matriz.

tacho $\bar{S}$ y $\Gamma(S) \Rightarrow \min S \times \overline{\Gamma(S)} = 1$

	I	II	III	IV	V	VI	VII	VIII	IX	
A	1	2	1	$\boxed{0}$	1	2	1	0	1	
B	0	3	3	2	$\boxed{0}$	2	2	2	2	
C	0	2	1	0	0	2	$\boxed{0}$	0	5	VII
D	3	4	$\boxed{0}$	3	3	4	0	3	0	
E	1	$\boxed{0}$	2	2	1	0	1	2	1	II
F	1	2	1	0	1	2	1	$\boxed{0}$	1	
G	1	0	1	1	1	$\boxed{0}$	0	1	$\boxed{0}$	VI
H	$\boxed{0}$	2	1	0	1	2	1	0	1	
I	1	0	1	1	1	$\boxed{0}$	1	2	1	*
		I				I	G		G	

¡Hay matching perfecto!

A IV : 5
B V : 2
C VII : 4
D III : 4
E II : 5
F VIII : 5
G IX : 9
H I : 7
I VI : 4

Σ 45 → S

Costos = {2, 3, 4, 5, 6, 7, 8, 9, 10, 11}

umbral = 7 ya lo corrimos en ejercicio V y vimos que la suma total es 46, como la suma del matching que minimiza el total es 45. entonces tenemos que correr el algoritmo con un umbral más alto ($S' \neq S$)

tomo umbral 8 (los mayores a 8 pongo ∞)

	I	II	III	IV	V	VI	VII	VIII	IX
A	∞	8	8	5	6	6	8	6	8
B	5	6	7	4	2	3	6	5	6
C	5	5	5	2	2	3	4	3	∞
D	8	7	4	5	5	5	4	6	4
E	8	5	8	6	5	3	7	7	7
F	8	7	7	4	5	5	7	5	7
G	∞	8	∞	8	8	6	∞	∞	∞
H	7	7	7	4	5	5	7	5	7
I	∞	6	8	6	6	4	8	8	8

↓ Resto filas y columnas.

	I	II	III	IV	V	VI	VII	VIII	IX	
A	∞	1	3	0	1	1	3	0	3	IV
B	0	2	5	2	0	1	4	1	4	I
C	0	1	3	0	0	1	2	0	∞	V
D	1	1	0	1	1	1	0	1	0	
E	2	0	5	3	2	0	4	3	4	II
F	1	1	3	0	1	1	3	0	3	VIII
G	∞	0	∞	2	2	0	∞	∞	∞	VI
H	0	1	3	0	1	1	3	0	3	X
I	0	0	4	2	2	0	4	3	4	X
	II	I		IV	3	I		II		

No hay MP.
Por T. Hall ∃ S ⊆ filas.
S = {A, B, C, E, F, G, H, I}
T = {I, III, IV, V, VI, VIII}
|S| = 8 |Γ(S)| = 6 y 6 < 8
∴ No puede haber MP. de ceros en está matriz
tacho S̄ y Γ(S) min S×Γ(S) = 2

	I	II	III	IV	V	VI	VII	VIII	IX	
A	∞	1	1	0	1	1	1	0	1	IV
B	0	2	3	2	0	1	2	2	2	I
C	0	1	1	0	0	1	0	0	∞	V
D	3	3	0	3	3	3	0	3	0	
E	2	0	3	3	2	0	2	3	2	II III
F	1	1	1	0	1	1	1	0	1	VIII
G	∞	0	∞	2	2	0	∞	∞	∞	VI VII
H	0	1	1	0	1	1	1	0	1	X
I	∞	0	2	2	2	0	2	3	2	X X
	X	I		II		I		II		

No hay MP
Por T. Hall ∃ S ⊆ filas.
S = {E, G, I} |S| = 3 |Γ(S)| = 2
T = {II, VI} y 2 < 3 ∴ No puede haber MP de ceros en está matriz.
tacho S̄ y Γ(S) ⇒ min S×Γ(S) = 2

	I	II	III	IV	V	VI	VII	VIII	IX
A	∞	3	1	0	1	3	1	0	1
B	0	4	3	2	0	3	2	2	2
C	0	3	1	0	0	3	0	0	∞
D	3	4	0	3	3	5	0	3	0
E	0	0	1	1	0	0	0	1	0
F	1	3	1	0	1	2	1	0	1
G	∞	0	∞	0	0	0	∞	∞	∞
H	0	3	1	0	1	3	1	0	1
I	∞	0	0	0	0	0	0	1	0

¡Hay matching perfecto!

A IV : 5
B V : 2
C VII : 4
D VIII : 4
E II : 5
F VIII : 5
G VI : 6
H I : 7
I IX : 8

Σ 46 → S'

Como $S' \neq S$
$46 \neq 45$
repetimos algoritmo aumentando el umbral a 9

Umbral = 9 (mayor a 9 le pongo ∞)

	I	II	III	IV	V	VI	VII	VIII	IX
A	9	8	8	5	6	6	8	6	8
B	5	6	7	4	2	3	6	5	6
C	5	5	5	2	2	3	4	3	9
D	8	7	4	5	5	5	4	6	4
E	8	5	8	6	5	3	7	7	7
F	8	7	7	4	5	5	7	5	7
G	∞	8	∞	8	8	6	9	9	9
H	7	7	7	4	5	5	7	5	7
I	9	6	8	6	6	4	8	8	8

↓ resto min fila y columna

	I	II	III	IV	V	VI	VII	VIII	IX	
A	1	1	3	0	1	1	3	0	3	IV
B	0	2	5	2	0	1	4	2	4	I
C	0	1	3	0	0	1	2	0	7	V
D	1	1	0	1	1	1	0	1	0	
E	2	0	5	3	2	0	4	3	4	II
F	1	1	3	0	1	1	3	0	3	VIII
G	∞	0	∞	2	2	0	3	2	3	VI
H	0	1	3	0	1	1	3	0	3	
I	2	0	4	2	2	0	4	3	4	
	H	I		H	B	I		H		

No hay matching perfecto
Por t. Hall ∃ S ⊆ filas
S = {A, B, C, E, F, G, H, I}
T = {I, II, IV, V, VI, VIII}
|S| = 8 |Γ(S)| = 6. y 6<8 ∴ no
Puede haber MP de 0 en esta matriz
Tacho S̄ y Γ(S) ⇒ min S×Γ(S) = 2

	I	II	III	IV	V	VI	VII	VIII	IX	
A	1	1	1	0	1	1	1	0	1	IV
B	0	2	3	2	0	1	2	2	2	I
C	0	1	1	0	0	1	0	0	5	V
D	3	3	0	3	3	3	0	3	0	
E	2	0	3	3	2	0	2	3	2	II
F	1	1	1	0	1	1	1	0	1	VIII
G	∞	0	∞	2	2	0	1	2	1	VI
H	0	1	1	0	1	1	1	0	1	
I	2	0	2	2	3	0	2	3	2	
	H	I		H	B	I	C	H		
		I				I				

No hay matching perfecto
Por T. Hall ∃ S ⊆ filas
S = {E, G, I}
T = {II, VI}
|S| = 3 |Γ(S)| = 2 y 2<3
∴ No puede haber matching perfecto de ceros en esta matriz
Tacho S̄ y Γ(S) ⇒ min S×Γ(S) = 1

	I	II	III	IV	V	VI	VII	VIII	IX	
A	1	2	1	0	1	2	1	0	1	
B	0	3	3	2	0	2	2	2	2	
C	0	2	1	0	0	2	0	0	5	VII
D	3	4	0	3	3	4	0	3	0	
E	1	0	2	2	1	0	1	2	1	II
F	1	2	1	0	1	2	1	0	1	
G	∞	0	∞	1	1	0	0	1	0	VI
H	0	2	1	0	1	2	1	0	1	
I	1	0	1	1	2	0	1	2	1	
		I				I	G		G	

¡Hay matching perfecto!

A IV : 5
B V : 2
C VII : 4
D III : 4
E II : 5
F VIII : 5
G IX : 9
H I : 7
I VI : 4

Σ = 45 → S'

Finalmente este es el matching que buscamos ya que coincide S con S' (S' = S)
Suma del primer hungaro con el hungaro con umbral = 9
Entonces la suma es minima y la minimiza al máximo ■

VIII) ¿Cuál es la complejidad del algoritmo que resuelve el problema de minimizar la suma de entre todos los matchings que minimizan el máximo? (ej: 5)

Primero se realizan $O(\log(n))$ ecuaciones (caso de busqueda binaria para encontrar los umbrales) del algoritmo de minimizar el máximo el cual tiene complejidad $O(n^{5/2})$

$$\Rightarrow O(\log(n)) \cdot O(n^{5/2}) = \boxed{O(n^{5/2} \cdot \log(n))}$$

Luego cuando encuentro el mínimo maximal corremos el algoritmo húngaro el cual tiene una complejidad es $O(n^3)$ (codificación de cambio de matriz en $O(n)$).

$$\Rightarrow \boxed{O(n^{5/2} \cdot \log(n)) + O(n^3) = \text{Complejidad total}}$$

IX) ¿Cuál es la complejidad del algoritmo que resuelve el problema de minimizar el máximo de todos los matching que minimizan la suma? Ej: VII.

Primero para calcular la suma mínima corremos el algoritmo húngaro el cual tiene complejidad $O(n^4)$, luego lo que hacemos es busqueda binaria para los umbrales esto tiene una complejidad de $O(\log(n))$ ejecuciones del algoritmo húngaro (en el peor de los casos) es decir que funcione recién en el último umbral por visitar) esto sería $O(\log(n)) \cdot O(n^3) = O(n^3 \log(n))$

$$\Rightarrow \boxed{O(n^3) + O(n^3 \log(n)) = \text{Complejidad total}}$$

X) ganancias que la empresa X obtendrá al asignar los trabajos I, II, etc a los trabajadores A, B, ..., etc. Me piden si la ganancia total de la empresa X sea la máxima Asignando todos los trabajos.

- nos piden que maximicemos la suma total, nosotros no tenemos un algoritmo que realice esto pero podriamos adaptar el problema a "minimizar la suma" y usar el algoritmo húngaro para esto deberiamos hacer una modificación vamos a restar el máximo en cada fila sin tenendo en cuenta el signo (-) y restar el mínimo en cada columna y aplicar luego el algoritmo húngaro a esa matriz de esta manera el matching que vamos a encontrar va a ser el que maximice la suma total.

restamos max de fila

a)

	I	II	III	IV	V	VI	VII
A	8	1	7	3	9	3	1
B	5	7	3	4	8	7	1
C	8	4	8	8	8	8	7
D	1	1	7	9	8	4	8
E	7	7	8	8	7	7	9
F	8	7	7	1	9	8	3
G	8	1	8	7	8	4	3

	I	II	III	IV	V	VI	VII
A	1	8	2	6	0	6	8
B	3	1	5	4	0	1	7
C	0	4	0	0	0	0	1
D	8	8	2	0	1	5	1
E	2	2	1	1	2	2	0
F	1	2	2	8	0	1	6
G	0	7	0	1	0	4	5

ahora restamos min de c/columna

	I	II	III	IV	V	VI	VII	
A	1	7	2	6	$\boxed{0}$	6	8	~~I~~
B	3	$\boxed{0}$	5	4	0	1	7	
C	$\boxed{0}$	3	0	0	0	0	1	
D	8	7	2	$\boxed{0}$	1	5	1	
E	2	1	1	1	1	2	$\boxed{0}$	
F	1	1	2	8	0	1	6	*
G	0	6	$\boxed{0}$	1	0	4	5	

No hay MP. sólo restando filas y columnas.
Por T. Hall $\exists\, S \subseteq$ filas
$S = \{A, F\}$
$T = \{V\}$
$|S| = 2 \quad |\Gamma(S)| = 1$
y $1 < 2$ ∴ no puede haber matching de ceros perfecto en está matriz.

Tacho $\bar{S}$ y $\Gamma(S)$ ⇒ min $S \times \overline{\Gamma(S)} = \boxed{1}$

	I	II	III	IV	V	VI	VII	
A	0	6	1	5	$\boxed{0}$	5	7	V
B	3	$\boxed{0}$	5	4	1	1	7	II
C	$\boxed{0}$	3	0	0	1	0	1	I
D	8	7	2	$\boxed{0}$	2	5	1	
E	2	1	1	1	3	2	$\boxed{0}$	
F	0	0	1	7	0	$\boxed{0}$	5	*
G	$\boxed{0}$	6	$\boxed{0}$	1	1	4	5	
	~~F~~	~~F~~			~~F~~	F		

¡Hay matching perfecto!

A V : 9
B II : 7
C I : 8
D IV : 9
E VII : 9
F VI : 8
G III : 8
Σ $\boxed{58}$

b) Voy a copiar la matriz directamente restando el máximo de la fila

	I	II	III	IV	V	VI	VII	
A	4	$\boxed{0}$	4	1	$\boxed{0}$	0	0	~~II~~
B	$\boxed{0}$	5	2	1	5	4	1	
C	8	4	3	3	8	2	$\boxed{0}$	
D	4	4	$\boxed{0}$	1	3	3	8	~~III~~
E	5	5	5	$\boxed{0}$	5	4	1	~~IV~~ IV
F	8	$\boxed{0}$	0	1	4	2	5	*
G	8	4	8	0	3	3	5	*
	~~F~~	~~F~~	~~G~~	A	A	A		

resto el min = 1

No hay MP sólo restando filas y columnas.
Por T. Hall $\exists\, S \subseteq$ filas
$S = \{E, G\} \quad |S| = 2 \quad 1 < 2$ ∴ no puede haber MP de 0 en esta matriz
$T = \{IV\} \quad |\Gamma(S)| = 1$

tacho $\bar{S}$ y $\Gamma(S)$ ⇒ min $S \times \overline{\Gamma(S)} = \boxed{1}$

tacho S y $\Gamma(S)$

min $S \times \overline{\Gamma(S)} = \boxed{2}$

	I	II	III	IV	V	VI	VII	
A	4	0	4	1	$\boxed{0}$	0	0	
B	$\boxed{0}$	5	2	1	5	4	1	
C	8	4	3	4	8	2	$\boxed{0}$	VII
D	4	4	$\boxed{0}$	1	3	3	8	
E	4	4	4	$\boxed{0}$	4	3	0	~~IV~~
F	8	$\boxed{0}$	0	1	4	2	5	
G	7	3	7	0	2	2	4	*
			G	G		E		

no hay MP por T. Hall ∃ S ⊆ filas $S = \{C, E, G\}$
$T = \{IV, VII\} \quad |S| = 3 \quad |\Gamma(S)| = 2$ y $2 < 3$ ∴ no puede haber MP de ceros en esta matriz

A IV	5
B I	6
C VII	9
D III	9
E IV	6
F II	9
G VI	5
Σ	$\boxed{49}$

	I	II	III	IV	V	VI	VII
A	4	0	4	4	$\boxed{0}$	$\boxed{0}$	2
B	$\boxed{0}$	5	2	4	5	4	3
C	6	2	1	4	6	0	$\boxed{0}$
D	4	4	$\boxed{0}$	7	3	3	10
E	2	2	2	$\boxed{0}$	2	1	0
F	8	$\boxed{0}$	0	7	4	2	7
G	5	1	5	0	0	$\boxed{0}$	4

¡Hay Matching Perfecto!

c) (resto 1)

	I	II	III	IV	V	VI	VII	VIII	
A	0	1	1	2	6	3	8	1	I
B	0	3	4	2	7	4	1	6	~~A~~
C	1	2	0	0	5	1	1	4	
D	1	0	1	2	2	4	1	0	
E	4	4	6	2	4	4	0	4	
F	6	5	4	3	0	1	8	1	
G	5	2	4	4	1	0	7	4	VI
H	5	5	1	1	3	0	7	2	~~H~~
	~~B~~					~~H~~			

No hay MP. Solo restando filas y columnas, por T. Hall $\exists$ $S \subseteq$ filas

$S = \{A, B, G, H\}$
$T = \{I, VI\}$
$|S| = 4$ $|\Gamma(S)| = 2$.
y $2 < 4$. $\therefore$ No puede haber MP de ceros en esta matriz.

tacho S y $\Gamma(S)$ $\Rightarrow$ min $S \times \Gamma(S) = 1$

	I	II	III	IV	V	VI	VII	VIII	
A	0	0	1	1	5	3	7	0	~~I~~
B	0	2	3	1	6	4	0	5	~~A~~
C	2	2	0	0	4	2	1	4	
D	2	0	1	2	1	5	1	0	II
E	5	4	5	2	3	5	0	4	~~VII~~
F	7	5	4	3	0	3	8	1	
G	5	1	3	3	0	0	6	3	
H	5	4	0	0	2	0	6	1	
	~~B~~	A					~~B~~	A	

¡Hay Matching perfecto!

A	VIII	8
B	I	9
C	III	7
D	II	8
E	VII	9
F	~~VI~~	9
G	VI	8
H	IV	8
		Σ 66

Falta el X y XII

Primero niego todo, luego resto el máximo +1 a cada

XI) a)

	I	II	III	IV	V	VI	VII
A	2	9	3	7	1	7	9
B	4	2	6	5	1	2	8
C	1	5	1	1	1	1	2
D	9	9	3	1	2	6	2
E	3	3	2	2	3	3	1
F	2	3	3	9	1	2	7
G	1	8	1	2	1	5	6

resto min colum $\rightarrow$

	I	II	III	IV	V	VI	VII
A	1	7	2	6	0	6	8
B	3	0	5	4	0	1	7
C	0	3	0	0	0	0	1
D	8	7	2	0	1	5	1
E	2	2	1	1	2	2	0
F	1	2	2	8	0	1	6
G	0	6	0	1	0	4	5

... y luego correr húngaro normalmente hecho atrás

Nota: Si yo elijo restar el máximo en la fila y el mínimo en la columna es lo mismo si hiciera pongo a todos negativo y luego resto el máximo de c/fila más 1).

XII) umbral = 5

	I	II	III	IV	V	VI	VII	
A	1	0	1	0	1	0	0	I
B	1	1	0	0	1	1	0	II
C	1	0	1	1	1	1	1	~~III~~
D	0	0	1	1	1	0	1	IV
E	1	1	1	1	1	1	1	~~V~~
F	1	1	1	0	1	1	0	VI
G	1	0	1	1	1	0	0	~~X~~
	~~G~~	~~E~~	~~G~~	~~G~~	~~G~~	~~C~~	C	

Elementos $\{1, 3, 4, 5, 7, 8, 9\}$

Al encontrar la máxima ganancia mínima los mayores al umbral lo pongo en 1.

C.C en 0

Hay M. perfecto.

A	I	8
B	II	7
C	VII	7
D	IV	9
E	V	7
F	VI	8
G	III	8

Costo: 7

Umbral 8 → No hay matching perfecto porque columna II no tendría 1's

	I	II	III	IV	V	VI	VII
A	1	0	0	0	1	0	0
B	0	0	0	0	1	0	0
C	1	0	1	1	1	1	0
D	0	0	0	1	1	0	1
E	0	0	1	1	0	0	1
F	1	0	0	0	1	1	0
G	1	0	1	0	1	0	0

Umbral 7

	I	II	III	IV	V	VI	VII	
A	1	0	1	0	1	0	0	I
B	0	1	0	0	1	1	0	II
C	1	0	1	1	1	1	1	III
D	0	0	1	1	1	0	1	IV
E	1	1	1	1	1	1	1	V
F	1	1	1	0	1	1	0	VI
G	1	0	1	1	1	0	0	*
	G	E	G	G	G	C	C	

Hay matching perfecto

A I : 8
B II : 7
C VII : 7 costo: 7
D IV : 9
E V : 7
F VI : 8
G III : 8

El matching que maximiza la ganancia mínima es con el umbral 7 y la ganancia mínima es 7.

por cami

VIII 1) Correr el Alg de minimizar el máximo costo → $O(n^3 \cdot \log(n))$

2) Poner ∞ en los costos mayores al mín costo máx que obtuve en el paso anterior. Esto es recorrer la matriz $(O(n^2))$ y setear en ∞ los costos de la matriz que cumplen la cond $(O(1))$ → $O(n^2) \cdot O(1) = O(n^2)$

3) Correr el algoritmo Húngaro en esta nueva matriz → $O(n^4)$

Compl. Total : Comp(1) + comp(2) + Comp(3)

$\rightarrow O(n^3 \log(n)) + O(n^2) + O(n^4) = \boxed{O(n^4)}$

IX 1) Correr el Algoritmo Húngaro → $O(n^4)$

2) Encontrar otra suma mínima pero con menor costo máximo. Esto implica hacer una búsq binaria $(O(\log(m))$ → sin incluir los elem mayor al máx costo del paso ant. donde $m = \#\{$costo menores al máx costo encontrado en 1$\}$

Hacer 2 chequeos a) ver que $\exists$ MP para ese umbral → $O(n^3)$ (una iteración del Alg. de minimizar el máx costo)

b) correr el alg húngaro para ver si la suma es mínima → $O(n^4)$

Notar: que el paso 2b no se realiza si no se encuentra MP en el paso 2a

Compl. total : Comp(1) + Comp(2a) + Comp(2b) = $O(n^4) + O(\log(m) \cdot n^3) + O(\log(m) \cdot n^4) = O(\log(m))$

2b) Elementos Matr.: {1, 4, 5, 6, 9}

Umbral = 5

	I	II	III	IV	V	VI	VII
A	0	$\boxed{1}$	0	0	1	0	1
B	$\boxed{1}$	0	0	1	0	0	1
C	0	1	$\boxed{1}$	1	0	1	1
D	1	1	1	$\boxed{1}$	1	1	0
E	0	0	0	1	0	0	$\boxed{1}$
F	0	1	1	1	$\boxed{1}$	1	0
G	0	1	0	1	1	$\boxed{1}$	0

Hay M. Perfecto

A II : 5
B I : 6
C III : 6
D IV : 5
E VII : 5
F V : 5
G VI : 5

Umbral = 6 no puede tener M. perfecto porque la fila A no tendrá 1's

Umbral = 6 " " " M. perfecto porque la fila A no tendría 1's.

∴ matching que maximiza la ganancia mínima es el obtenido con Umbral 5 y el costo mínimo sería 5.

c) Elem de la matriz: {1, 2, 3, 4, 5, 6, 7, 8, 9}

	I	II	III	IV	V	VI	VII	VIII	
A	$\boxed{1}$	1	1	1	0	1	0	1	~~I~~
B	1	$\boxed{1}$	0	1	0	1	$\boxed{1}$	0	~~II~~
C	1	1	$\boxed{1}$	1	0	1	1	0	
D	1	1	1	$\boxed{1}$	1	0	1	1	IV
E	1	1	0	1	$\boxed{1}$	1	1	1	~~V~~
F	0	0	0	1	1	$\boxed{1}$	0	1	~~VI~~
G	0	$\boxed{1}$	0	0	1	1	0	0	*
H	0	0	1	1	1	1	0	$\boxed{1}$	
	~~B~~	~~G~~		~~B~~	~~G~~	~~G~~	B	E	

Hay M. Perfecto

A I : 9
B VII : 8
C III : 7
D IV : 6
E V : $\boxed{5}$
F VI : 7
G II : 6
H VIII : 7

Umbral 7

	I	II	III	IV	V	VI	VII	VIII
A	$\boxed{1}$	1	1	1	0	0	0	1
B	1	0	0	$\boxed{1}$	0	0	1	0
C	1	0	$\boxed{1}$	1	0	1	1	0
D	1	$\boxed{1}$	0	0	0	0	1	1
E	0	0	0	1	0	0	$\boxed{1}$	0
F	0	0	0	0	$\boxed{1}$	1	0	1
G	0	0	0	0	1	$\boxed{1}$	0	0
H	0	0	1	1	0	1	0	$\boxed{1}$

Hay M. perfecto

A I : 9
B IV : 7
C III : 7
D II : 8
E VII : 9
F V : 9
G VI : 8
H VIII : 7

Costo mín: $\boxed{7}$

∴ El matching que minimiza la ganancia mínima es el obtenido con umbral 7 y el costo mínimo sería 7

Proba Grafo bipartito $G=(X \cup Y, E)$ con $|X| = |Y| = 11$ y $|E| = 111$ tiene MP.

(X) Sea A el subconj más grande de X tq $|\Gamma(a)| \neq 11$ $\forall a \in A$

Es claro q! el vértice q! tenga, la mínima cant de vecinos en X estará en A. Le llamaremos V_{min} y a su cant de vecinos le llamaremos A

Propongo que el mín. K posible es igual a $|A|$. Demo: Como todos los vértices en $X-A$, tienen 11 vecinos, entonces ahí están $(11-|A|)*11$ de los 111 lados que hay en total.

Por lo tanto de A salen $111-(11-|A|)*11$. Simplificando tenemos $111-121+11|A| = 11|A|-10$

Como nadie en A tiene 11 vértices, para llegar al mínimo K posible hagamos q! todos los vértices de A tengan 10 vecinos excepto V_{min}

En este caso saldrían $10*(|A|-1)+K$ lados de A.

Pero $10*(|A|-1)+|A| = 11|A|-10$ ya que $11*|A|-10 = 1|A|+10|A|-10 = |A|+10(|A|-1)$ Por lo tanto $K=|A|$ $\forall B \subseteq A$ pasa: 1) $|\Gamma(B)| \geq K$

Por lo tanto $|\Gamma(B)| \geq K \geq$ min K posible $=|A| \geq |B|$ 2) $|A| \geq |B|$

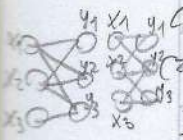
Como B era cualq subconj de A, vemos q se cumplen la cond de Hall para todos y debe existir un MP. No hace falta borrar conj fuera de A ya que si incluimos cualq elemento de $X-A$ entonces $|\Gamma(X-A)| = 11$ y sigue valiendo la cond de Hall.

Como $|X|=|Y|=11 \Rightarrow$ ese Matching también es perfecto.

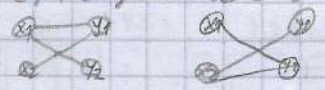
(X) Por cami! Sea $G=(X \cup Y, E)$ con $|X|=n=|Y|$ con $|E|=n^2-n+1$ la idea es proba que $\exists$ MP para $|E|=n^2-n+1$

CASO BASE: $n=0 \rightarrow G$ vacío $\rightarrow$ no hay matching

CASO BASE $n=1 \rightarrow |E| = 1^2-1+1=1$, i.e, tengo un lado entre 2 nodos $\Rightarrow$ MP.



CASO BASE $n=2 \rightarrow |E| = 2^2-2+1=3$ G



CASO " $n=3 \rightarrow |E| = 3^2-3+1=7$

Invariante (a ojo): Siempre q tengo al menos un vértice sin la cant máx de lados, siempre tengo a lo sumo $n-1$ vértices con la cant máx de lados] p/ G bipartito

HI: Hay Matching perfecto para $|X|=n=|Y|$ y $|E|=n^2-n+1$

Queremos ver que vale para $n+1$

$|E| = (n+1)^2-(n+1)+1 = n^2+2n+1-n = n^2+n+1$. Por HI sabemos que n^2-n+1 aristas forman un matching entre los vértices $x_1, \dots, x_n$ e $y_1, \dots, y_n$ por lo que tenemos $\underbrace{n^2+n+1}_{|E|} - \underbrace{(n^2-n+1)}_{\text{lados del M. ant}} = 2n$ lados para asignar

notar: La división de $|E|/(n+1)$ tiene cociente n y resto 1, lo que nos indica que siempre va a haber al menos un nodo con $n+1$ lados en X y uno en Y. Por lo tanto y como G es bipartito hay 2 casos que nos importa en como asignamos los $2n$ aristas nuevas.

a) $\exists$ una arista entre los nodos x_{n+1} e y_{n+1}

b) $\nexists$ una arista entre los nodos x_{n+1} e y_{n+1}

Prueba A: Puedo matchear x_{n+1} con y_{n+1}

Prueba B: Como $\nexists$ lado entre x_{n+1} e y_{n+1}, esto nos dice que $\exists$ vértices x_i, y_j con $i,j=1,\dots,n$ tales que $\delta(x_i)=n+1=\delta(y_j) \Rightarrow 0 < \delta(x_{n+1}), \delta(y_{n+1}) < n+1$

Si matcheo x_{n+1} con y_j tq $\delta(y_j)=n+1$, entonces tengo que quitar del matching el lado $x_k y_j$ con $k=1,\dots,n$ si $x_k y_{n+1} \in E$ los matcheo y tengo matching perfecto. Sino, matcheo el lado $x_k y_j$, y desmatcheo el lado $x_{k'} y_{j'}$ con $j,k'=1,\dots,n$ y me fijo si puedo agregar $x_{k'} y_{n+1}$ al matching. Puedo realizar este proceso un máximo de n veces xq $\delta(y_{n+1}) \geq 1$ y eventualmente incluir y_{n+1} al matching.

El caso en el que agrego al matching primero el lado $x_i y_{n+1}$ con $\delta(x_i)=n+1$ y luego acomodo x_{n+1} es análogo ∎

Práctico 7: Códigos

1) encontrar nº de errores

a) $C_1 = \{0000000, 1111110, 1010100, 0101010\}$

$\boxed{C_1}$

dH	0000000	1111110	1010100	0101010
0000000	0	6	3	3
1111110	6	0	3	3
1010100	3	3	0	6
0101010	3	3	6	0

$\boxed{\delta(C_1) = 3}$

C_1 detecta $\delta - 1$ errores

$\Rightarrow \delta(C_1) = 3 \rightarrow 3 - 1 = 2$ Detecta 2 errores

$t = \left[\frac{\delta - 1}{2}\right] \Rightarrow$ (corrige t errores.

$\hookrightarrow \frac{3-1}{2} = \boxed{1} \rightarrow$ corrige 1 error.

transpuesta

C_2 $\{0000000, 1111111, 111100$

dH	0000000	1111111	1111000	0000111
0000000	0	7	4	3
1111111	7	0	3	4
1111000	4	3	0	7
0000111	3	4	7	0

$\delta(C_2) = 3$

C_2 detecta $\delta - 1$ errores

$\delta(C_2) = 3 \Rightarrow 3 - 1 = 2$ detecta 2 error

$t = \left[\frac{\delta - 1}{2}\right] \Rightarrow \frac{3-1}{2} = \boxed{1}$ corrige 1 error

$\boxed{C_3}$

dH	000000000	111111111	111100000	000011111	101010101	010101010
000000000	0	9	4	5	5	4
111111111	9	0	5	4	4	5
111100000	4	5	0	9	5	4
000011111	5	4	9	0	4	5
101010101	5	4	5	4	0	9
010101010	4	5	4	5	9	0

$\delta(C_3) = 4$ C_3 detecta $\delta(C_3) = 4 \Rightarrow 4 - 1 = \boxed{3 \text{ errores detecta}}$ $t = \left[\frac{\delta - 1}{2}\right] \Rightarrow \frac{4-1}{2} = \frac{3}{2} = \boxed{1}$ corrige 1 error.

$\binom{8}{2}$

(II) Se tiene un CB de long 8 tq dist. mínima entre palabras es 5. Dar CS para el nº de palabra del código

$C \subseteq \{0,1\}^8$

$n = 8$

$\delta = 5$ $\quad t = \left[\frac{\overset{\to 5}{\delta} - 1}{2}\right] = \boxed{2}$

$\Rightarrow |C| \leq \frac{2^8}{1+8+\binom{8}{2}} \leq \frac{256}{1+8+28} \leq \frac{256}{37} \approx \overset{\to 6}{6{,}91}$

$\Rightarrow \boxed{|C| \leq 6}$ → Cota Superior es 6

En código lineal la cota superior es la misma pero $|C| = 2^K$ para algún K tengo que buscar el 2^K que este más cercano de 6 $\Rightarrow |C| = 2^2 = 4$

$\Rightarrow$ cota es $\boxed{4}$

(3) Se tiene un códi binario de long 16 tq la dist mínima entre palabras es 7. Dar Cota pl el nº de palabras del código. Da una cota sup si además se requiere que el código sea lineal.

$n = 16$ $\quad C \subseteq \{0,1\}^{16}$

$\delta = 7$ $\quad t = \left[\frac{\delta - 1}{2}\right] = \left[\frac{7-1}{2}\right] = \boxed{3}$

$\Rightarrow |C| \leq \frac{2^{16}}{1+16+\binom{16}{2}+\binom{16}{3}} \leq \frac{65536}{1+16+120+560} \leq 94{,}02 \approx 94$

$\Rightarrow |C| \leq 94$ → Cota Superior es 94

- En código lineal la cota sup es la misma pero $|C| = 2^K$ para algún K

$\Rightarrow 2^6 = 64$ Cota es $\boxed{64}$

(4) Probar que todo código binario lineal con parámetros $(n, K, \delta) = (23, \overset{\swarrow 12}{11}, 7)$ es perfecto.

$\Updownarrow$ es igual a la cota de Hamming

- Código binario lineal es perfecto si: $|C| = \frac{|A|}{\sum_{i=0}^{t}\binom{n}{i}} = \frac{2^n}{\sum_{i=0}^{t}\binom{n}{i}}$

$n = 23$ $\quad t = \left[\frac{\delta - 1}{2}\right] = \left[\frac{7-1}{2}\right] = [3] = \boxed{3}$

$K = 12$

$\delta = 7$ $\quad t = 3$

cota de Hamming $\quad |C| = \frac{2^{23}}{1+23+\binom{23}{2}+\binom{23}{3}} = \frac{8388608}{1+23+253+1771} = \boxed{4096}$

$|C| = 2^K = 2^{12} = \boxed{4096}$

=

(5)

$$G = \left[\begin{array}{cccc|ccccc} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 \end{array}\right]$$

y C el código generado por G.

a) ¿Cuál es la long de C? ¿Cuál es su dimensión?

- la longitud es el nº de columnas. $\Rightarrow$ RTa = $\boxed{9}$
- Dimensión del código es K y K es el nº de filas de G

$\Rightarrow$ Dimensión es $\boxed{5}$

b) Supongamos que desea mandar el msj 10101 11010 00111.

¿Cuales son las palabras del código que debería usar para mandar el mensaje?

$$[10101] * G = (1.1+0.0+1.0+\overset{1}{0.0+1.0})\ (1.0+0.1+1.0+\overset{0}{0.0+1.0})\ (1.0+0.0+1.1+\overset{1}{0.0+1.0})$$
$$(1.0+0.0+1.0+\overset{0}{0.1+1.0})\ (1.1+0.1+1.0+\overset{1}{0.1+1.0})\ (1.1+0.0+1.1+0.1+1.1)$$
$$(1.0+0.1+\overset{1}{1.1+0.1+1.0})\ \text{Falta}\ (1.0+0.0+1.0+\overset{0}{0.0+1.1})$$
$$(1.0+0.0+1.1+0.0+1.1)$$
$$= \boxed{101011110}$$

$[11010] * G = \boxed{110101000}$

$[00111] * G = \boxed{001111100}$

Las palabras del código que debería usar para mandar el mensaje sería

101011110, 110101000, 001111100.

c) Dar matriz de chequeo de C. Como $G = [I \mid A] \Rightarrow H = [A^T \mid I]$

$$H = \left[\begin{array}{ccccc|cccc} 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

d) Calcular $\delta(C)$

$|x| = d_H(x, \vec{0}) = \#(1\text{'s en } x)$ peso de hamming

$\delta(C) = \min\{|x| : x \in C \wedge x \neq 0\}$ / $\delta(C)$ = mínimo nº de columnas LD de H

$3 \leq \delta(C) \leq 4$ (3 1's, 4 1's)

- H no tiene columna 0 ni columnas repetidas.

$H^{(i)} \neq 0 \ \forall i \quad H^{(i)} \neq H^{(j)} \ \forall i \neq j \Rightarrow \delta(C) \geq 3$

$|H^{(i)}| \geq 3 \ \forall i \leq 6 \Rightarrow H^{(i)} \neq H^{(j)} + H^{(r)}$ con $j, r > 6$

(vec)

dominio está la identidad

$$[H^1] = \begin{bmatrix}1\\1\\0\\0\end{bmatrix} = \underbrace{\begin{bmatrix}1\\0\\0\\0\end{bmatrix} + \begin{bmatrix}0\\1\\0\\0\end{bmatrix}}_{\begin{bmatrix}1\\1\\0\\0\end{bmatrix}\text{ peso } 2} \quad H^6 + H^7$$

$\therefore \boxed{\delta(C) = 3}$ pues $H^1 + H^6 + H^7 = 0$

d) Supongamos que se reciben las palabras 10011001 y 110000001. ¿Cuáles son las palabras más probables que se hayan mandado? ¿A qué msj corresponden?

$w_1 = 10011001$ $\qquad$ $w_2 = 110000001$

$$H w_1^T = \begin{bmatrix} & & \\ & H & \\ & & \end{bmatrix} \cdot \begin{bmatrix}1\\0\\0\\1\\1\\1\\0\\0\\1\end{bmatrix} = H^{(1)} + H^{(4)} + H^{(5)} + H^{(6)} + H^{(9)}$$

$$= \begin{bmatrix}1\\1\\0\\0\end{bmatrix} + \begin{bmatrix}1\\1\\1\\0\end{bmatrix} + \begin{bmatrix}0\\1\\0\\1\end{bmatrix} + \begin{bmatrix}1\\0\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\1\end{bmatrix} = \begin{bmatrix}1\\1\\1\\0\end{bmatrix}$$

Como resultado obtenemos H^4, la palabra que es más posible que se haya mandado es: $v_1 = w_1 + e_4 = 10011001 + 000100000$

$$= \underbrace{\boxed{10001}}_{\text{palabra}}\boxed{1001}$$

Como en la matriz generadora los primeros 5 columnas corresponden a la identidad entonces los primeros 5 bits del código son el mensaje.

$$\boxed{10001}$$

$$H w_2^T = \begin{bmatrix} & & \\ & H & \\ & & \end{bmatrix} \cdot \begin{bmatrix}1\\1\\0\\0\\0\\0\\0\\0\\1\end{bmatrix} = H^{(1)} + H^{(2)} + H^{(9)}$$

$$\begin{bmatrix}1\\1\\0\\0\end{bmatrix} + \begin{bmatrix}1\\0\\1\\0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\1\end{bmatrix} = \begin{bmatrix}0\\1\\1\\1\end{bmatrix}$$

resultado obtenemos H^3, la palabra que es más posible q' se haya mandado es:

$v_2 = w_2 + e_3 = 110000001 + 001000000$ → palabra $\boxed{11100}$

$= \boxed{111000001}$

idem $\Rightarrow$ El mensaje más probable que se haya enviado $\boxed{10001\ 11100}$

f) recibe 001111010? y 011100011?

$$H \times [001111010]^T = H \times \begin{bmatrix} 0\\0\\1\\1\\1\\1\\0\\1\\0 \end{bmatrix} = H^{(3)} + H^{(4)} + H^{(5)} + H^{(6)} + H^{(8)}$$

$$= \begin{bmatrix} 0\\1\\1\\1 \end{bmatrix} + \begin{bmatrix} 1\\1\\1\\0 \end{bmatrix} + \begin{bmatrix} 0\\1\\0\\1 \end{bmatrix} + \begin{bmatrix} 1\\0\\0\\0 \end{bmatrix} + \begin{bmatrix} 0\\0\\1\\0 \end{bmatrix}$$

$$t = \left[\frac{\delta(c)-1}{2}\right] = \left[\frac{3-1}{2}\right] = \boxed{1}$$

↳ corrige

$$= \begin{bmatrix} 0\\1\\1\\0 \end{bmatrix}$$ → Este resultado no corresponde a una columna de la matriz de chequeo, sabemos que el código corrige 1 error, pero no se puede corregir esto quiere decir que ocurrieron más de 1 error. ✓

$$H \times [011100011]^T = H \times \begin{bmatrix} 0\\1\\1\\1\\0\\0\\0\\1\\1 \end{bmatrix} = H^{(2)} + H^{(3)} + H^{(4)} + H^{(8)} + H^{(9)}$$

$$\begin{bmatrix} 1\\0\\0\\0 \end{bmatrix} + \begin{bmatrix} 0\\1\\1\\1 \end{bmatrix} + \begin{bmatrix} 1\\1\\1\\0 \end{bmatrix} + \begin{bmatrix} 0\\0\\1\\0 \end{bmatrix} + \begin{bmatrix} 0\\0\\0\\1 \end{bmatrix}$$

$$= \begin{bmatrix} 0\\0\\0\\0 \end{bmatrix}$$ → El resultado al multiplicar por la matriz de chequeo es 0, quiere decir que la palabra recibida pertenece al código ✓

(6)

$$H = \begin{array}{c} \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \end{array} \\ \left[\begin{array}{cccc|cccc} 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \end{array}\right] \\ \underbrace{\qquad\qquad}_{Id} \quad \underbrace{\qquad\qquad}_{A^T} \end{array}$$

Y sea C el código asociado a ella

a) Describir C explícitamente (dar palabras que constituyen el código)

b) $\delta(c)$?

c) Sup. recibe la palabra 00111000. Asumiendo que se produjo a lo sumo error de transmisión ¿Qué palabra le fue enviada?

d) Recibe la palabra 11100111 ¿Qué puede concluir?

*a) $|C| = ?$ ¿ $k = \dim(C) = \#\text{columnas}(H) - \#\text{filas}(H)$

$$= 8 - 4$$
$$= 4$$

$\therefore |C| = 2^4 = 16$ palabras en el código

Cómo doy las palabras?

$$G = \left[\begin{array}{cccc|cccc} 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

A^T $\quad$ I

$[1110] \times G = 0110\ 1110$
$[1101] \times G = 1110\ 1001$
$[1111] \times G = 0000\ 1111$
$[0000] \times G = 0000\ 0000$
$[0001] \times G = 0110\ 0001$
$[0010] \times G = 1110\ 0010$
$[0011] \times G = 1000\ 0011$
$[0100] \times G = 0101\ 0100$
$[0101] \times G = 0011\ 0101$
$[0110] \times G = 1011\ 0110$
$[0111] \times G = 1101\ 0111$
$[1000] \times G = 1101\ 1000$
$[1001] \times G = 1011\ 1001$
$[1010] \times G = 0011\ 1010$
$[1011] \times G = 0100\ 1011$
$[1100] \times G = 1000\ 1100$

Palabras

b) $\delta(c) = ?$ ¿ $H^{(i)} \neq 0 \ \forall i$ $\quad H^{(i)} \neq H^{(j)} \ \forall i \neq j \Rightarrow \delta(c) \geq 3$.

$|H^{(i)}| \geq 3 \ \forall i \geq 5 \Rightarrow H^{(i)} \neq H^{(j)} + H^{(\tau)}$ con $j, \tau < 5$

$H^{(6)} = H^{(2)} + H^{(4)}$ $\qquad H^{(8)} = H^{(2)} + H^{(3)}$

$$\begin{bmatrix}0\\1\\0\\1\end{bmatrix} = \begin{bmatrix}0\\1\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\1\end{bmatrix} \qquad \begin{bmatrix}0\\1\\1\\0\end{bmatrix} = \begin{bmatrix}0\\1\\1\\0\end{bmatrix} + \begin{bmatrix}0\\0\\1\\0\end{bmatrix}$$

$\therefore \delta(c) \geq 3$ y es 3 pues $H^{(2)} + H^{(3)} + H^{(8)} = 0$

$$\begin{bmatrix}0\\1\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\1\\0\end{bmatrix} + \begin{bmatrix}0\\1\\1\\0\end{bmatrix} = \begin{bmatrix}0\\0\\0\\0\end{bmatrix}$$

c) $H \times \begin{bmatrix}0\\0\\1\\1\\1\\0\\0\\0\end{bmatrix} = H^{(3)} + H^{(4)} + H^{(5)}$

$$\begin{bmatrix}0\\0\\1\\0\end{bmatrix} + \begin{bmatrix}0\\0\\0\\1\end{bmatrix} + \begin{bmatrix}1\\1\\0\\1\end{bmatrix} = \begin{bmatrix}1\\1\\1\\0\end{bmatrix} \rightarrow$$ Este resultado es la columna H^7 de la matriz de chequeo. la palabra enviada es (*)

(*) $w = 00111000$
$e_7 = 00000010$ $\rightarrow 00111000 + 00000010 = \boxed{10011010}$

d) $H \times \begin{bmatrix}1\\1\\1\\0\\0\\1\\1\\1\end{bmatrix} = H^{(1)} + H^{(2)} + H^{(3)} + H^{(6)} + H^{(7)} + H^{(8)}$

$$\begin{bmatrix}1\\0\\0\\0\end{bmatrix} + \begin{bmatrix}0\\1\\0\\0\end{bmatrix} + \begin{bmatrix}0\\0\\1\\0\end{bmatrix} + \begin{bmatrix}0\\1\\0\\1\end{bmatrix} + \begin{bmatrix}1\\1\\1\\0\end{bmatrix} + \begin{bmatrix}0\\1\\1\\0\end{bmatrix}$$

$= \begin{bmatrix}0\\0\\1\\1\end{bmatrix} \rightarrow$ El resultado no corresponde a una columna de la matriz de chequeo, sabemos que el código corrige

$t = \left\lfloor \frac{\delta(c)-1}{2} \right\rfloor = \left\lfloor \frac{3-1}{2} \right\rfloor = 1$ error, como no se puede corregir quiere decir que ocurrieron 2 más de 1 error

Probablemente 2 xq el código detecta $\delta(c) - 1 = 3 - 1 = 2$ errores ☒

(7) ¿Es de código lineal C con M Generadora G tq $\delta(c)$ NO sea igual a la menor cant de unos que aparece en alguna fila de G.

$$G = \left[\begin{array}{ccc|cccc} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{array}\right]$$

$C = \{0000000, 0011111, 0101110, 0110011, 1101001, 1000111, 1011000, 1110110\}$ 8 puntos

$$H = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$|C| = k \dim(C)$
$= \#\text{columnas}(H) - \#\text{filas}(H)$
$= 7 - 4$
$= 3$

$\Rightarrow k = 2^k \rightarrow 2^3 = 8$ palabras.

Acá $\delta(c) = 3$ (no hay columna repetida ni columna 0.) y $H1 + H4 + H5 = \begin{bmatrix}0\\0\\0\end{bmatrix}$ y la mínima cant de unos en una fila de G es 4.

¿Es de código lineal?

$G = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$ $\quad C$ = espacio fila(G) $= \{000, 111, 011, 100\}$

$2^k = 2^2 = 4$
$k = 2$ filas.

$= \delta(c) = \min\{|v|, v$ es una fila de $G\}$
$= 1 \neq 2$ (filas)
↳ es la mínima distancia de hamming.

$\min\{dH(v,w), v, w \in C, v \neq w\}$.

$\#\{i : v_i \neq w_i\}$
(nº bit en donde difiere la palabra.

(VIII)

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}$$

y C el código asociado a ella.

a) Escribir 5 palabras que estén en C_d. Cuántas palabras tiene en total C?

b) Calcular $\delta(C)$

c) Sup. que Ud recibe la palabra 11100000000011. Asumiendo que se produjo a lo sumo un error de transmisión ¿Qué palabras le fue enviada?

(a) $|C| = ¿?$ $k = \dim(C) = \#\text{col}(H) - \#\text{filas}(H)$

$= 14 - 5$

$= 9.$

$\therefore |C| = 2^9 = 512$ palabras en el código

Para escribir 5 palabras tengo que ???

$H = [Id \mid A^t] \Rightarrow G = [A^t \mid Id]$

$$G = \left[\begin{array}{ccccc|ccccccccc} 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right]$$

A^t

Palabras

000000000 x G = 000000 | 000000000

000000001 x G = 10111 | 000000001

000000010 x G = 00110 | 000000010

000000011 x G = 10001 | 000000011

000000100 x G = 11010 | 000000100

000000101 x G = 01101 | 000000101

b) $\delta(C)$ = mínimo nº de columnas LD de H.

Como H no tiene columna 0 ni repetidas $\Rightarrow \delta(C) \geq 3$

y $\delta(C) = 3$ porque $H^6 = H^1 + H^2$

$$\begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

y si hago $H^1 + H^2 + H^6$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = 0$$

$\therefore \delta(C) = 3$

c) $$H \times \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} = H^1 + H^2 + H^3 + H^{13} + H^{14} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} \rightarrow$$ corresponde a columna 8 (H^8)

Ahora $W = 11100000000011 + e_8 (0000000 1000000)$

$= \boxed{11100000100001 1}$

palabra enviada.

XI) 65 preguntas c/preg → siempre, frecuentemente, Deve3 cuando, Rara ve3, Nunca

M cheque $(n-k)$ dimensión
generadora $k \times n$.

Se quiere codificar 32 preguntas, resp frecuentemente
código sea capaz corregir un error por dato y que codifique todos los datos posibles

a) Diseñe código lineal q satisfaga esto, dando matriz de chequeo e prop de menor tamaño posible.

b) Escriba 2 palabras que esten en su código

c) tome una de sus palabras de b) y cambiele los 2 primeros dígitos.
Sup que esa es la palabra que se recibe. Prediga que deducirá la persona que la recibe, de acuerdo con el código diseñado por Ud. Explique bien porqué?

(a) Sup que para c/1 de las 32 preguntas hay 5 respuestas posible

$2^{n-k} - 1 \geq n$ $\quad n = 12$, $k = 9$ $\quad 2^{12-9} - 1 \geq 12$

$2^3 - 1 \geq 12$

$\underbrace{65}_{7 \text{ bits}} \times \underbrace{5}_{3 \text{ bits}} = 325$ NO

$G = [\, I \mid A \,]$

$n = 13$ $k = 9$ $\quad 2^{13-9} - 1 \geq 13$

$2^4 - 1 \geq 13$

$15 \geq 13$ ✓

$\Rightarrow n = 13$

9×13

$13 - 9 = 4$

$\delta = 3$

10 bits. $= 1024$

9 bits $= 512$

$k = 9$ $\quad |C| = 2^k$

1001	1000
0011	0100
1010	0010
0100	0001
9	4

$n = p + k$

$k = 9$

$$H = \begin{array}{c} n-k \\ \\ \end{array} \left[\begin{array}{c|c} A^t & I \\ k & n-k \end{array} \right] \begin{array}{c} n-k \\ \\ \end{array}$$

n

$2^{n-k} - 1 \geq n$ $\quad 2$

$2^{n-K} - 1 \geq n$ n tiene q' ser mayor a $K=9$

$2^{13-9} - 1 \geq 13$ la matriz tiene que ser $n-K \times n$

$15 \geq 13$ ✓ $\Rightarrow 13-9 \times 13$

4×13

$$H = \left[\begin{array}{cccc|ccccccccc} 1&0&0&0&1&1&0&1&1&0&1&0&1 \\ 0&1&0&0&1&0&1&1&0&1&1&0&0 \\ 0&0&1&0&0&1&1&1&0&0&0&1&1 \\ 0&0&0&1&0&0&0&0&1&1&1&1&1 \end{array}\right]$$

$Id = n-K = 4$ $\quad 13-4=9$

$G = [A^T | Id]$

$$G = \left[\begin{array}{cccc|ccccccccc} 1&1&0&0&1&0&0&0&0&0&0&0&0 \\ 1&0&1&0&0&1&0&0&0&0&0&0&0 \\ 0&1&1&0&0&0&1&0&0&0&0&0&0 \\ 1&1&1&0&0&0&0&1&0&0&0&0&0 \\ 1&0&0&1&0&0&0&0&1&0&0&0&0 \\ 0&1&0&1&0&0&0&0&0&1&0&0&0 \\ 1&1&0&1&0&0&0&0&0&0&1&0&0 \\ 0&0&1&1&0&0&0&0&0&0&0&1&0 \\ 1&0&1&1&0&0&0&0&0&0&0&0&1 \end{array}\right]$$

b) $W1 = 1100100000000$

$W2 = 0110110000000$

c) tomo W1 y cambio 11 por 00

$W1' = 0000100000000$

va a chequear si pertenece al codigo

$$H W1'^T = H \times \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = H^{(5)} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Como el resultado corresponde a una columna entonces la palabra pertenece al codigo.

deduzca que la palabra fue

$\Rightarrow W1' + e^5 = 0000100000000 + 0000100000000$

$= 0000000000000$.

La palabra mas cercana a W1' es 0000000000000 por ende la persona que lo recibe va a creer que se envio esa palabra.

Codigos Hamming → todas columnas ≠ de cero. Código lineal q' tiene una matriz de chequeo con algún nº de filas (y 2^q, sus columnas son todas las columnas ≠ de cero con q filas

12) q filas con columna i-ésima igual a la representación binaria de i

H_i rep. Binaria de i

$H \cdot W^T = 0$

$\neq 0$

$W = 0000101001000011000000100000001$

$0 \overset{?}{=} H^5 + H^7 + H^{10} + H^{15} + H^{22} + H^{31}$. Tengo representar el 5, 7, 10, 15, 22, 31 en binario de longitud 5)

$$\Rightarrow \begin{bmatrix} 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{pmatrix} = H^{(14)}$$

∴ la palabra enviada más probable fue $W \times e_{14} = 0000101001000011000000100000001$

$+ 0000000000000001000000000000000$

$= \boxed{0000101001000111000000100000001}$

IX)

$$H = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} & x_{11} & x_{12} & x_{13} & x_{14} & x_{15} & x_{16} \end{matrix} \\ \left[\begin{array}{cccccc|cccccccccc} 1&0&0&0&0&0&1&1&1&1&1&0&0&1&0&1 \\ 0&1&0&0&0&0&1&1&1&1&1&0&1&1&0&0 \\ 0&0&1&0&0&0&1&1&0&0&0&1&1&0&1&1 \\ 0&0&0&1&0&0&1&1&1&1&1&1&0&1&1&1 \\ 0&0&0&0&1&0&0&1&1&1&0&1&1&0&a&b \\ 0&0&0&0&0&1&0&0&0&1&1&1&1&0&c&d \end{array}\right] \end{array}$$

Escribir dos palabras no nulas q' estén en C, decir cte palabra tiene en total C, $\delta(C)$?

se recibe 1000 000000 000 111. Se produjo a lo sumo un error de transmisión, det la palabra enviada si esto es posible o indicar porqué no se puede. esta depende de valores $a,b,c,d \in \{0,1\}$

$$G = \left[\begin{array}{cccccc|cccccccccc} 1&1&1&1&0&0&1&0&0&0&0&0&0&0&0&0 \\ 1&1&1&1&1&0&0&1&0&0&0&0&0&0&0&0 \\ 1&1&0&1&1&0&0&0&1&0&0&0&0&0&0&0 \\ 1&1&0&1&1&1&0&0&0&1&0&0&0&0&0&0 \\ 1&1&0&1&0&1&0&0&0&0&1&0&0&0&0&0 \\ 0&0&1&1&1&1&0&0&0&0&0&1&0&0&0&0 \\ 0&1&1&0&1&1&0&0&0&0&0&0&1&0&0&0 \\ 1&1&0&1&0&0&0&0&0&0&0&0&0&1&0&0 \\ 0&0&1&1&a&c&0&0&0&0&0&0&0&0&1&0 \\ 1&1&1&1&b&d&0&0&0&0&0&0&0&0&0&1 \end{array}\right]$$

$|C| = ?$ $K = \dim(C)$

$K = \#\text{columnas}(H) - \#\text{filas}(H)$

$K = 16 - 6$

$K = 10$

$\Rightarrow |C| = 2^{10} = 1024$ palabras en total

$W^1 = 1000000000 \times G = 111100|1000000000$
$W^2 = 0000000100 \times G = 110100|0000000100$
$W^3 = 0000001100 \times G = 101111|0000001100$
$W^4 = 0100000000 \times G = 111110|0100000000$

palabras en el código

Veamos ahora cuanto vale $\delta(C)$ primero podemos ver que 0 no es una columna de H (matriz de chequeo) $\Rightarrow$ por lo menos $\delta(C) > 1$.

CASO 1 luego si a y c son ambos 1 habría una columna repetida la cual es $\begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ Corresponde a $H^{(12)}$ y $H^{(15)}$ y como estamos en $\mathbb{Z}_2$ esas 2 columnas son LD, el mínimo nº de columnas LD de H es 2 y por teorema.
$\Rightarrow \boxed{\delta(C) = 2}$

CASO 2: Si $a \neq c$, entonces H no tiene columna repetidas y no tiene columna 0 $\therefore \delta(C) \geq 3$ y como además $H^{(5)} + H^{(7)} + H^{(8)} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$ $\therefore \boxed{\delta(C) = 3}$
$(H^{(8)} = H^{(7)} + H^{(5)}$ ($\exists$ conj LD con 3 columnas) $(\delta(C) \leq 3)$

* Se recibe la palabra $W = 1000000000000111$ y se asume que se produjo a lo sumo un error de transmisión. det la palabra.

$$H \cdot W^T = H \cdot \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix} = H^{[1]} + H^{[14]} + H^{[15]} + H^{[16]}$$

$$= \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ c \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 1 \\ 1 \\ b \\ d \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ a+b \\ c+d \end{bmatrix}$$

Ahora voy a ver (caso) según valores le de a a+b y c+d.
(continuo atrás)

Papel de fibra de caña de azúcar.

CASO 1 Si $a+b=0 \wedge c+d=0$

$11 \wedge 10 \Rightarrow H.w^t = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$ corresponde a columna 14 $(H^{(14)})$

$\Rightarrow w + e_{14} = 1\,000\,000\,000\,000\,111 + 0\,000\,000\,000\,000\,100$

$= \boxed{1\,000\,000\,000\,000\,011}$ palabra enviada

CASO 2 Si $a+b=0 \wedge c+d=1$

$\Rightarrow H.w^t = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{bmatrix}$ corresp a columna 11 $(H^{(11)})$

$\Rightarrow w + e_{11} = 1\,000\,000\,000\,000\,111 + 0\,000\,000\,000\,100\,000$

$= \boxed{1\,000\,000\,000\,100\,111}$ palabra enviada

CASO 3 Si $a+b=1 \quad c+d=1$.

$\Rightarrow H.w^t = \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 1 \end{bmatrix}$ corresponde a columna 10 $(H^{(10)})$

$\Rightarrow w + e_{10} = 1\,000\,000\,000\,000\,111 + 0\,000\,000\,001\,000\,000$

$= \boxed{1\,000\,000\,001\,000\,111}$ palabra enviada

CASO 4 Si $a+b=1 \quad c+d=0$

$\Rightarrow H.w^t \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix}$ corresponde a columna 9 $(H^{(9)})$

$\Rightarrow w + e_9 = 1\,000\,000\,000\,000\,111 + 0\,000\,000\,010\,000\,000$

$= \boxed{1\,000\,000\,010\,000\,111}$ palabra enviada

$2^{20} \to 1048576$

$t = \frac{\delta - 1}{2} = 1$

X) Codificar 1 millón palabras. Código corrige 1 error

a) ¿Menor tamaño que debe tener la matriz?
b) Escribir una matriz que satisfaga las cond del a)
c) Escribir 2 palabras de peso menor o igual a 6 q' esten en el código y una palabra de peso mayor o igual a 15 q' este en el código
d) Tomar 1 de esas palabras y cambiar los 2 primeros digitos. Sup q' esa es la palabra que se recibe. Prediga que deducirá la persona que la recibe, de acuerdo con el código diseñado por Ud. Explique bien xq!!

a) Los números de 0 a 999999

$n = 1$ millón, un código lineal tiene 2^k palabras para algún k, la potencia de 2 más cercana a 1 millón es $2^{20} = 1048576$ entonces $\boxed{k = 20}$

$\boxed{k = \dim(c)}$

para corregir un error tiene que $t = \frac{\delta - 1}{2} = 1$ despejando δ llegamos a que $\delta(c) \geq 3$ (ent no hay columnas repetidas ni columnas 0)

Sea n la long del código. Por teorema $\delta(c) \leq n - \dim(c) + 1$

$$3 \leq n - 20 + 1$$
$$3 + 20 \leq n + 1$$
$$23 - 1 \leq n$$
$$22 \leq n$$

Luego $n \leq 2^{n-k} - 1$
$n \leq 2^{n-20} - 1$

el n tiene que ser mayor a 20 $\Rightarrow$ tomo $n = 21$

$21 \leq 2^{21-20} - 1$
$21 \leq 1$ ✗

$n = 22$ $\quad 22 \leq 2^{22-20} - 1$
$22 \leq 4 - 1$
$22 \leq 3$ ✗

$n = 23$ $\quad 23 \leq 2^{23-20} - 1$
$23 \leq 8 - 1$
$23 \leq 7$

$n = 24$ $\quad 24 \leq 2^{24-20} - 1$
$24 \leq 16 - 1$
$24 \leq 15$ ✗

$n = 25$ $\quad 25 \leq 2^{25-20} - 1$
$25 \leq 32 - 1$
$\boxed{25 \leq 31}$ ✓ $\Rightarrow n = 25$ y el mínimo n posible.

$2^{n-k} - 1 \geq n$

$\Rightarrow$ a) Mínimo tamaño de la matriz es por $n = r + k$ es 5×25
$25 = 5 + 20 \to 25 = 25$

$$H = \begin{array}{c} \\ n-k \end{array} \begin{array}{c} \overbrace{\qquad\qquad}^{n} \\ \left[\begin{array}{c|c} Id & A^T \end{array}\right] \\ \underbrace{\qquad}_{n-k} \quad \underbrace{\qquad}_{k} \end{array} n-k$$

b)
$$H = \left[\begin{array}{ccccc|cccccccccccccccccccc} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}\right]$$

$\underbrace{\qquad}_{n-k}$
$25 - 20$
$\boxed{5}$

c) Tengo que armar la matriz generadora $G = [A^T \mid Id]$

20×20

$$G = \left[\begin{array}{ccccc|ccc}
1&1&0&0&0&1&0&0\\
1&0&1&0&0&0&1&0\\
0&1&1&0&0&0&0&1\\
1&1&1&0&0&0&0&0\\
1&0&0&1&0&0&0&0\\
0&1&0&1&0&0&0&0\\
1&1&0&1&0&0&0&0\\
0&0&1&1&0&0&0&0\\
1&0&1&1&0&0&0&0\\
0&1&1&1&0&0&0&0\\
1&1&1&1&0&0&0&0\\
1&0&0&0&1&0&0&0\\
0&1&0&0&1&0&0&0\\
1&1&0&0&1&0&0&0\\
0&0&1&0&1&0&0&0\\
1&0&1&0&1&0&0&0\\
0&1&1&0&1&0&0&0\\
1&1&1&0&1&0&0&0\\
0&0&0&1&1&0&0&0\\
1&0&0&1&1&0&0&0\\
0&1&1&1&1&0&0&0
\end{array}\right]$$

$w_1 = 11000\,0|100000000000000000000$ Peso de esta palabra es 3 y 3 cumple con ≤ 6 ✓

(sumando todo)

$w_2 = 01111|111111111111111111111$ peso de esta palabra es 24 y cumple ≥ 15

d) Tomo w_1 y cambio 11 por 00 del principio

$w_3 = 00000|100000000000000000000$

la persona que recibe la palabra va a calcular $H\,w_3^T$

$$H * \begin{bmatrix}0\\0\\0\\0\\1\\0\\0\\0

$\Rightarrow |A| = |B|$

$$\boxed{\begin{array}{l} \bullet\ A = \{A' \subseteq \{0,1\}^{12} : \#A' = 512\} \\ \bullet\ B = \{B' \subseteq \{0,1\}^{12} : \#B' = 3584\} \\ \bullet\ L = \{L' \subseteq \{0,1\}^{12} : L \text{ es lineal}\} \end{array}}$$

Ahora tengo que ver si se cumple $|A \cap L| > |B \cap L|$

Sabemos que todo código lineal cumple $|C| = 2^k$ con $k \in \mathbb{N}$ y no $\exists$ un k tq $2^k = 3584$

$\Rightarrow B \cap L = \emptyset$ y $|B \cap L| = 0$

Pero $2^k = \overbrace{512}^{A'}$ si $\exists$ un k y es $k=9$ por lo que $A \cap L =$ "todos los códigos posibles finales de 512 elementos del conjunto de 4096 con longitud 12 $\neq \emptyset$

$\therefore |A \cap L| > |B \cap L|$

$\underbrace{|A \cap L|}_{\neq \emptyset} > 0$

$$\boxed{\therefore \exists C \subseteq \{0,1\}^{12} : \#C = 512 \wedge C \text{ es lineal}}$$

$\therefore$ Se cumple que $|A| = |B|$ pero que $|A \cap L| > |B \cap L|$

(14) Probar que si C es un código binario que corrige un error pero no corrige 2, ent. C es perfecto $\overset{\Leftrightarrow}{\text{sii}}$ $\exists$ r tq C tiene long $\underbrace{2^r - 1}_{\text{ndig de } \mathcal{H}_r}$ y $\underbrace{2^{2^r - r - 1}}_{2^{\dim(\mathcal{H}_r)} = |\mathcal{H}_r|}$ elementos.

Sabemos que el código es perfecto si:

$$\underbrace{\#C}_{2^{\underbrace{k}_{\dim(C)}}} = \frac{2^n}{\sum_{r=0}^{t} \binom{n}{r}} \quad (|A| \subseteq \{0,1\}^n \Rightarrow |A| \leq 2^n$$

nº filas.

lo que se por enunciado C corrige 1 error y no 2 errores, C código, long n.

C es perfecto $\Leftrightarrow \exists r \in \mathbb{N} : n = 2^r - 1 \wedge \#C = 2^{2^r - r - 1}$

C es perfecto si $\#C = \dfrac{2^n}{\sum_{i=0}^{1} \binom{n}{i}}$ $\quad \left|\dfrac{\delta(C) - 1}{2}\right| = t$ y $t = 1 \Rightarrow$ por eso es 1

$\Rightarrow \#C = \dfrac{2^n}{\binom{n}{0} + \binom{n}{1}} \Rightarrow \boxed{\#C = \dfrac{2^n}{1+n}}$

Ahora debo probar el $\Leftrightarrow$ 1º voy a demostrar el $\Leftarrow$) Suponiendo el antecedente

Sea $r \in \mathbb{N}$ tq $n = 2^r - 1 \wedge \#C = 2^{2^r - r - 1}$.

C es perfecto. $\#C = \dfrac{2^n}{1+n}$ ahora reemplazo n por $n = 2^r - 1$

$\Rightarrow \#C = \dfrac{2^{2^r - 1}}{\cancel{1} + 2^r - \cancel{1}}$

$\Rightarrow \#C = \dfrac{2^{2^r - 1}}{2^r}$ exponente como (-)

$\Rightarrow \#C = 2^{2^r - 1 - r}$ se cumple por antecedente fin $\Leftarrow$) ∎

Ahora la $\Rightarrow$) Sup antecedente.

Sabemos que $\#C = \frac{2^n}{1+n}$ por demo anterior

$1+n$ divide a 2^n

$\hookrightarrow (1+n) \mid 2^n$

$\Rightarrow$ y $1+n$ es potencia de 2

$\exists r \in \mathbb{N} : n = 2^r - 1$

$r = 1 \cdot$ y $n = 2^r - 1 = 1 \Rightarrow |C| = \frac{2}{2} = 1$

$\Rightarrow$ tienen la misma cardinalidad

$n = 2^1 - 1 = 1$

$\#C = \frac{2^1}{1+1} = 1$ $\quad \#C = \frac{2^{2^r-1}}{1+2^r-1} = \frac{2^{2^1-1}}{1+2^1-1} = \frac{2^1}{2} = 1$

por demo anterior $\#C = \frac{2^n}{1+n}$ $\quad \exists r \in \mathbb{N} . n = 2^r - 1 \wedge \#C = \frac{2^{2^r-1}}{1+2^r-1}$

$\Rightarrow \frac{2^{2^r-1}}{1+2^r-1} = \frac{2^{2^r-1}}{2^r} = 2^{2^r-1-r} = 2^{2^r-r-1}$.

$\therefore \exists r \in \mathbb{N} : n = 2^r - 1 \wedge \#C = 2^{2^r-r-1}$ ■

(15) N. idea

(XVI) caballeros $\Rightarrow$ verdad. picaros $\Rightarrow$ mienten. Hay 2048 llaves (K). Una de esas abre la puerta de una habitación que tiene oro y resolución problema P=NP. 2047 llaves restantes destruyen la habitación si la usa e infecta la compu.

Ayudar: 15 hab. están dispuestos a responderles preguntas "SI o NO" sólo 1 vez. C/u sabe cual es la llave pero no se sabe quien es caballero y quien picaro.

Persona 16 dice: "Si soy un caballero, ent. entre esas 15 personas hay a lo sumo un sólo picaro"

Deducción: evalúo lo que dice el 16° habitante.

p	q	$p \Rightarrow q$
V	V	V
V	F	F
F	V	V
F	F	V

CASO 1.

P = "Si soy caballero" $\Rightarrow$ q = "entre esa 15 pers. hay a lo sumo un sólo picaro".

CASO 1: 16° hab es caballero $\Rightarrow v(p) = true$. y por ende $v(q) = true$.

$\vee \neg p \Rightarrow \neg q$ = True

CASO 2: 16° hab es pícaro $v(p) = false$ por ende $v(\overset{p}{false} \Rightarrow q) = true$ siempre. es decir siempre hay a lo sumo 1 pícaro en la isla. (si q es true o false)

$\Rightarrow$ El 16° habitante dice la verdad.

Vamos a codificar las 2048 llaves. Para ello necesito un código C tal que $|C| = 2048 = 2^{11}$. Por la Cota de hamming $n \geq 15 \Rightarrow G$ va a ser de la forma $\{0,1\}^{11 \times 15}$ con la identidad adelante. También pedimos que G nos garantice que la matriz de chequeo (H) no tenga columnas iguales para que $\delta(C) = 3$ y $\therefore \tau = \left\lfloor \frac{\delta(C)-1}{2} \right\rfloor = 1$. Sabemos que esa H existe porque $\mathcal{H}_r$ tiene esa característica (con $r = 4$) es decir $\mathcal{H}_4$ (por teórico) pero con

$\left(\begin{array}{ll} n = 2^r - 1 & \wedge\ K = 2^r - r - 1 \\ \downarrow = 2^4 - 1 & 11 = 2^4 - 4 - 1 \\ 15 = 15\ \checkmark & 11 = 11\ \checkmark \end{array} \right)$

Una permutacion en las columnas para dejar la identidad adelante.
Luego lo que hacemos es darle la matriz generadora G a los 15 habitantes, le pedimos que codifiquen la llave correcta y le preguntamos al habitante i-esimo cual es el bit i-esimo en nuestro código que representa la llave que abre la puerta de la habitación con oro
Con $i = 1, \dots, 15$
De esta manera gasto las 15 preguntas para construirme la palabra w que codifica que llave es la correcta para abrir dicha puerta.
Una vez obtenida w lo que hago es obtener primero la matriz de cheque luego multiplico por la w^T (esto es $H \cdot w^T$) y puede suceder 2 casos:

a) $H \cdot w^T = 0$ (columna) $\Rightarrow$ la llave correcta es la interpretación en decimal de los primeros 11 bits de w, ya que lo que sigue es la identidad

b) $H \cdot w^T = H^{(i)}$ con $i = 1, \dots, 15$ (sabemos que si no es 0, siempre será $H^{(i)}$ porque hay a lo sumo 1 pícaro (1 error). En este caso la llave que abre la puerta será la interpretación en decimal de los primeros 11 bits de w' con $w' = w + e_i$ (corregido).

(15) Ni idea no muy importante hacer solo si me da curiosidad

Versión occidental.

```
g = 0
flag = 1
while (flag)
  P := {s} ; X = s.
  while (X ≠ T AND FLAG)
    if (Γ+(x) ≠ ∅)
      Tomo y ∈ Γ+(x).
      Agrego y a P.
      Hacer x = y.        } avanzar (A)
    else (Γ+(x) = ∅)
      if (x ≠ s)
        tomo z como dem anterior a x en P.
        borro x de P.
        borro el lado zx del NA.
        Hacer x = z.      } retroceder. (R)
      else
        flag = 0.
      F.
    F.
  od
  if (X == T)
    Aumenta flujo a lo largo de P.
    borrar lados saturados.   } incrementar (I)
  F.
od
return (g).
```

Compl (A ~ AX) = O(n) y hay O(m) palabras

∴ Compl. total es O(nm)

Una corrida dinámica occidental es una palabra

AA ~ I A A R ~ A I ~~~~

Miremos subpalabras de la forma A ~ AX. en donde.

X = I ó R.

- C/ avanzar es O(1)
- C/ retroceder es O(1)

C/ I es recorrer un camino de long n dos veces ∴ O(n).

¿Cuantos A's hay en A ~ AX? C/ A mueve el pivote X a un nivel superior ∴ Hay O(n) A's en C/ A ~ AX, no puede superar la cant. de niveles

∴ comp. (A ~ AR) = $\overbrace{O(n)}^{A's}$ + $\overbrace{O(1)}^{por\ R}$ = O(n)

∴ compl (A ~ AI) = $\overbrace{O(n)}^{A's}$ + $\overbrace{O(n)}^{por\ I}$ = O(n)

¿Cuantas hay? Hay O(m) palabras.

- C/ R borra un lado.
- C/ R borra al menos 1 lado (aumenta flujo y borra el o los lados saturados)