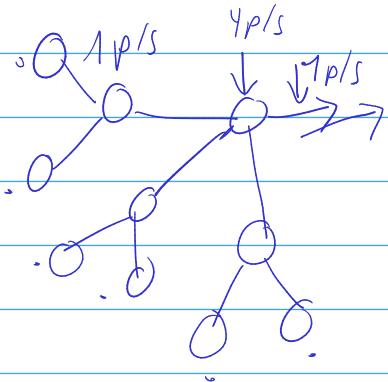
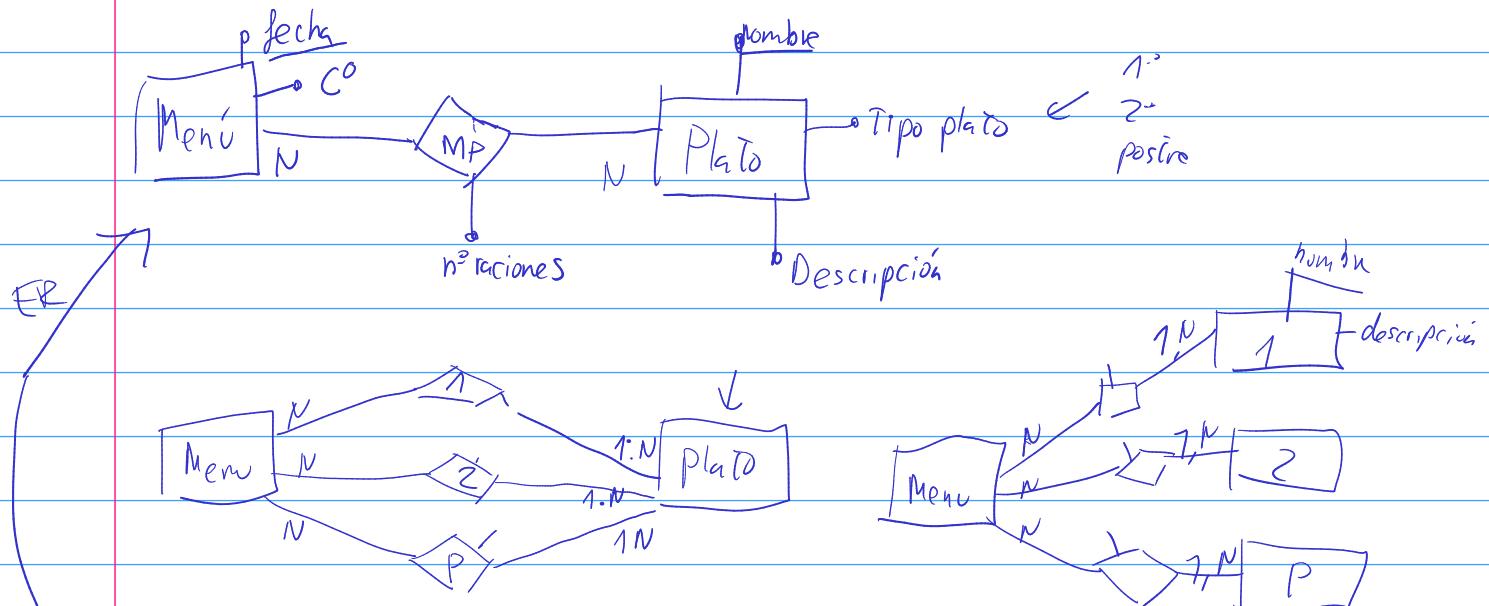
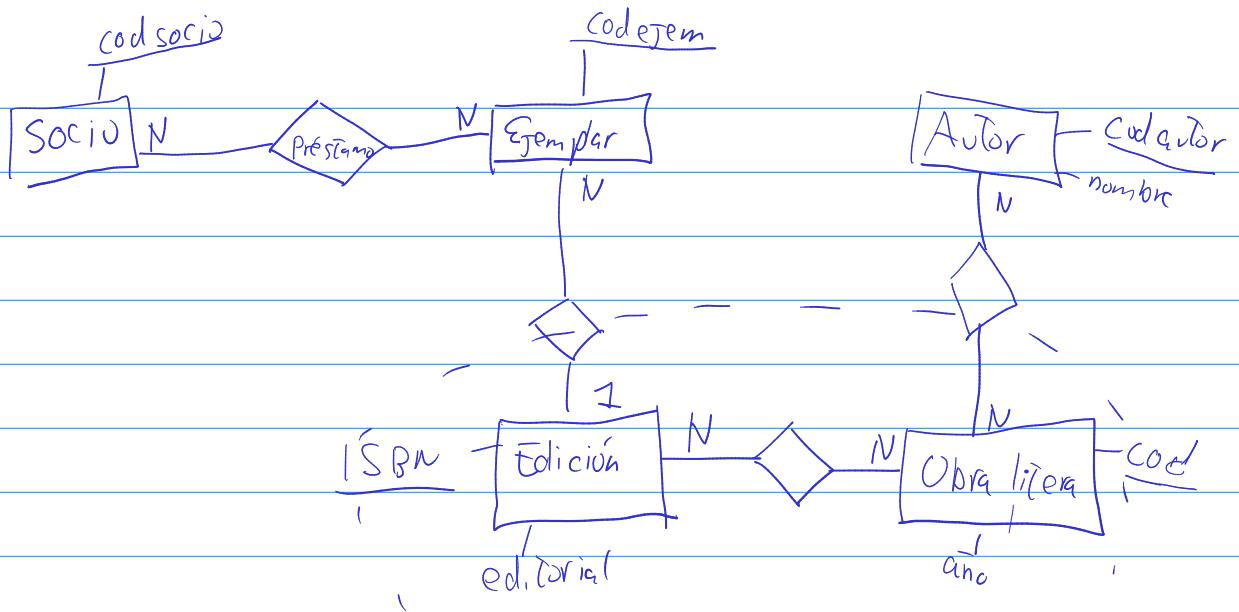


Topología	Coste	Tolerancia a fallos	Privacidad	Congestión	Otros criterios
Bus	1º poco	5º grande	10º	10º	
Estrella	3º	4º	2º	3º o 4º	
Árbol	4º	3º	3º	3º	
Anillo	2º	6º	4º		
Anillo doble	3º	2º	4º	1º o 2º 1º o 2º	
Malla completa	10º	1º	1º	1º	

Método cable
nº adaptadores
interconexión

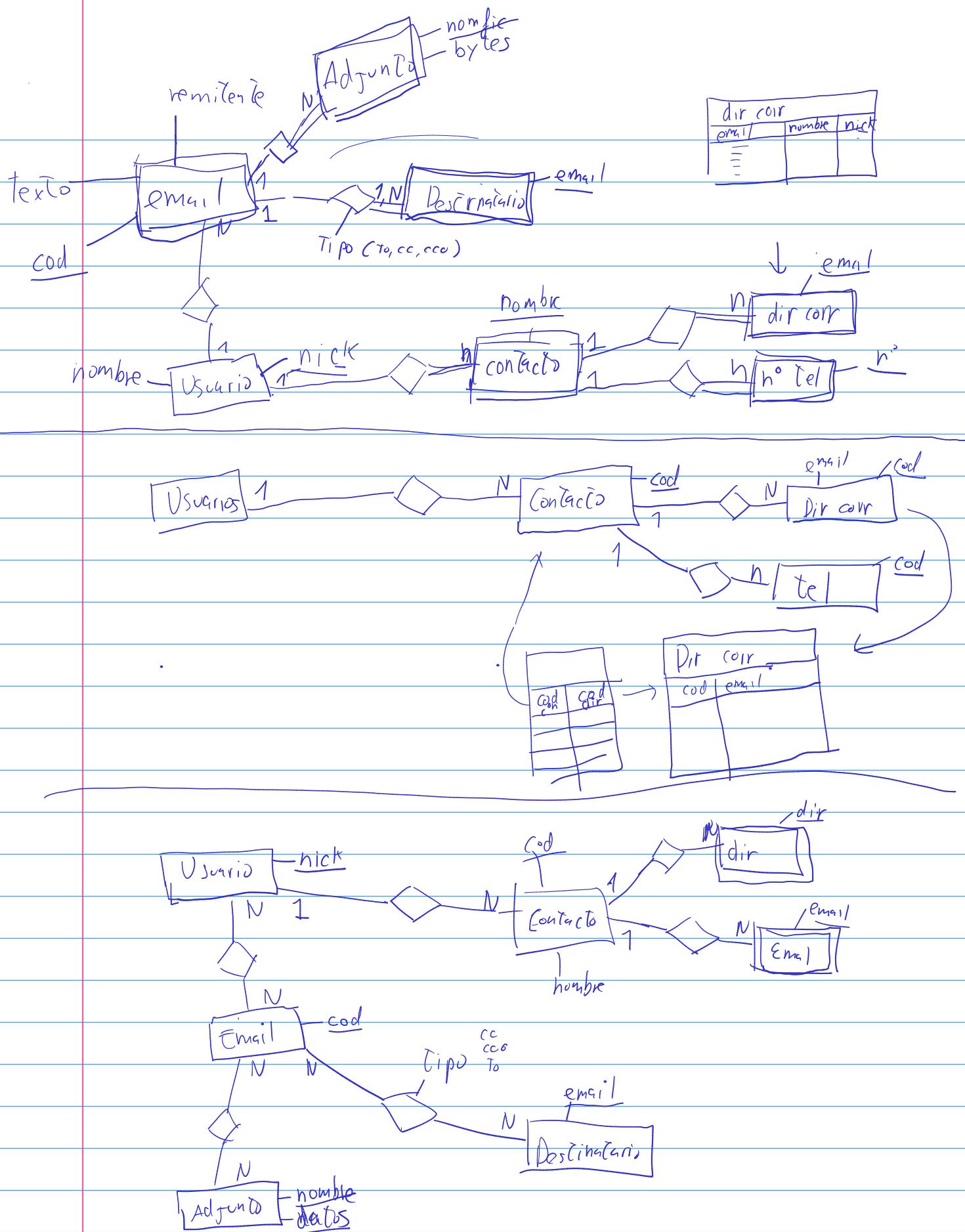


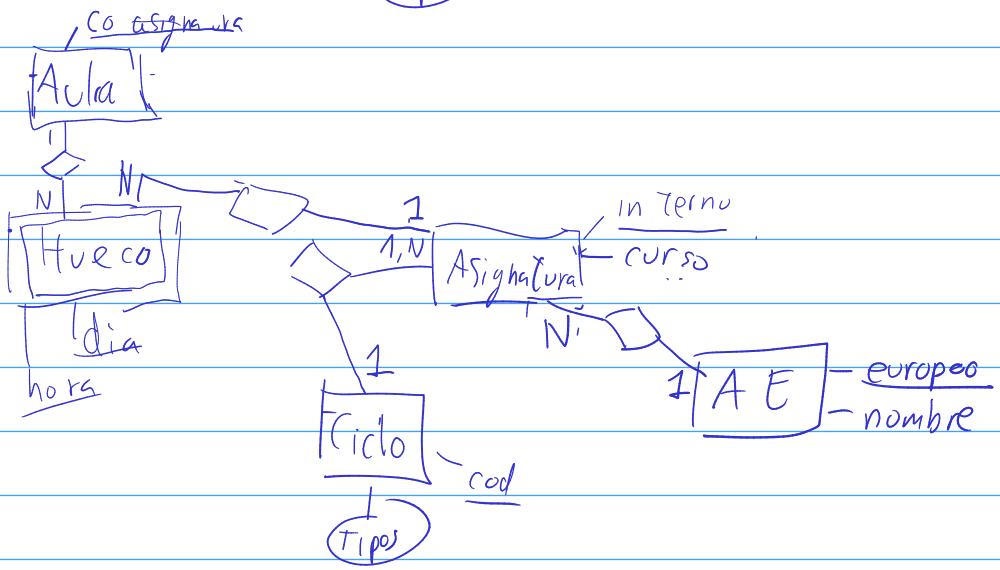
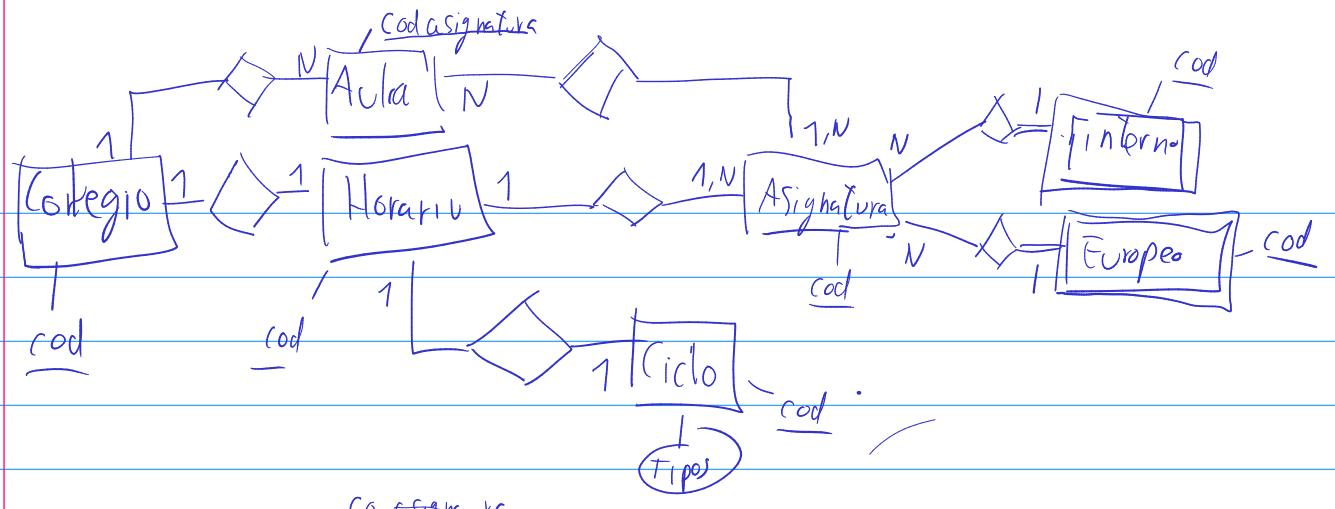


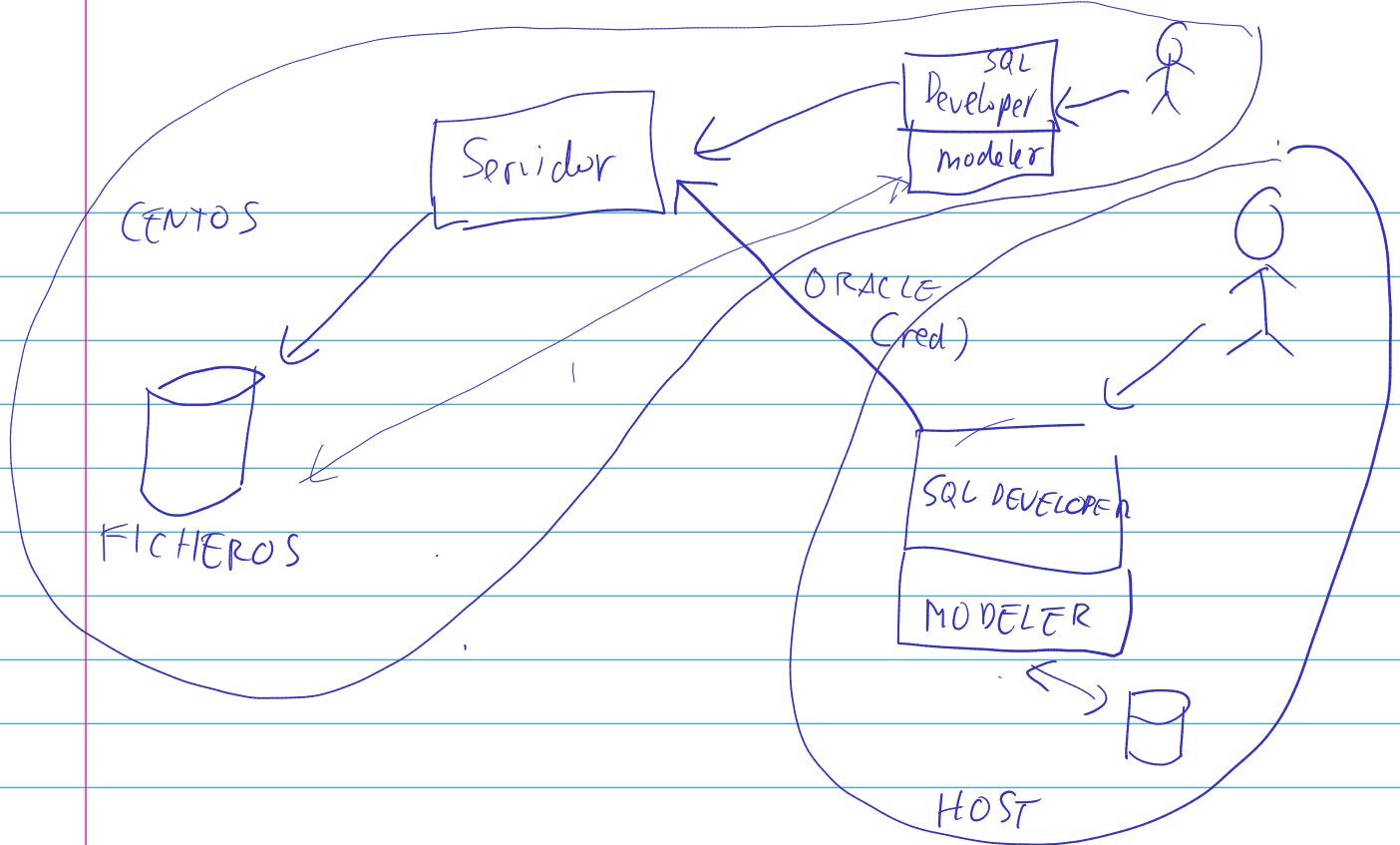
menu	fecha	temp
/	/	/
/	/	/
/	/	/
/	/	/

MP		
n?	fecha	nombre
-	-	-
-	-	-
-	-	-

Platō		
<u>Nombre</u>	Type	Descripción







ENTIDAD

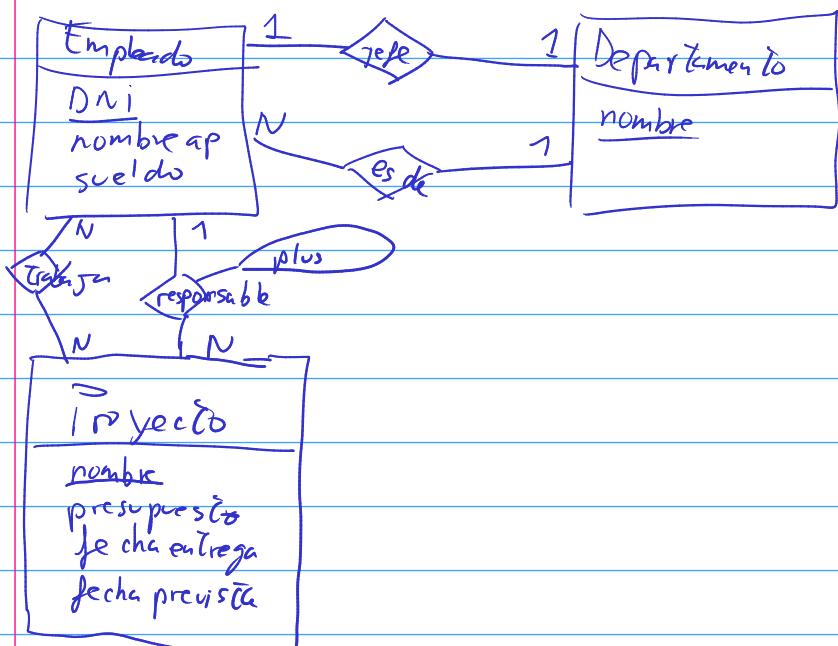
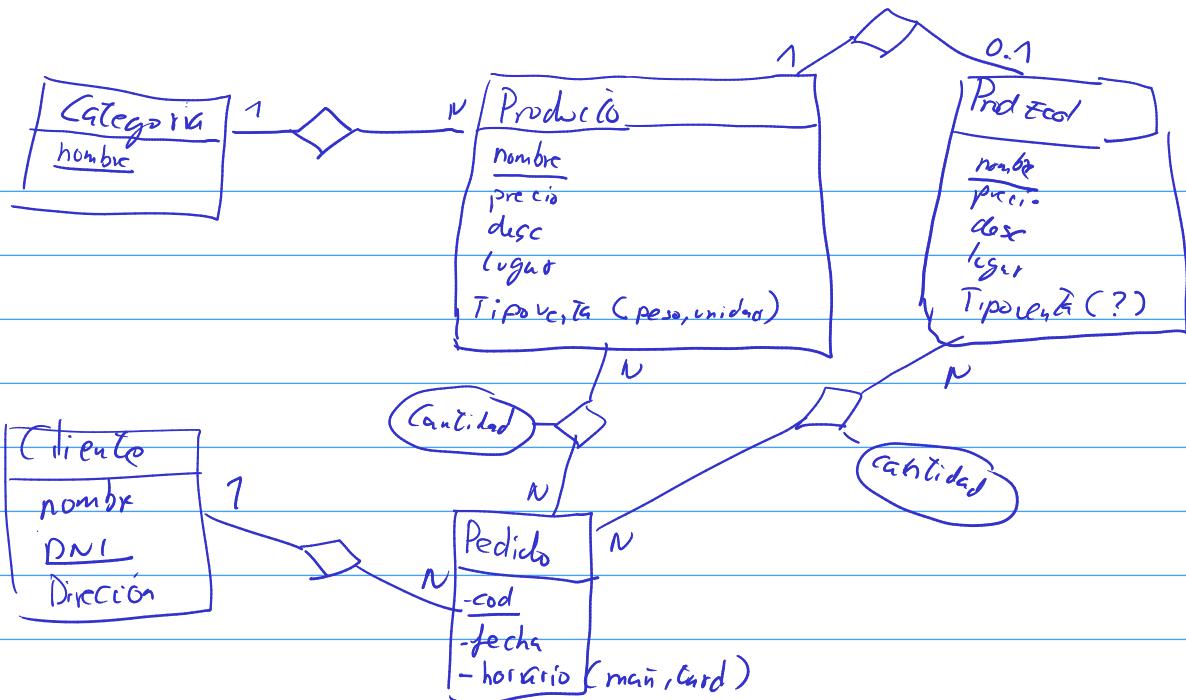
- Hay más de una instancia?
- Tiene atributos?
- Todas las instancias tienen los mismos?
- Tiene identificador?

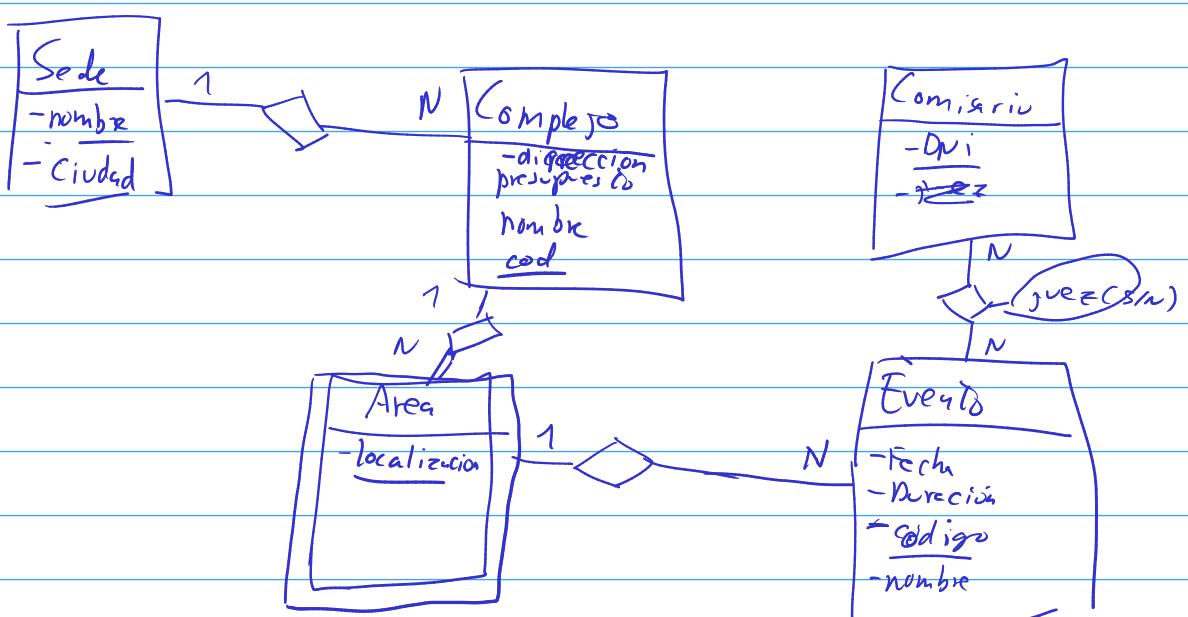
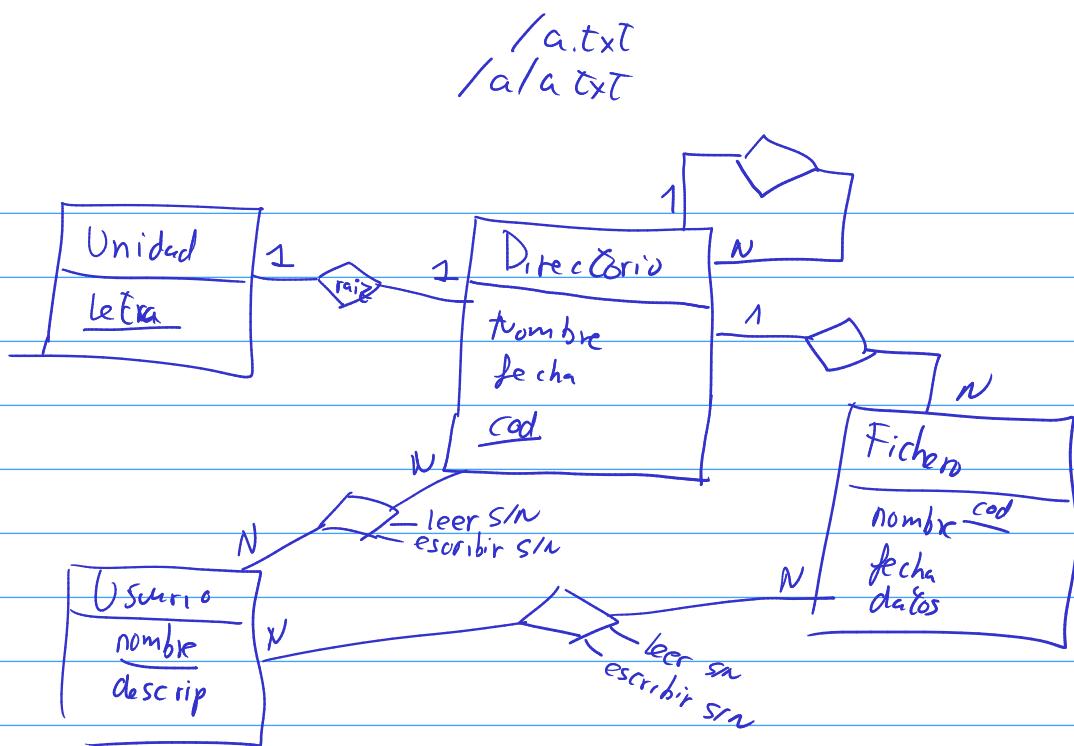
ATRIBUTO

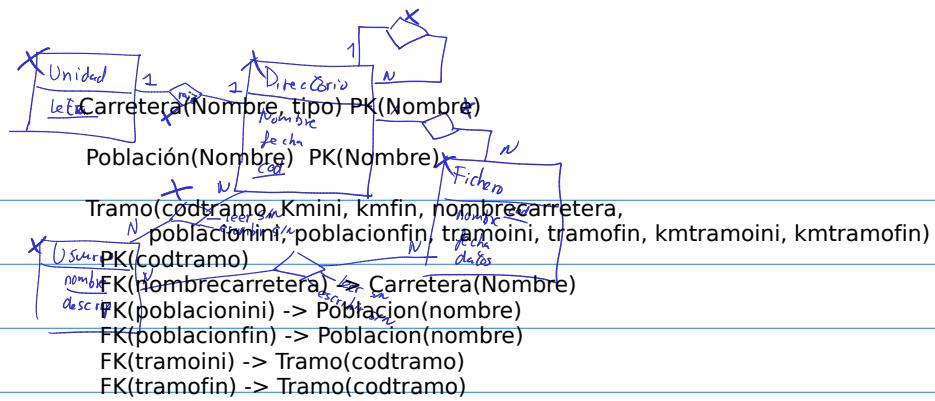
- Multivalorado?
- Ejemplos
- Catalogos?

INTERACCIONES

- Cardinatidad $1-1$
 $1-n$
 $n-1$
 $n-n$







Unidad (Letra, Carretera, Directorio, Poblacion, Tramo, Fichero)

Directorio (cod, nombre, fecha, datos, codpadre)

Fichero (cod, nombre, fecha, datos, codpadre)

Usuario (nombre, descripcion, permisos)

Permisos Directorio (nombreusu, caddir, leer, escribir)

Permisos Ficheros (nombreusu, codfic, leer, escribir)

Unidad(letra, codraiz)
pk(letra)
fk(codraiz) -> Directorio(cod)

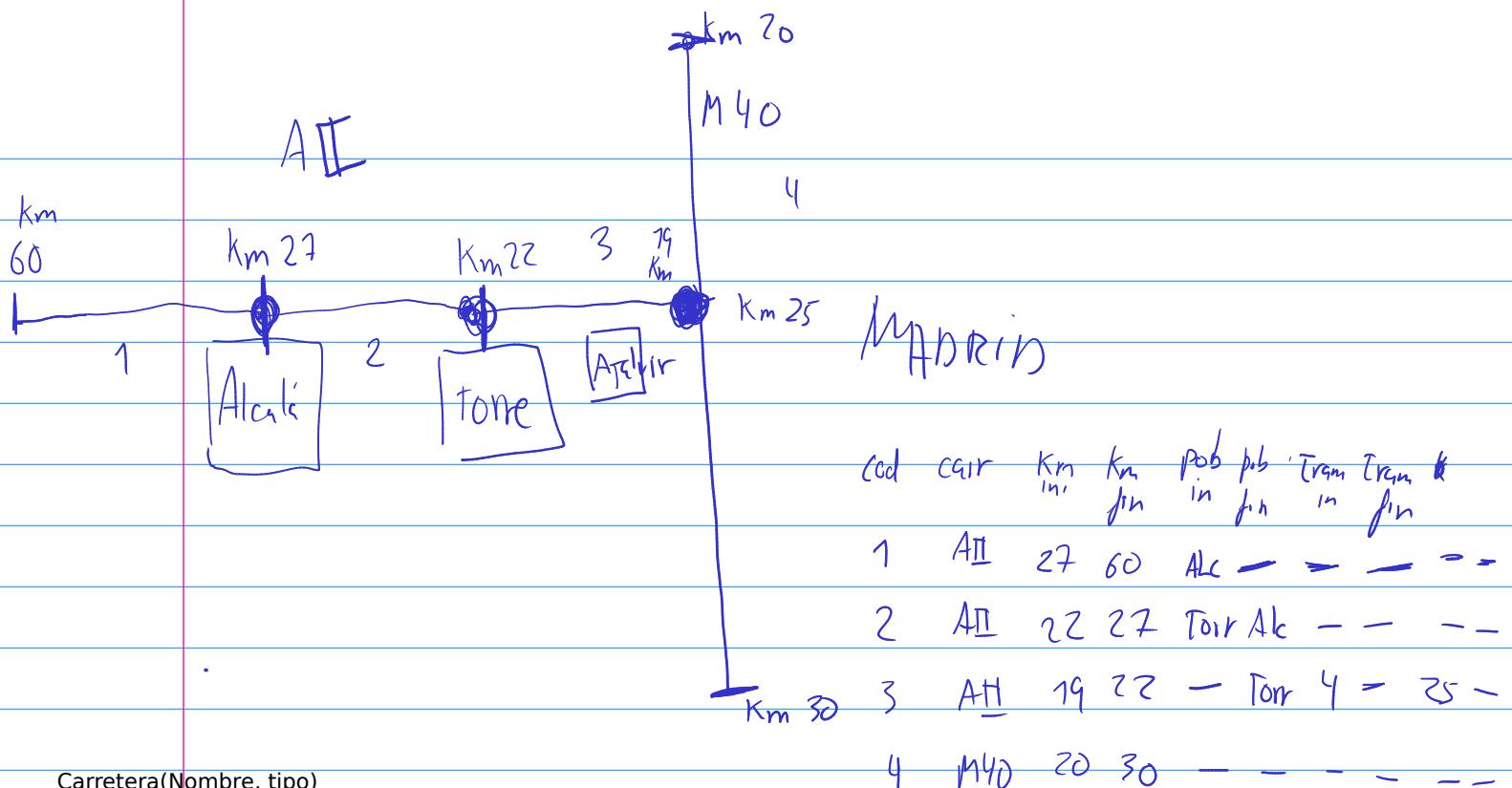
Directorio(cod,nombre,fecha,codpadre)
pk(cod)
fk(codpadre) -> Directorio(cod)

Fichero(cod, nombre, fecha, datos, codpadre)
pk(cod)
fk(codpadre) -> Directorio(cod)

Usuario(nombre, descripcion)
pk(nombre)

PermisosD(nombreusu, caddir, leer, escribir)
pk(nombreusu,codir)
fk(nombreusu)->Usuario(nombre)
fk(caddir)->Directorio(cod)

PermisosF(nombreusu, codfic, leer, escribir)
pk(nombreusu,codfic)
fk(nombreusu)->Usuario(nombre)
fk(codfic)->Fichero(cod)



Carretera(Nombre, tipo)
PK(Nombre)

Población(Nombre)
PK(Nombre)

Tramo(codtramo, Nombrecarretera, Kmmini, kmfin, poblacionini, poblacionfin, tramoini, tramofin, kmtramoini, kmtramofin)
PK(codtramo)

FK(nombrecarretera) -> Carretera(Nombre)

FK(poblacionini) -> Poblacion(nombre)

FK(poblacionfin) -> Poblacion(nombre)

FK(tramoini) -> Tramo(codtramo)

FK(tramofin) -> Tramo(codtramo)

TramoPasaPorPoblacion(codtramo, nombre)

PK(codtramo, nombre)

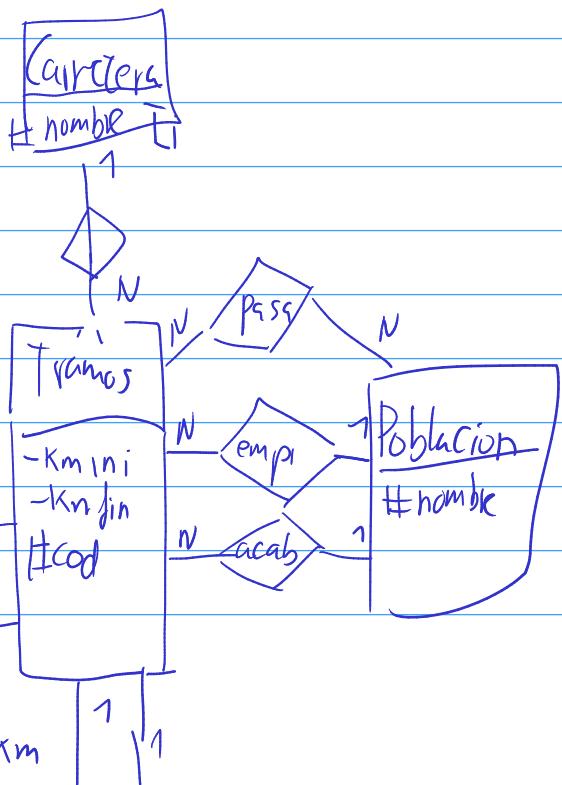
FK(codtramo) -> Tramo

FK(nombre) -> Poblacion

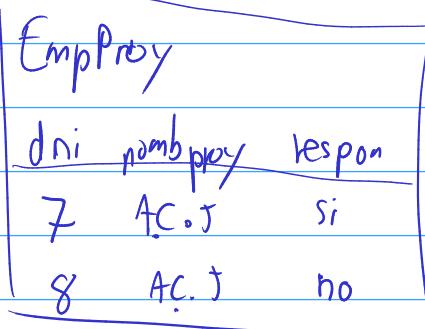
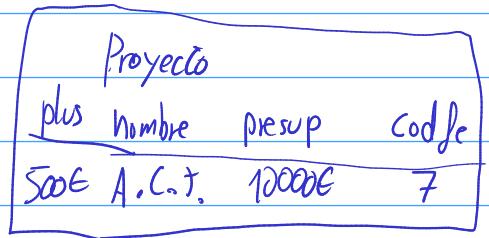
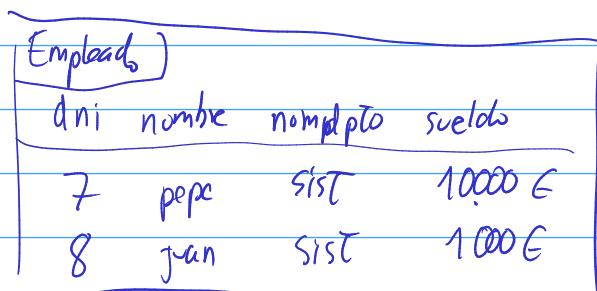
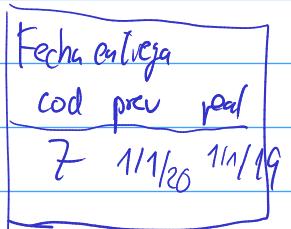
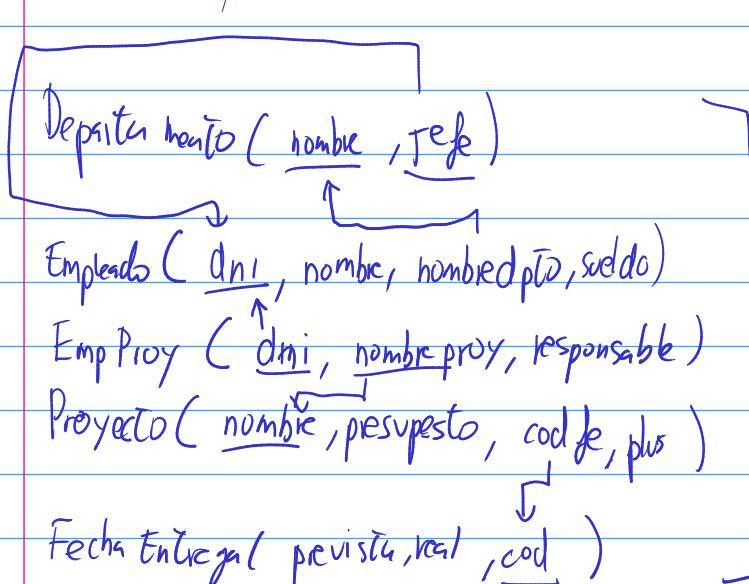
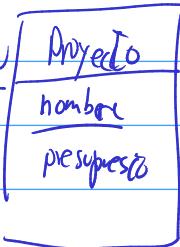
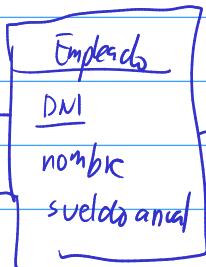
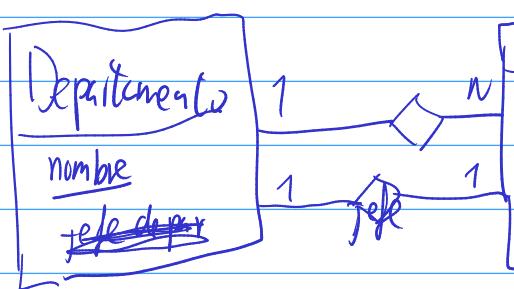
3 Ajalut

4 Madrid

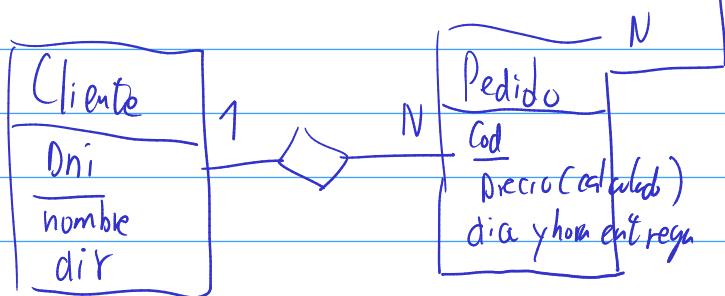
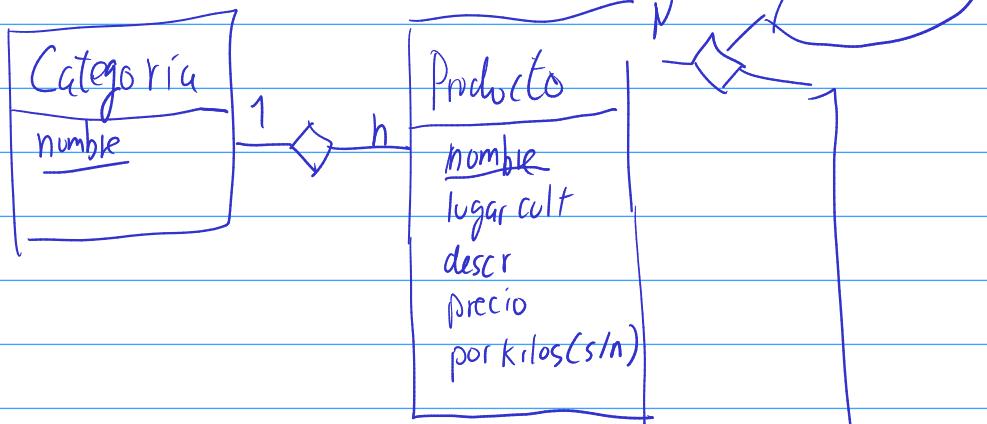
3 Madrid



EJEMPLO MAC



Otro Ejemplo: MAL



Definiciones de clases:

- Cliente (DNI, nombre, dir)
- Pedido (Cod, dia hora entrega, 1DNI)
- (estu(cant, cantidad, cod prod, prod / version))
- Producto (nombre, lugar, desc, precio, por kilos, categoria)
- Categoría (nombre)

Producto		
nombre	precio	eco
manzanas	2€	s
peras	3€	
manzanas	1€	n

Productor Pedido				
cod	cantidad	prod	version	pedido
3 Kg	manzana	eco		1
4 Kg	peras	no eco		1

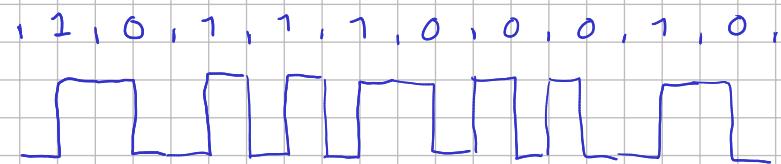
Pedido			
cod	fecha	DNI	
1	manzana peras	Pepe	

Codifica los dígitos 1011100010 en:

Manchester

0 ↑
1 ↓

Manchester diff
0 cambia al inicio
1 cambian en medio



NRZL

0 -5V
1 +5V

NRZI

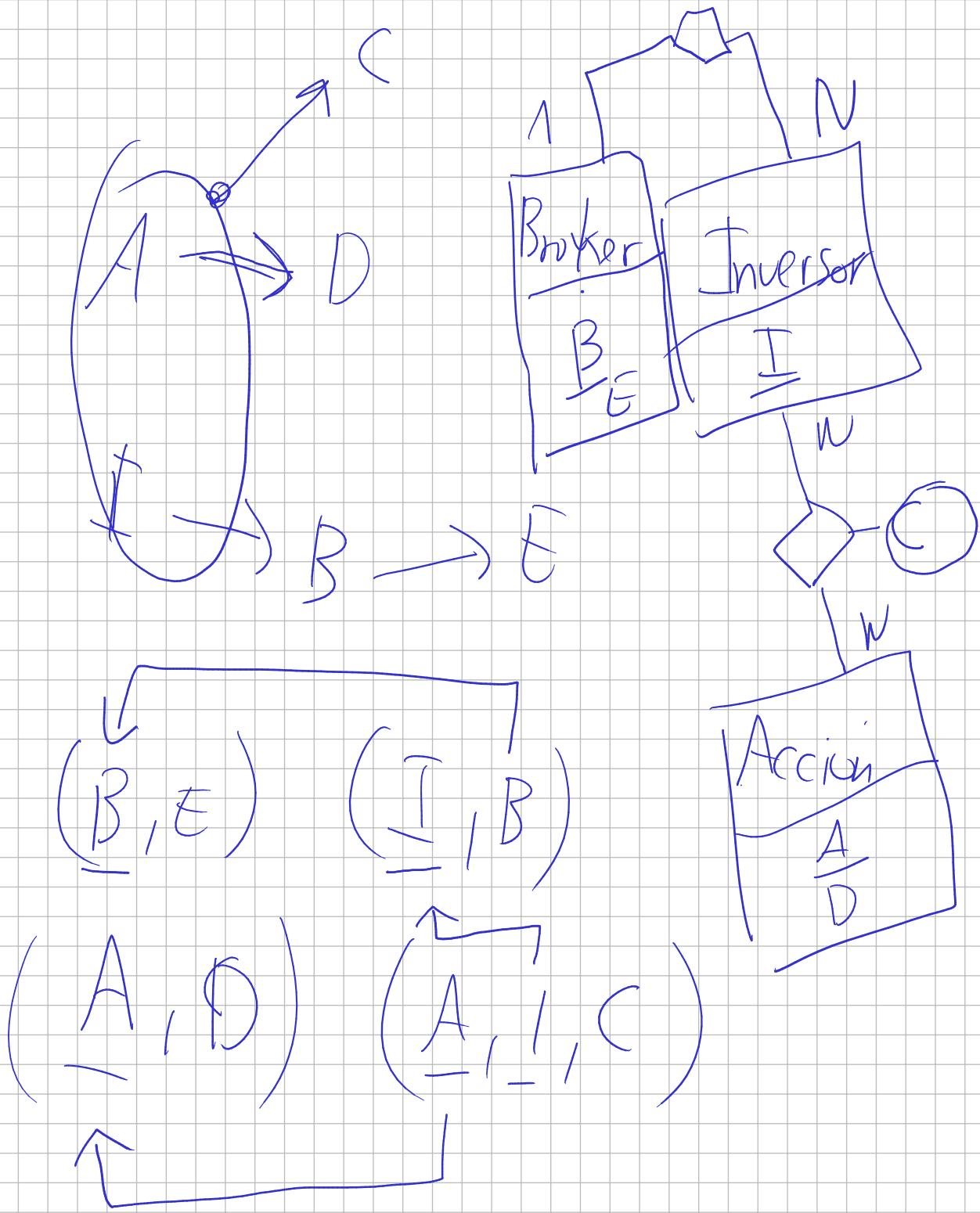
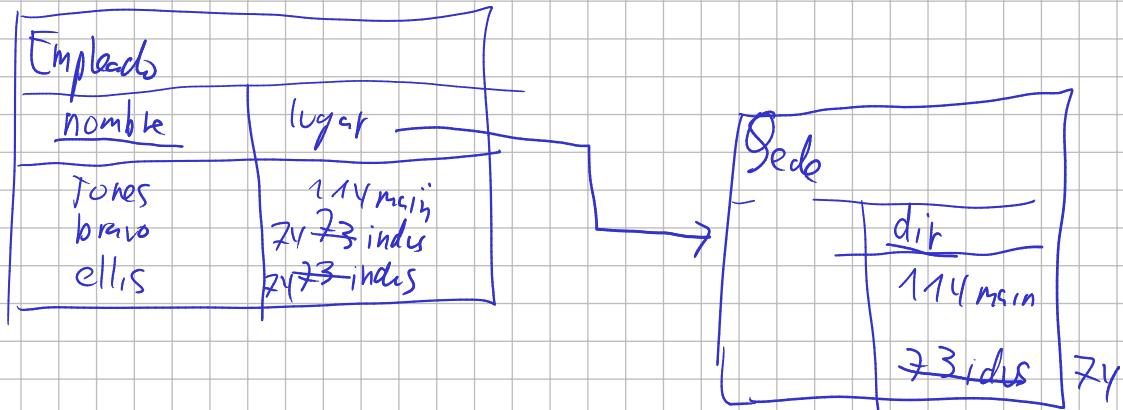
1 cambia

-5V +5V

AMI

1 alterna
0 0V





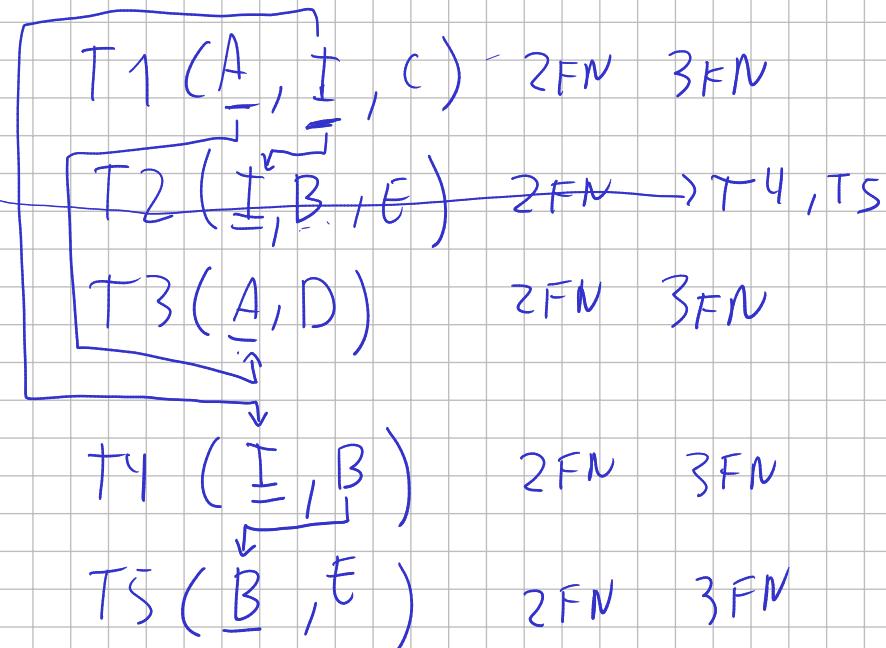
$A \rightarrow D$

$I \rightarrow B$

$I, A \rightarrow C$

$B \rightarrow E$

~~$T(A, B, C, D, E, I)$~~ 1FN $\rightarrow T_1, T_2, T_3$



$T_1(\underline{A}, \underline{I}, C)$

$T_3(\underline{A}, D)$

$T_4(\underline{I}, B)$

$T_5(\underline{B}, E)$

Acciones de Inversor

Acciones

Inversor

Broker

$A \rightarrow J$

$J, I \rightarrow H$

$B, E \rightarrow D$

$B, A \rightarrow C$

Exp	Nombre alumno	asignatura	notJulio
1	maria mer	MAT	7
1	maria mer	engl	8

$Exp \rightarrow \text{nombre}$

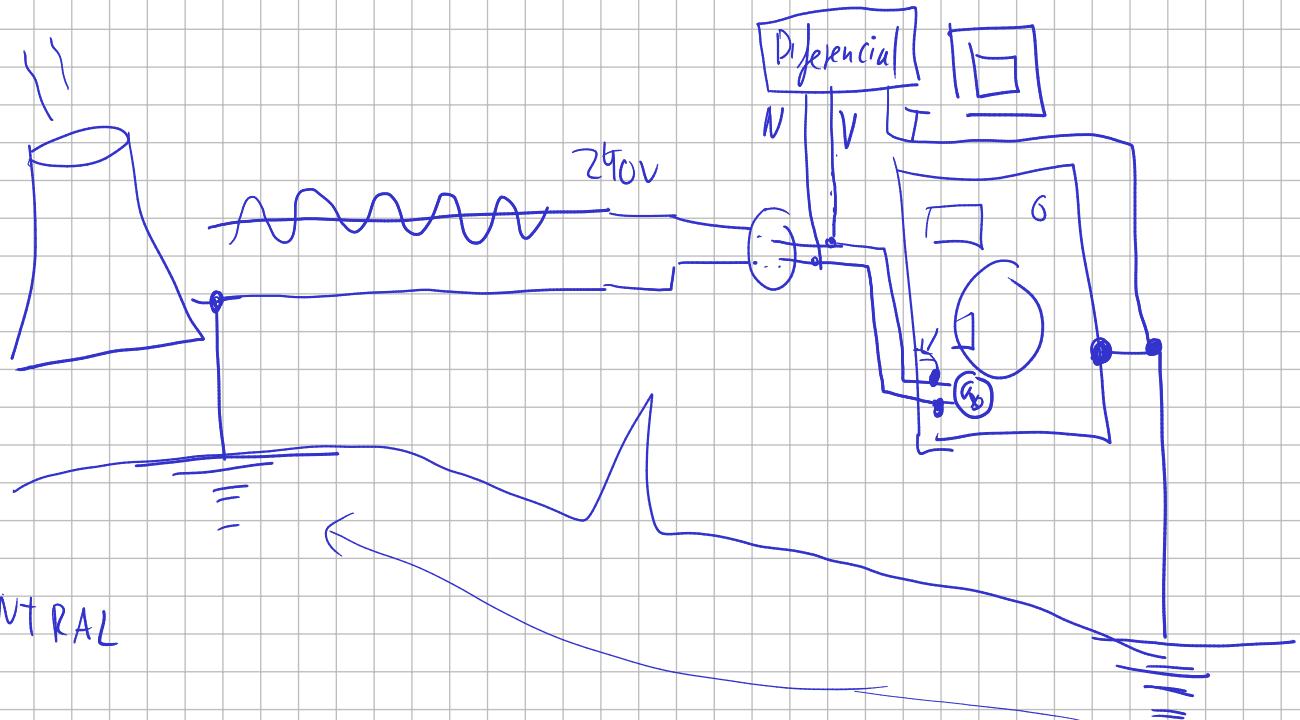
$\text{asig}, \text{exp} \rightarrow \text{noth}$

1FN

2FN

$(\underline{\text{Exp}}, \underline{\text{asig}}, \text{noth})$

$(\underline{\text{Exp}}, \text{nombre})$



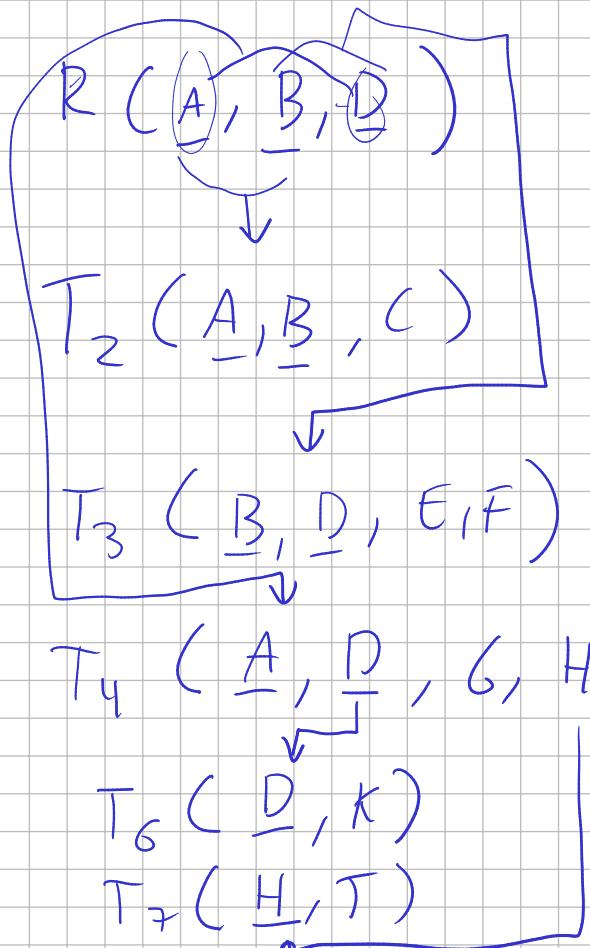
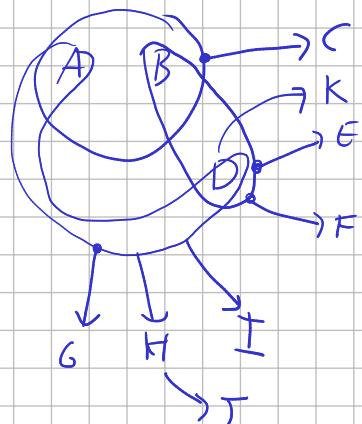
Sean: la relación $R = \{A, B, C, D, E, F, G, H, I, J, K\}$ y el conjunto de DF:

$$\begin{aligned} AB &\rightarrow C \\ BD &\rightarrow EF \\ AD &\rightarrow GHI \\ H &\rightarrow J \\ D &\rightarrow K \end{aligned}$$

Se pide:

Encontrar una clave de R y demostrar que es clave.
Descomponer el conjunto de relaciones resultantes en 3NF.

$T_1 (A, B, C, D, E, F, G, H, I, J, K)$	1FN	T_1, T_2, T_3, T_4, T_5
$T_5 (A, B, D)$	2FN	3FN
$T_1 (A, D, G, H, I, J)$	2FN	T_6, T_7
$T_2 (A, B, C)$	2FN	3FN
$T_3 (B, D, E, F)$	2FN	3FN
$T_4 (D, K)$	2FN	3FN
$T_6 (A, D, G, H, I)$	2FN	3FN
$T_7 (H, J)$	2FN	3FN
.		
$T_5 (A, B, D)$		$T_4 (D, K)$
$T_2 (A, B, C)$		$T_3 (B, D, E, F)$
$T_6 (A, D, G, H, I) \quad (H, J)$		



Para cada cliente: Número de cliente (único), Direcciones de envío (varias por cliente), Saldo, Límite de crédito (depende del cliente, pero en ningún caso debe superar los 3.000.000 pts), Descuento.

Para cada artículo: Número de artículo (único), Fábricas que lo distribuyen,

Existencias de ese artículo en cada fábrica, Descripción del artículo.

Para cada pedido: Cada pedido tiene una cabecera y el cuerpo del pedido.

La cabecera está formada por el número de cliente, dirección de envío y fecha del pedido.

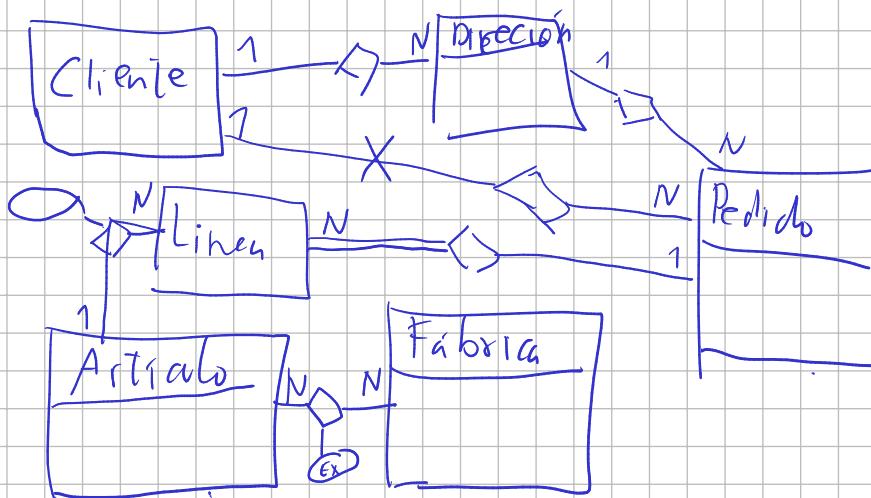
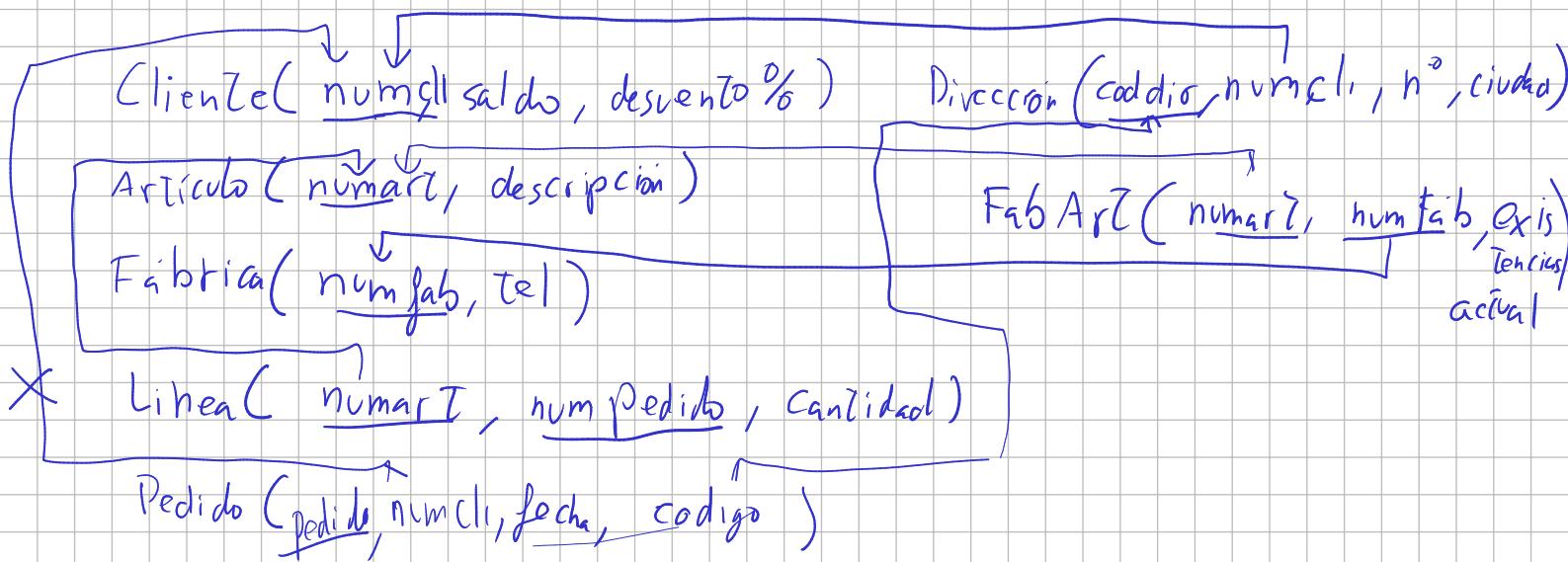
El cuerpo del pedido son varias líneas, en cada línea se especifican el número del artículo pedido y la cantidad.

Además, se ha determinado que se debe almacenar la información de las fábricas.

Sin embargo, dado el uso de distribuidores, se usará: Número de la fábrica (único) y Teléfono de contacto.

También, por información estratégica, se podría incluir información de fábricas alternativas respecto de las que ya fabrican artículos para esta empresa.

Nota: Una dirección se entenderá como Nº, Calle y Ciudad. Una fecha incluye hora.



Booleans

T F

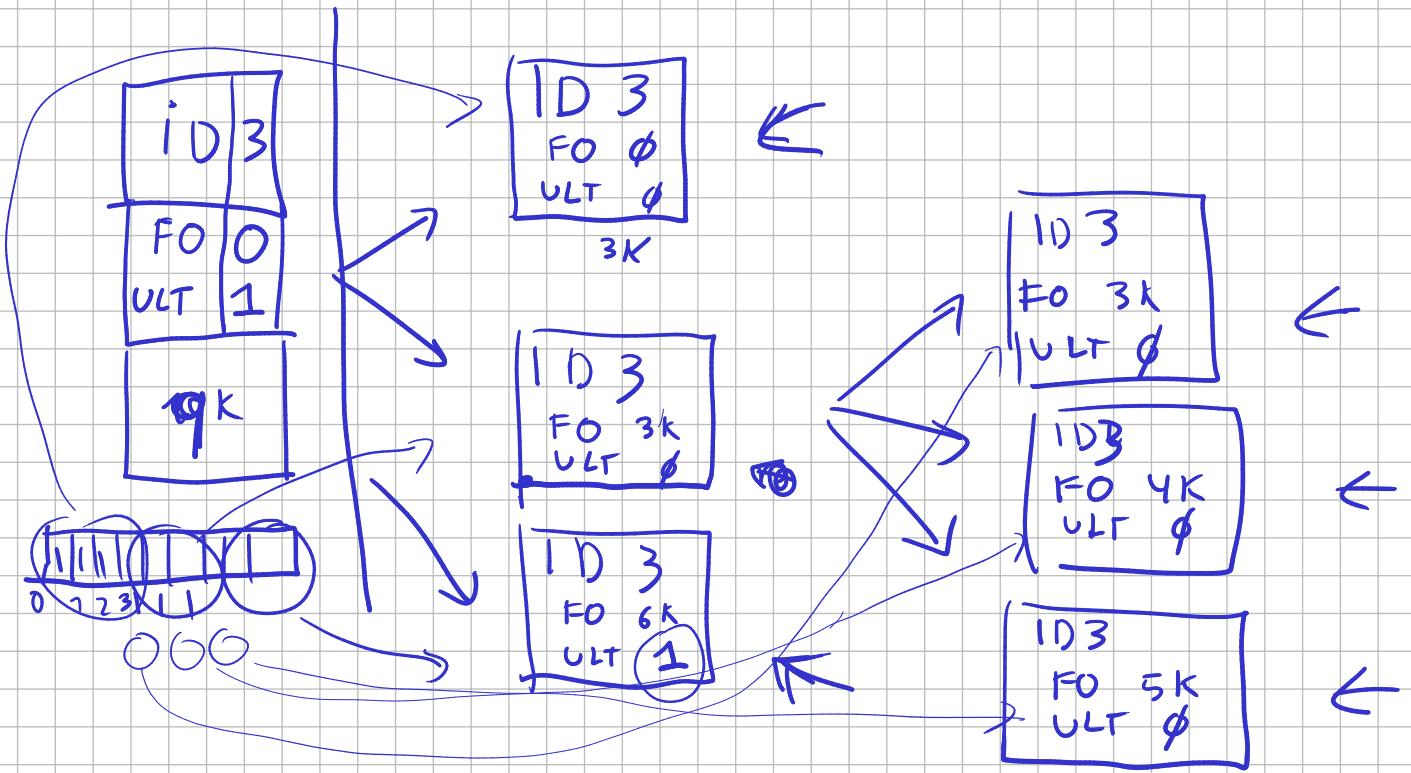
AND
OR
NOT

		GR	AND
F	F	F	F
F	T	T	F
T	F	T	F
T	T	T	T

	NOT
F	T
T	F

$$\rightarrow \text{NOT} (F \text{ and } (T \text{ or } \text{not}(F)))$$

Not T or T
F and F



AND		OR
F	F	F
T	F	T
F	F	T
T	T	T

F	NULL	F	NULL
T	NULL	NULL	T

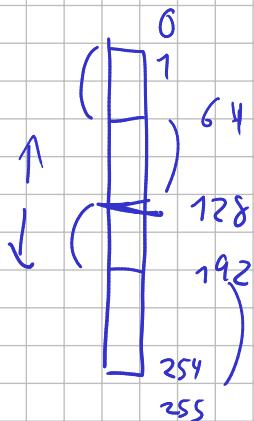
$y > \text{NULL} \rightarrow \text{NULL}$
 $y < \text{NULL} \rightarrow \text{NULL}$
 $y = \text{NULL} \rightarrow \text{NULL}$
 $\text{NULL} < 'hoh' \rightarrow \text{NULL}$
 $\text{NULL} = \text{NULL} \rightarrow \text{NULL}$
 $\text{NULL} \neq \text{null} \rightarrow T$
 $5 \text{ is null} \rightarrow F$

~~192.168.1.0/24~~

(254) $2^8 - 1 - 1$

11000000.10101000.00000001.00000000	11000000.10101000.00000001.00000000
11000000.10101000.00000001.00000001	1111100000000000
11000000.10101000.00000001.10000000	1111100000000000
11000000.10101000.00000001.11000000	1111100000000000

192.168.1.0/26 2^{6-2}
 192.168.1.64/26 2^{6-2}
 192.168.1.128/26 2^{6-2}
 192.168.1.192/26 2^{6-2}



$$2^6 - 2 = 64 - 2 = 62$$

$\times 4$

(248)

RED	1 st	ultimo	Broadcast
192.168.1.0/26	192.168.1.1	192.168.1.62	192.168.1.63
192.168.1.64/26	192.168.1.65	192.168.1.126	192.168.1.127
192.168.1.128/26	192.168.1.129	192.168.1.190	192.168.1.191
192.168.1.192/26	192.168.1.193	192.168.1.254	192.168.1.255

124 → 126:4

124 → 129:32

24.32.17.123/12 hos C

24.32.0.0/12 red

112 → /14

24.0010|0000.0.0

24	001000	00.0.0.0	→ 24.32.0.6/14
24	001001	00.0.0.0	24.36.0.0/14
24	001010	00.0.0.0	24.40.0.0/14
24	001011	00.0.0.0	24.44.0.0/14

red	1 ^o	último	Broadcast
24.32.0.6/14	24.32.0.1	24.35.255.254	↓1
24.36.0.0/14	24.36.0.1	24.39.255.254	+1
24.40.0.0/14	24.40.0.1	24.43.255.254	+1
24.44.0.0/14	24.44.0.1	24.47.255.254	+1

16 → 120 → 16 redes

129.192.0.0/16

- Dividir en 2 → 2¹ mitad libre A

- Con la 1^o mitad → C

- Dividir en 2

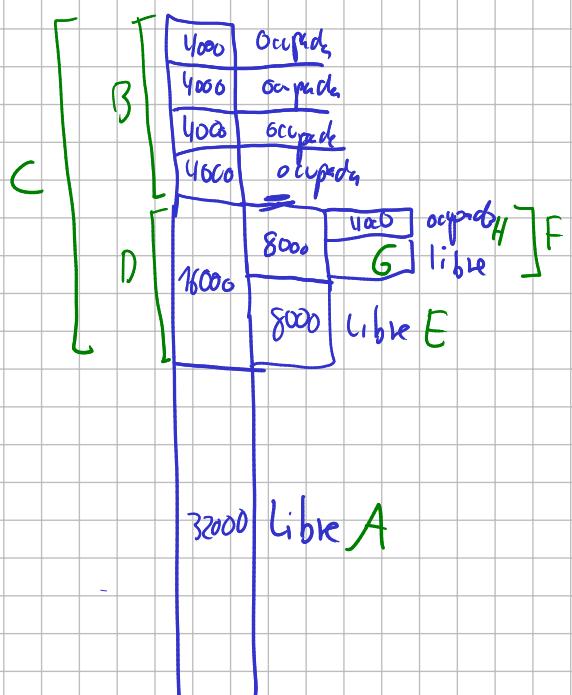
B - Con la 1^o mitad → 11 redes ocupadas

D - Con la 2^o mitad

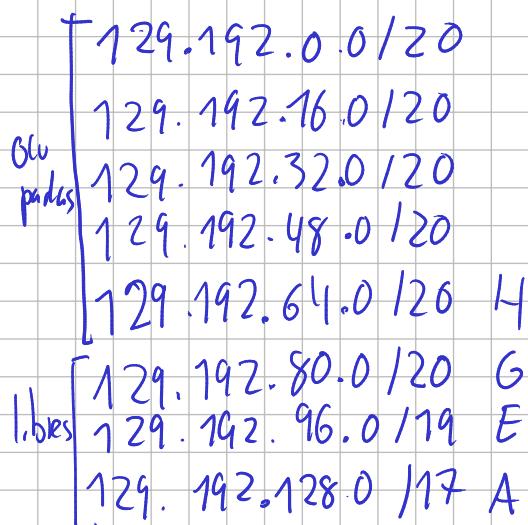
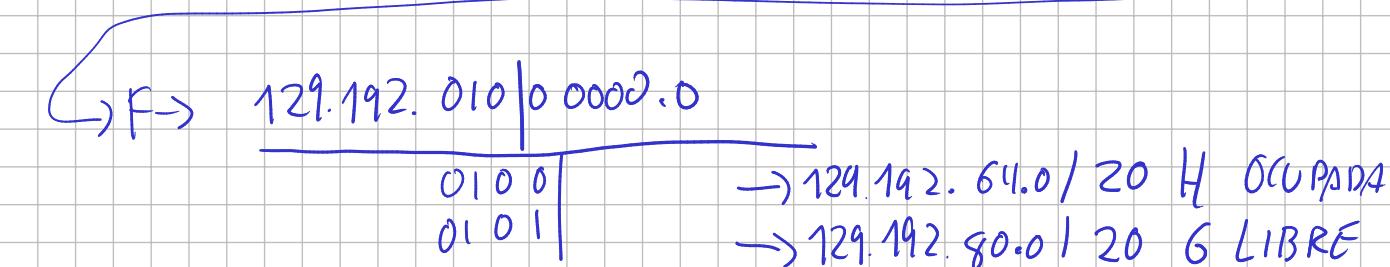
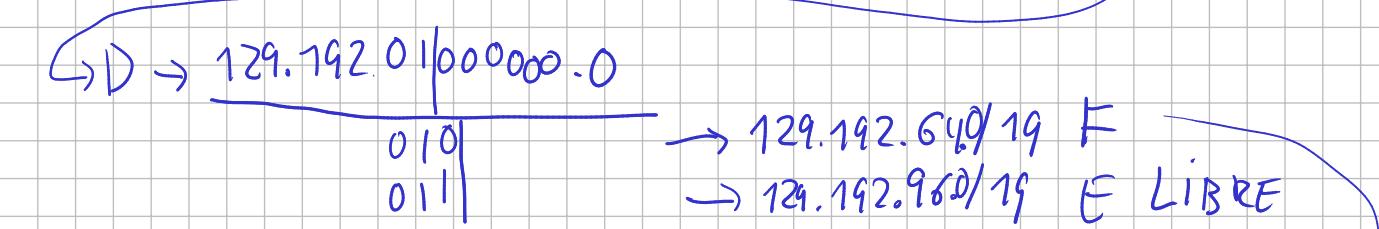
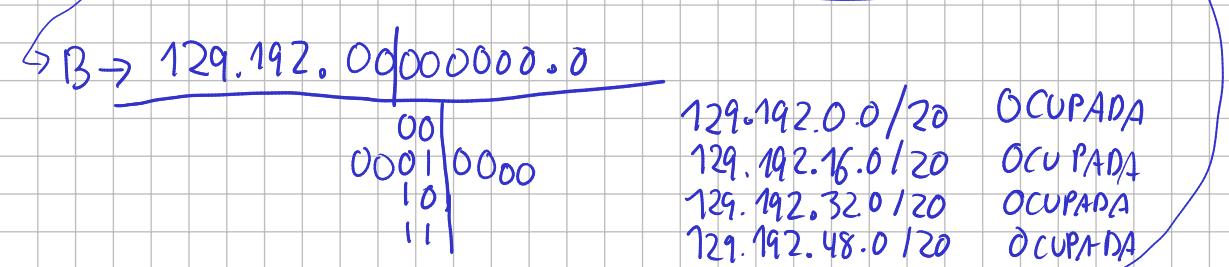
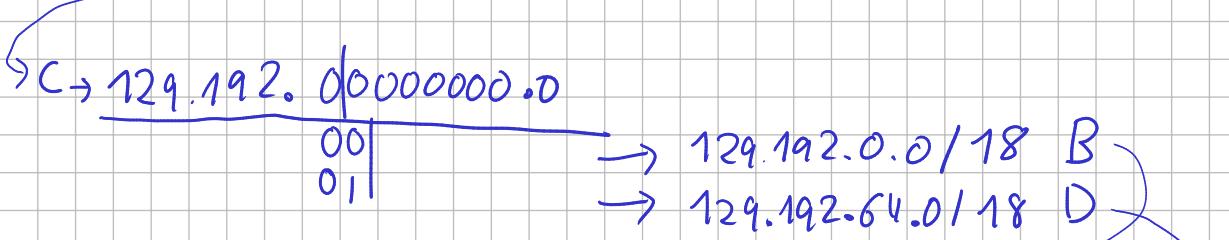
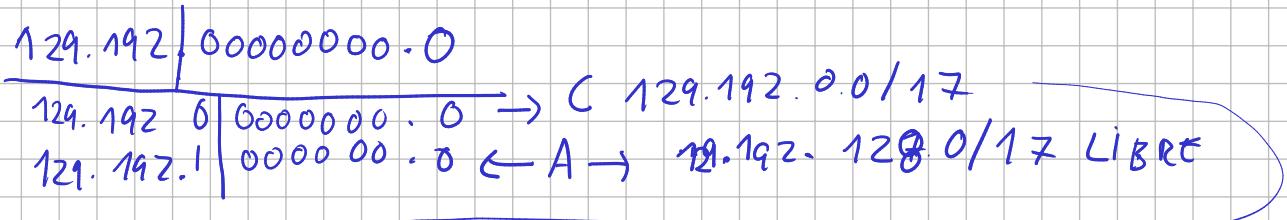
- Partir en 2 → 2¹ mitad libre E

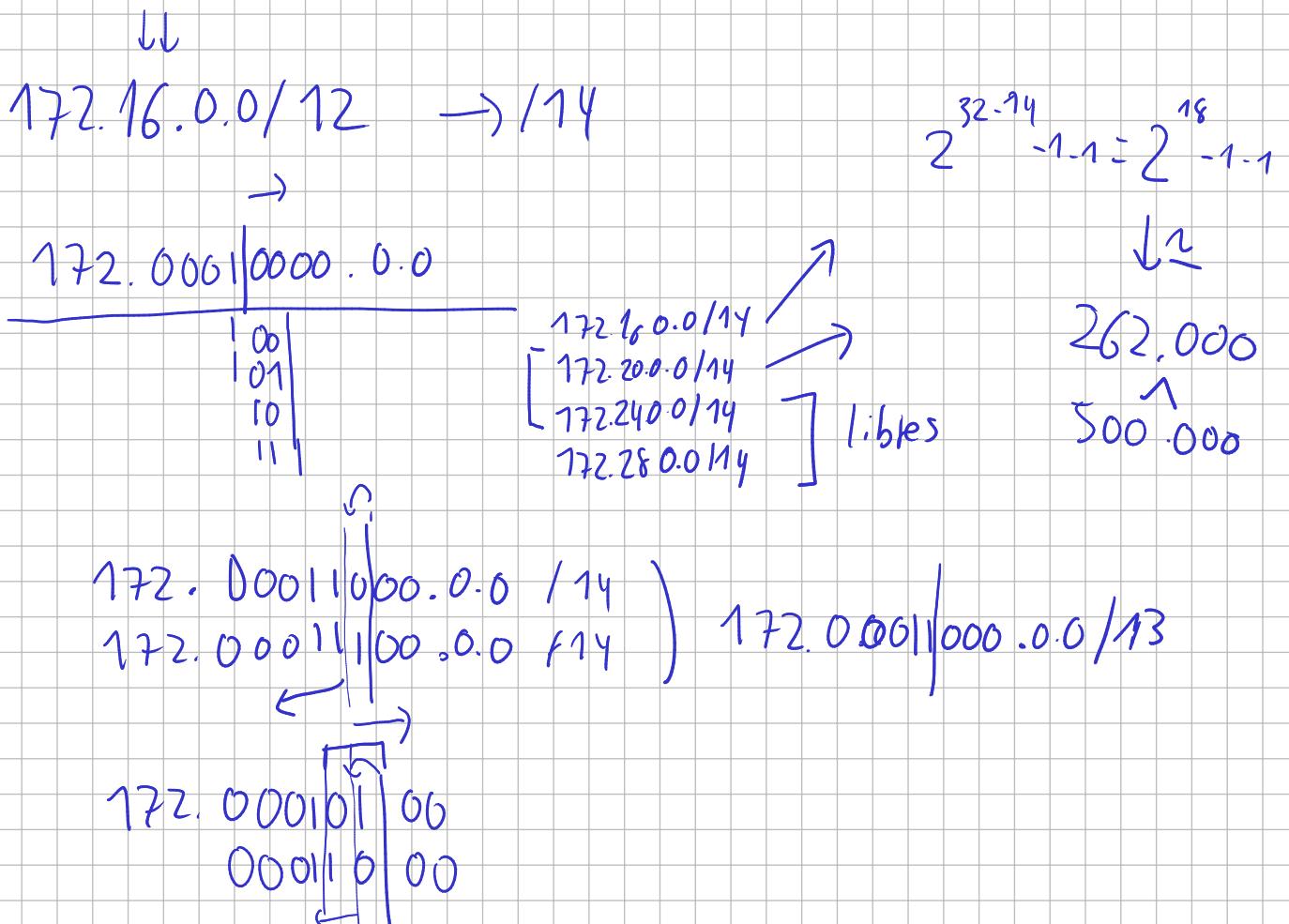
F - 1^o mitad, en 2 → 2¹ mitad libre G

- 1^o mitad → ocupada H



129.192.0.0/16





$[192.168.23.0/24]$
 $[192.168.22.6]$

$[192.168.23.0/24]$
 $[192.168.24.0]$

$[192.168.24.0/24]$
 $[25]$
 $[26]$
 $[27]$

$[192.168.22.0/24]$
 $[23]$
 $[24]$
 $[25]$

↓
2001:0DB8:7200::/39, se desea dividirla en 8 redes

↑ 16 ↑ 32 ↑ 40 ↑ 48

39	0
7 001000	00
7 001001	00
7 001010	00
7 001011	00
7 001100	00
7 001101	00
7 001110	00
7 001111	00

→ 2001:0DB8:7200::/42

2001:0DB8:7240::

2001:0DB8:7280::

2001:0DB8:72C0::

2001:0DB8:7300::

2001:0DB8:7340::

2001:0DB8:7380::

2001:0DB8:73C0::

RED

1º

ÚLTIMO

2001:0DB8:7200::/42
2001:0DB8:7240::/42
2001:0DB8:7280::/42
2001:0DB8:72C0::/42

2001:0DB8:7300::/42
2001:0DB8:7340::/42
2001:0DB8:7380::/42
2001:0DB8:73C0::/42

2001:0DB8:7200::1

2001:0DB8:7240::1

2001:0DB8:7380::1

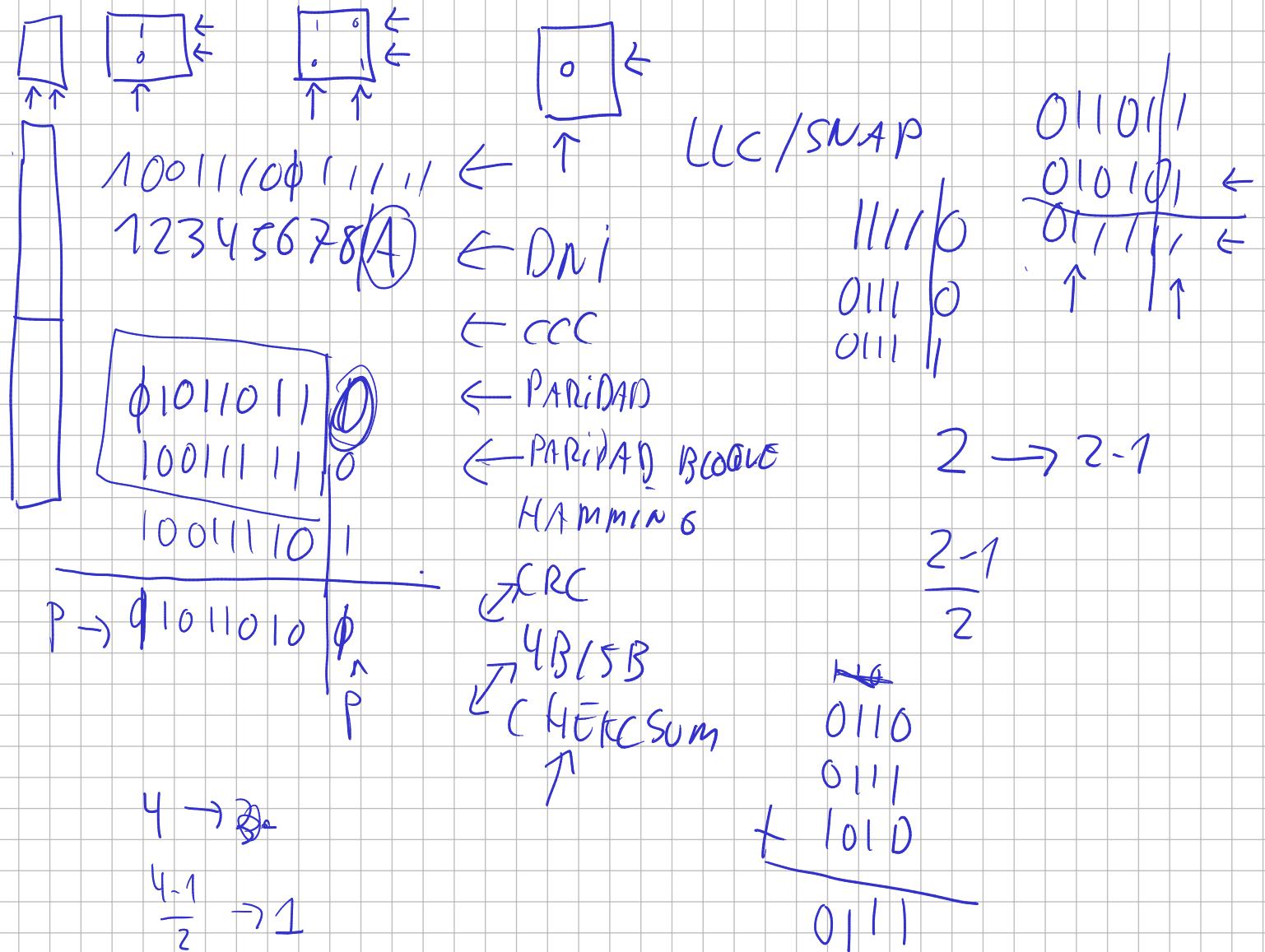
2001:0DB8:73C0::1

2001:0DB8:723F:FFFF:FFFF:FFFF:FFFF:
FFFF:

2001:0DB8:73BF:FFFF:FFFF:FFFF:FFFF:
FFFF:

2001:0DB8:73FF:FFFF:FFFF:FFFF:FFFF:
FFFF:

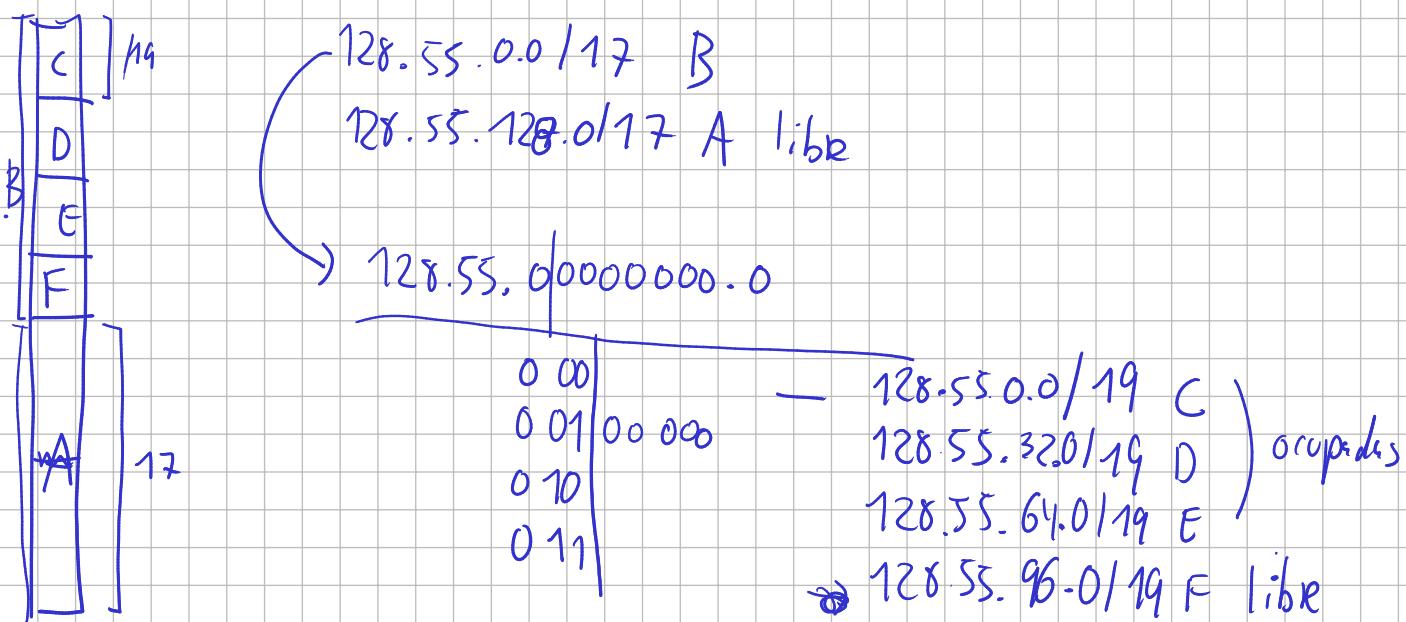
$$n^{\circ} \text{ hosts } 2^{128-42} - 1 = 2^{86} - 1$$



ubnetting

Conseguir 3 redes de 5000 hosts a partir de una red de clase B

128.55.0.0/16 → 19 5000 → 13 bits de host
19 red



1. Un host tiene la dirección ::f482:667B:2422, con máscara /110. Calcula su red, y a partir de ahí, el primer host de su red, el último y el número máximo de hosts en su red

2. Se tiene una red IP con dirección 30.23.0.0/17. Se necesita dividir esta red para conseguir 5 subredes iguales, lo más grandes posible. Describe (en forma de tabla), las redes resultantes, su primer host, su último host, y el número total de hosts. Indica también si queda alguna red sin usar tras la división, y cuáles son.

3. Por una línea se van a enviar los siguientes datos: 1100001, 0011010, 1110001, 1100111, 0010100. Calcula la tabla de paridad de bloque de este mensaje

↓
 ::f482:667B:2422/110 ip
 ↓
 ::f482:6678:0/110 red
 ::f482:6678:1/110 primo
 ::f482:6678:FFFF/110 último

B → 1011
 ↑
 ↓
 2¹⁰⁸⁻¹¹⁰⁻¹
 2¹⁸⁻¹ = 262143

Ejercicio 1

30.23.0.0/17 → /20
 30.23.00000000.0

000	30.23.0.0/20	ocupados	↑	30.23.00001111.11111110
001	30.23.16.0/20		↑	30.23.15.254
010	30.23.32.0/20		↑	30.23.31.254
011	30.23.48.0/20		↑	30.23.47.254
100	30.23.64.0/20		↑	30.23.48.254
101	30.23.80.0/20		↑	30.23.63.254
110			↑	30.23.79.254
111			↑	30.23.95.254

Libre

3.23.96.0/19 libre

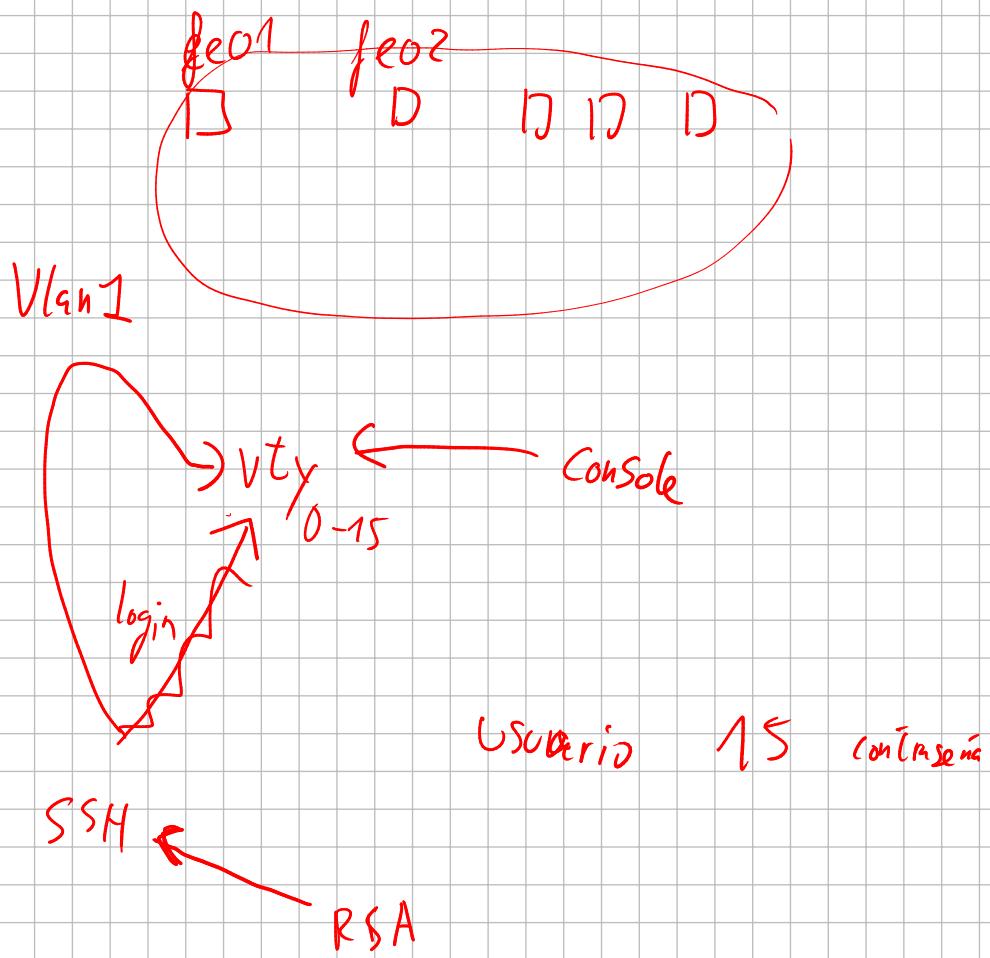
↑
 2¹²⁻²
 2¹³⁻²

Ejercicio 2

Par

1	1	0	0	0	0	1	1
0	0	1	1	0	1	0	0
1	1	1	0	0	0	1	0
1	1	0	0	1	1	1	1
0	0	1	0	1	0	0	0
1	1	1	1	0	0	1	1

Ejercicio 3



→ Pregunto

~~ber~~

blocueo



—

