

Quien tiene un  
algoritmo...



...¡tiene un  
tesoro!

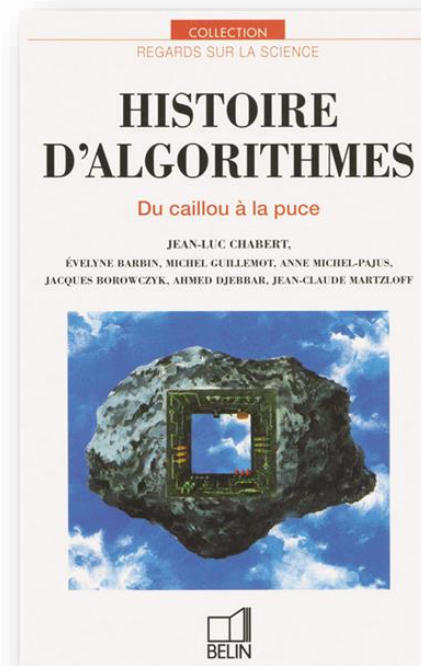


**Unas primeras ideas...**



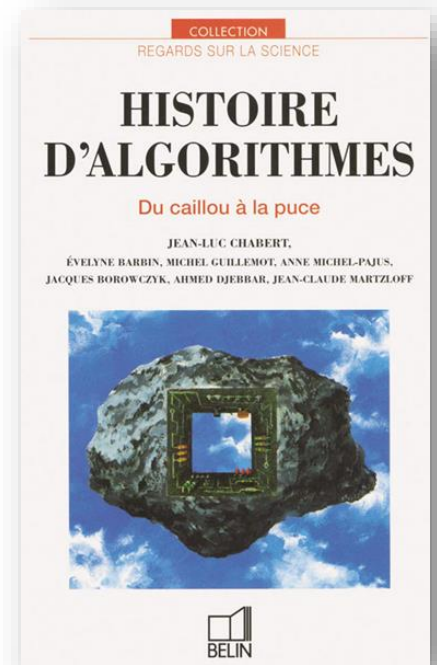
## ¿Qué es un algoritmo?

Un conjunto finito y organizado de instrucciones que ha de dar respuesta a un problema.



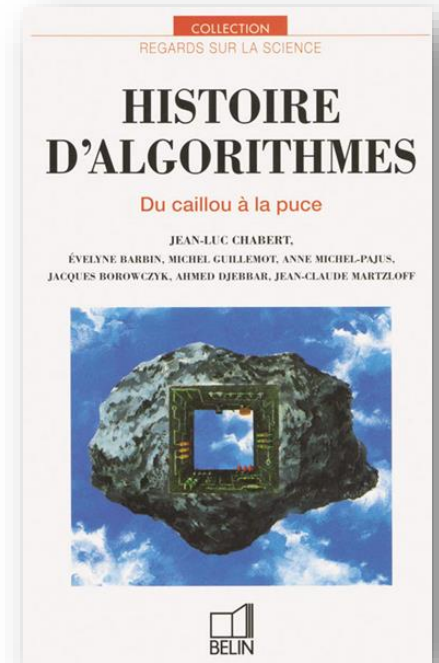
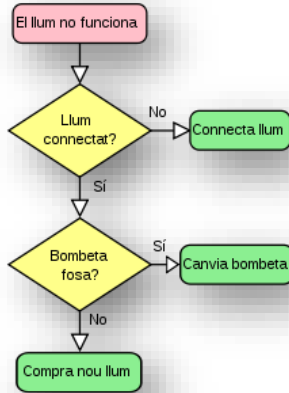
## Algunas características de un algoritmo

- Es un proceso que se ejecuta paso a paso.
- La acción de cada etapa está estrictamente determinada.
- La ejecución se acaba en un número finito de pasos.



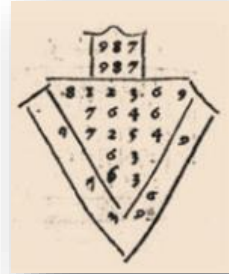
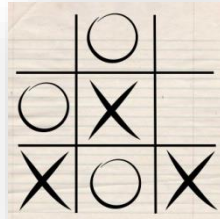
## Algunas características de un algoritmo?

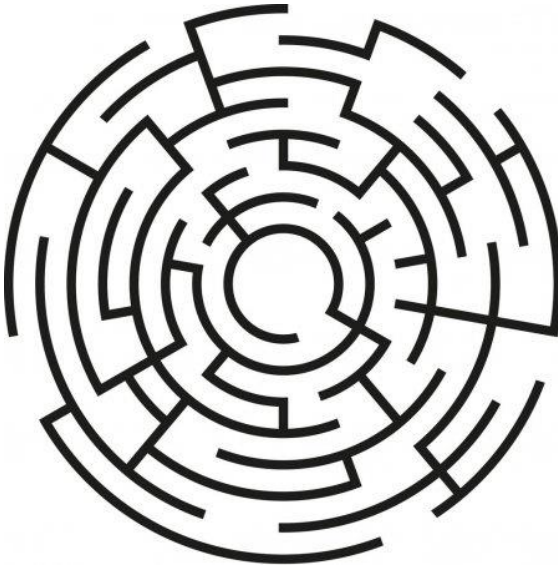
- El algoritmo ha de poder ser escrito en un cierto lenguaje.



## Organización de la sesión

- Aplicar
- Interpretar-Justificar
- Construir





**Apliquemos**

## Laberintos





## Laberintos



Catedral de Chartres



Catedral de Amiens

## Laberintos



Laberinto de Horta

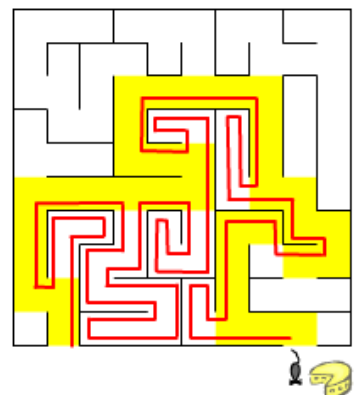
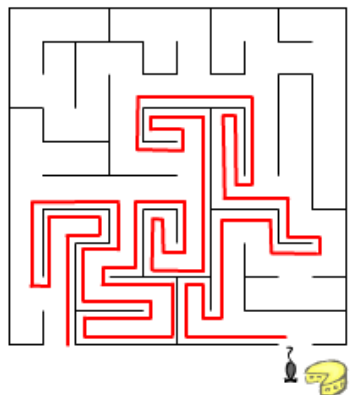
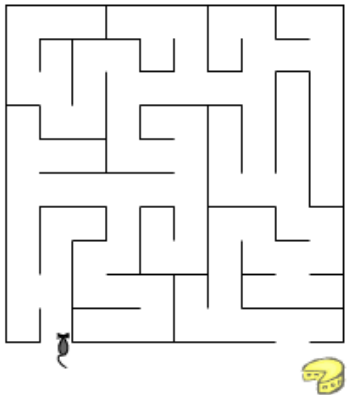
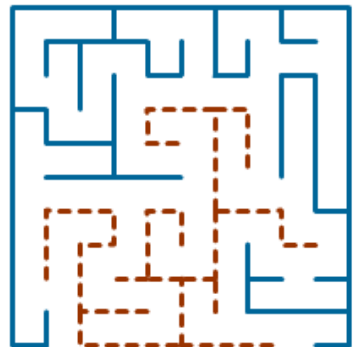
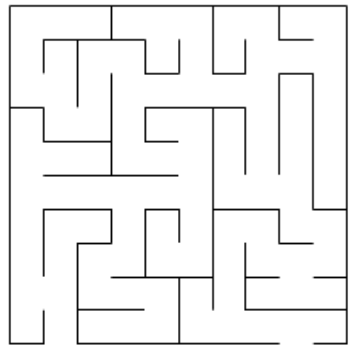


Laberinto de Hampton Court

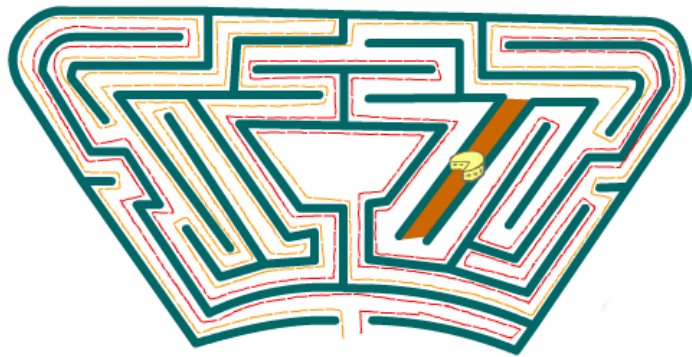
# Laberintos



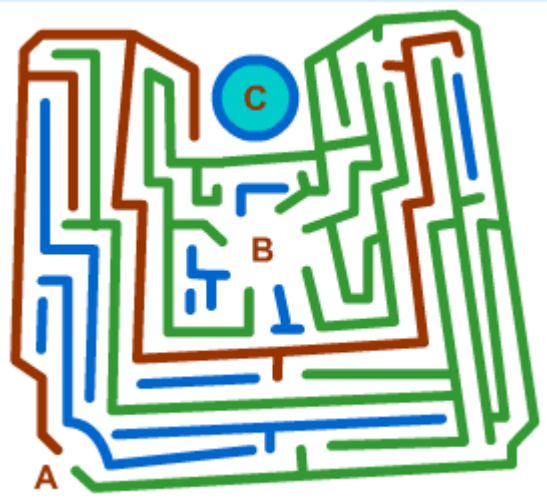
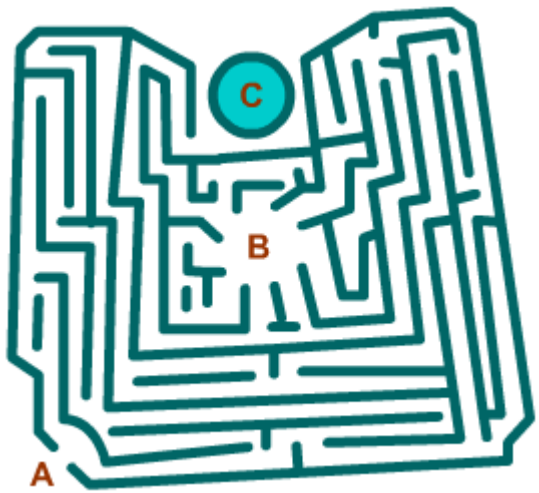
# Laberintos



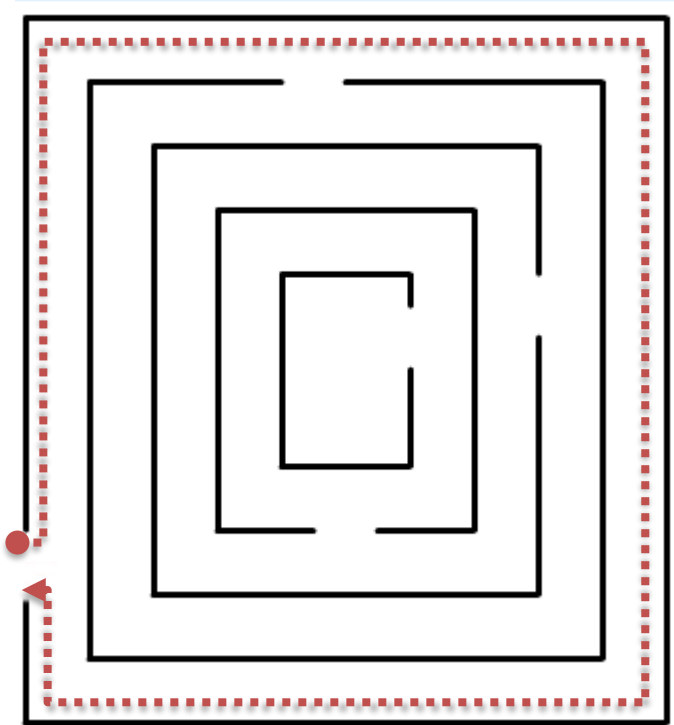
# Laberintos



# Laberintos



# Laberintos



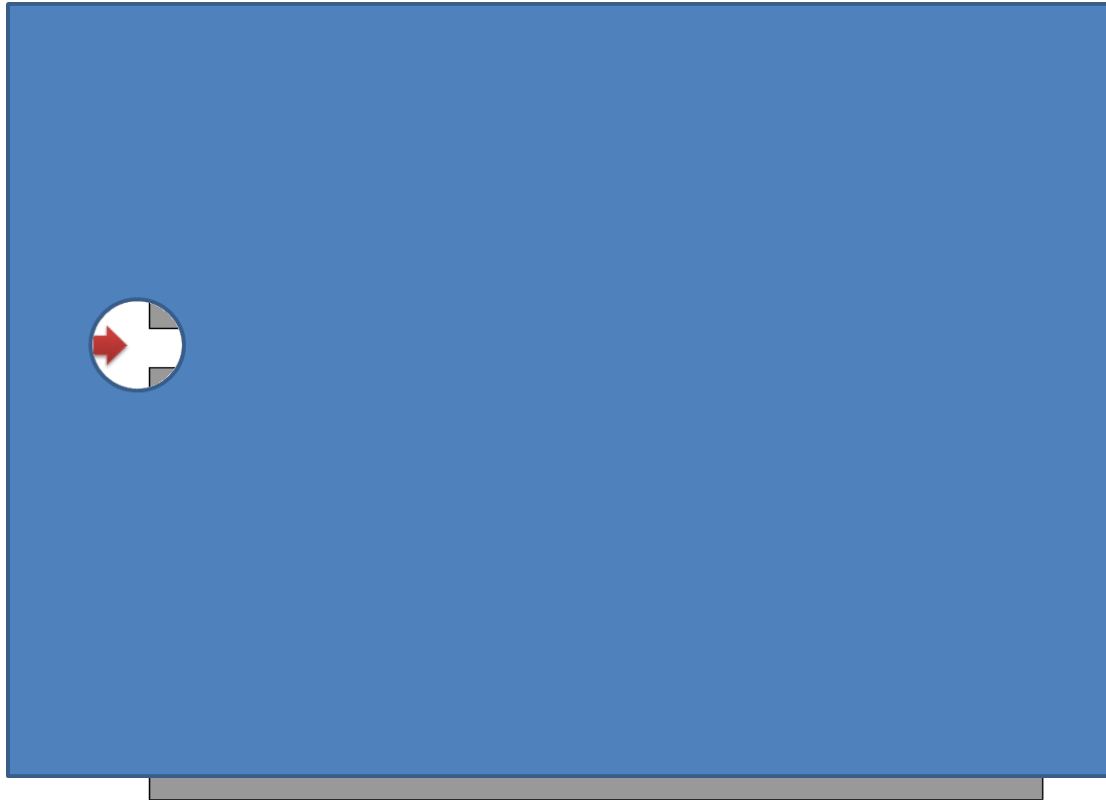
# Laberintos

## Algoritmo de Trémaux





## Laberintos

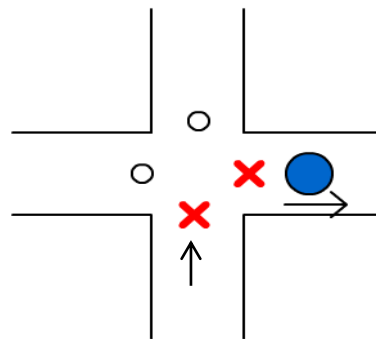
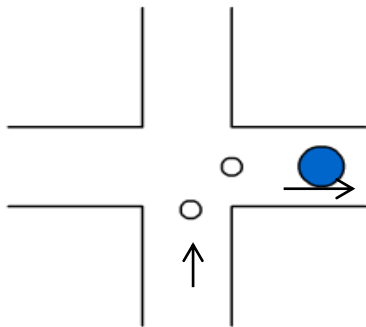


## Laberintos

### Algoritmo de Trémaux

Dos tipos de marcas:

- Una para cuando entramos o salimos de un pasillo por primera vez (Abrimos).
- Otra para cuando volvemos a pasar por encima de esta marca (Cerramos).

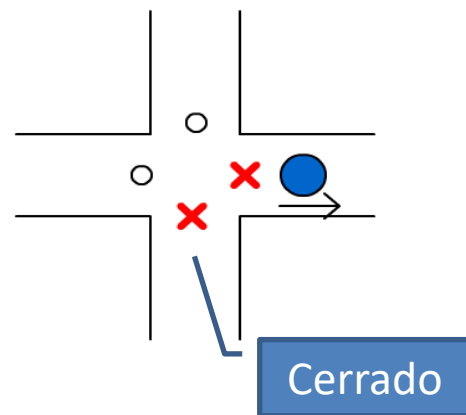
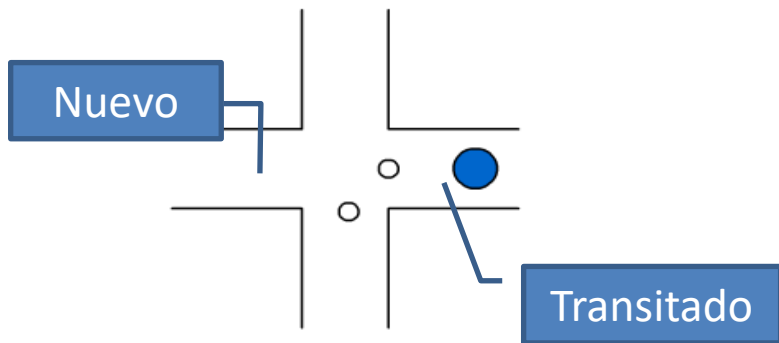


## Laberintos

### Algoritmo de Trémaux

Vocabulario:

- *Camino nuevo*: no hemos pasado nunca.
- *Camino transitado*: está marcado, hemos pasado una vez.
- *Camino cerrado*: no hemos de volver a pasar (ya hemos “ido y vuelto”)



## Laberintos

### Algoritmo de Trémaux

- 1) Llegamos a un cruce “nuevo”, por el que no hemos pasado nunca, y por un camino “nuevo”.

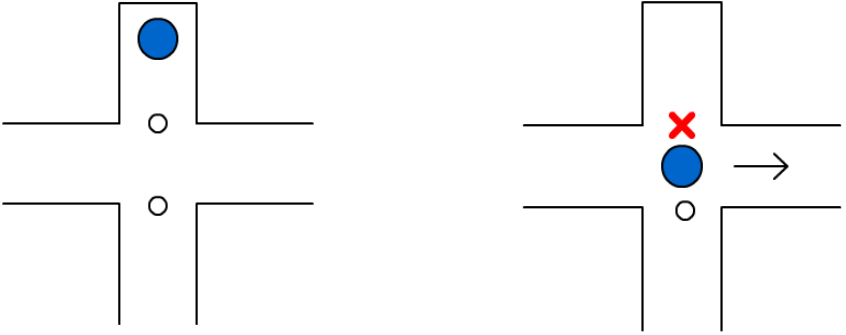


- Elegimos un nuevo camino al azar.
- Marcamos el que dejamos y el que tomamos.

# Laberintos

## Algoritmo de Trémaux

2) Si llegamos a un camino sin salida...

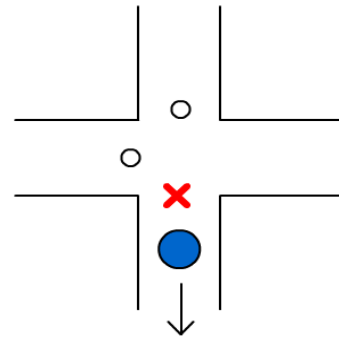
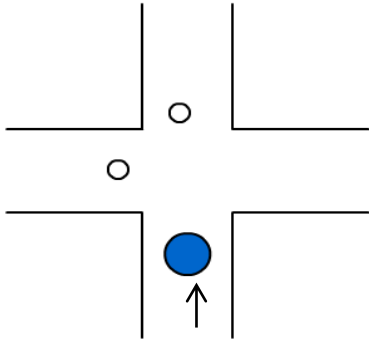


- Retrocedemos, lo cerramos y seguimos andando.

## Laberintos

### Algoritmo de Trémaux

3) Si llegamos por un camino “nuevo” a un cruce en el que vemos algún camino transitado...

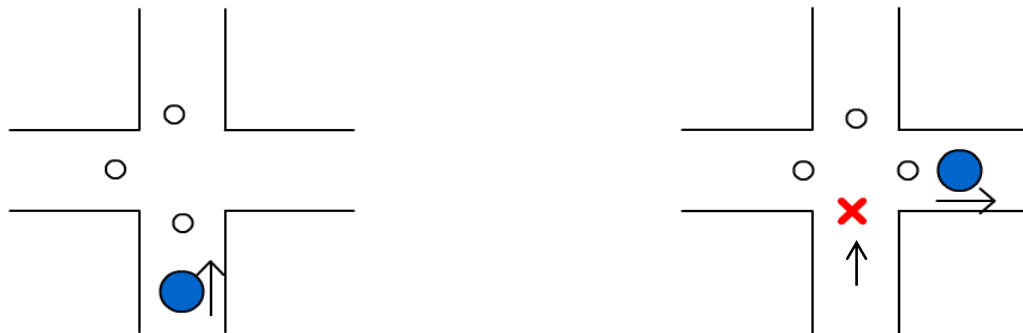


- Cerramos el camino por el que veníamos y retrocedemos.
- Cuando lleguemos a la entrada cerramos el camino que dejamos y continuamos.

## Laberintos

### Algoritmo de Trémaux

4) Si llegamos a un cruce por un camino “transitado” y hay algún camino “nuevo”...

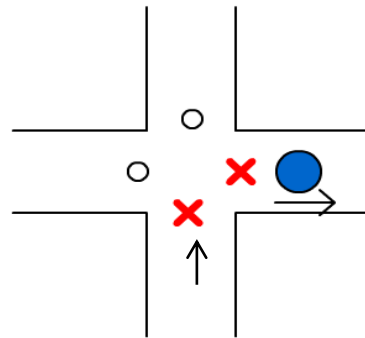
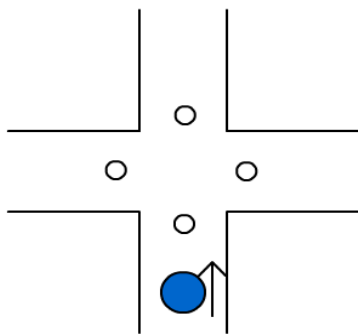


- Cerramos el camino por el que veníamos.
- Tomamos el camino “nuevo” y lo marcamos como “transitado”.

## Laberintos

### Algoritmo de Trémaux

5) Si llegamos a un cruce por un camino “transitado” y en el cruce están todos “transitados” o “cerrados” ...



- Cerramos el camino por el que veníamos.
- Tomamos uno de los “transitados” y lo remarcamos como “cerrado”.



# Quien tiene un algoritmo, tiene un tesoro

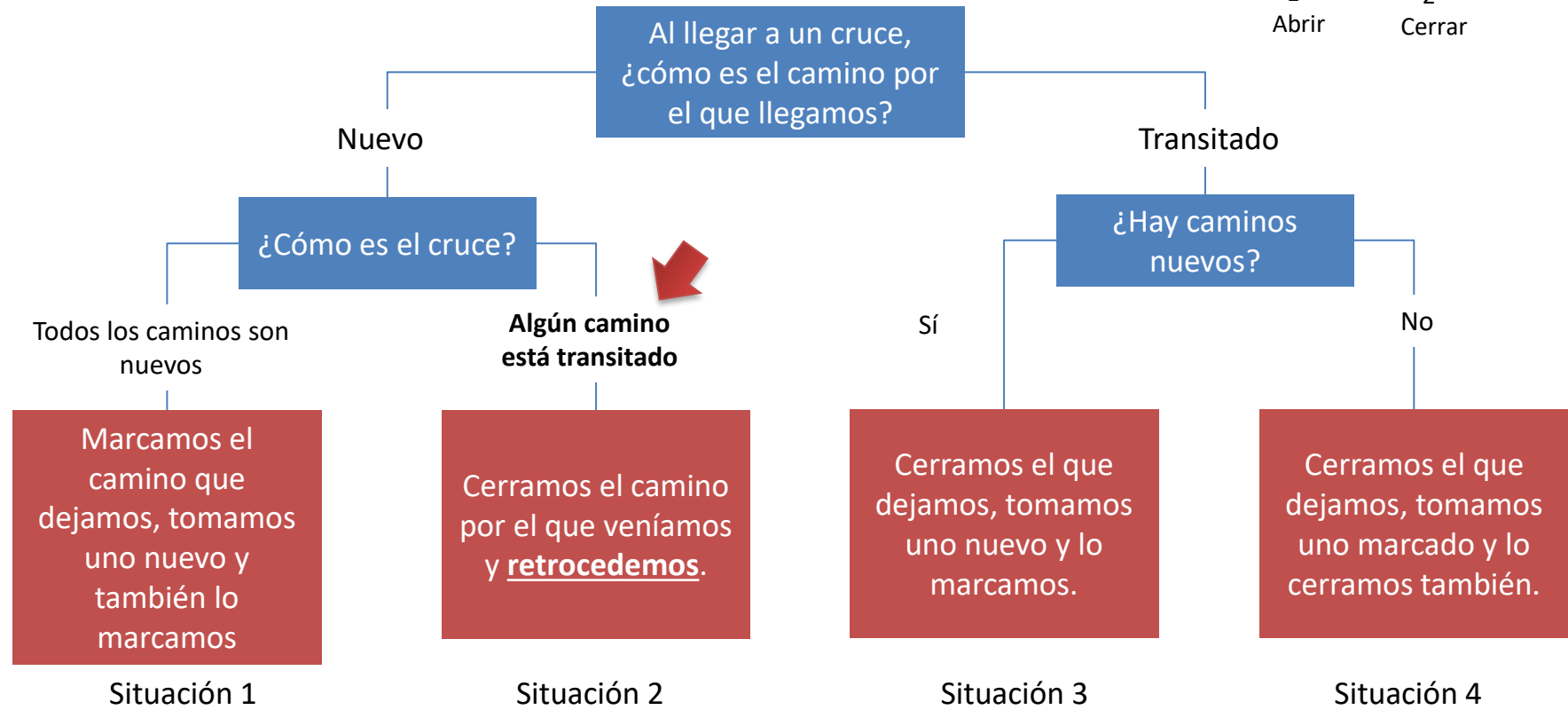
Siempre marcamos el camino por el que venimos y el que cogemos



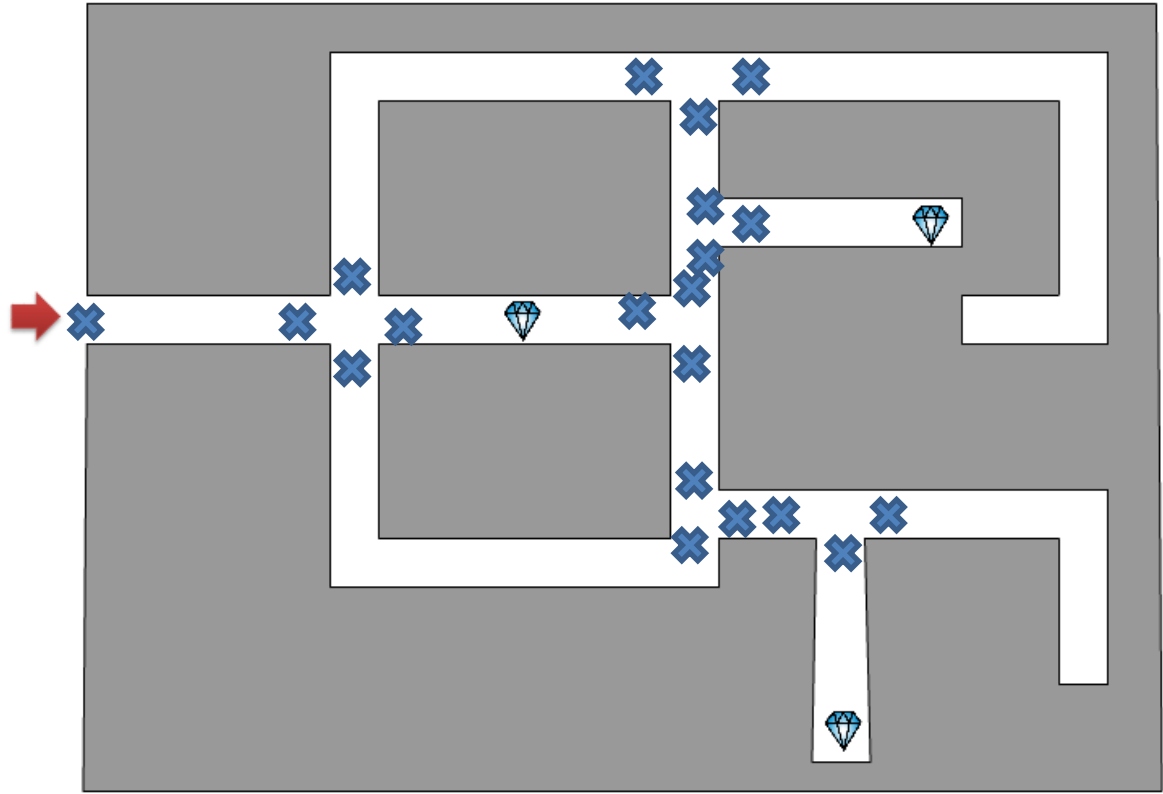
1ª  
Abrir



2ª  
Cerrar



# Laberintos



Quien tiene un algoritmo, tiene un tesoro



**Interpretemos - Justifiquemos**

# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



- 1 ————— 627
- 2 ————— 1254
- 4 ————— 2508
- 8 ————— 5016
- 16 ————— 10032
- 32 ————— 20064
- 64 ————— 40128
- 128 ————— 80256
- 256 ————— 160512

# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



1	—————	627
2	—————	1254
4	—————	2508
8		5016
16		10032
32		20064
64		40128
128	—————	80256
256	—————	160512

# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



- 1 ————— 627
- 2 ————— 1254
- 4 ————— 2508
- 8 ————— 5016
- 16 ————— 10032
- 32 ————— 20064
- 64 ————— 40128
- 128 ————— 80256
- 256 ————— 160512

# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



1 ————— 627

2 ————— 1254

4 ————— 2508

8 ————— 5016

16 ————— 10032

32 ————— 20064

64 ————— 40128

128 ————— 80256

256 ————— 160512

# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



1 ————— 627

2 ————— 1254

4 ————— 2508

8 ————— 5016

16 ————— 10032

32 ————— 20064

64 ————— 40128

128 ————— 80256

256 ————— 160512



# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



1	627
2	1254
4	2508
8	5016
16	10032
32	20064
64	40128
128	80256
256	160512

# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

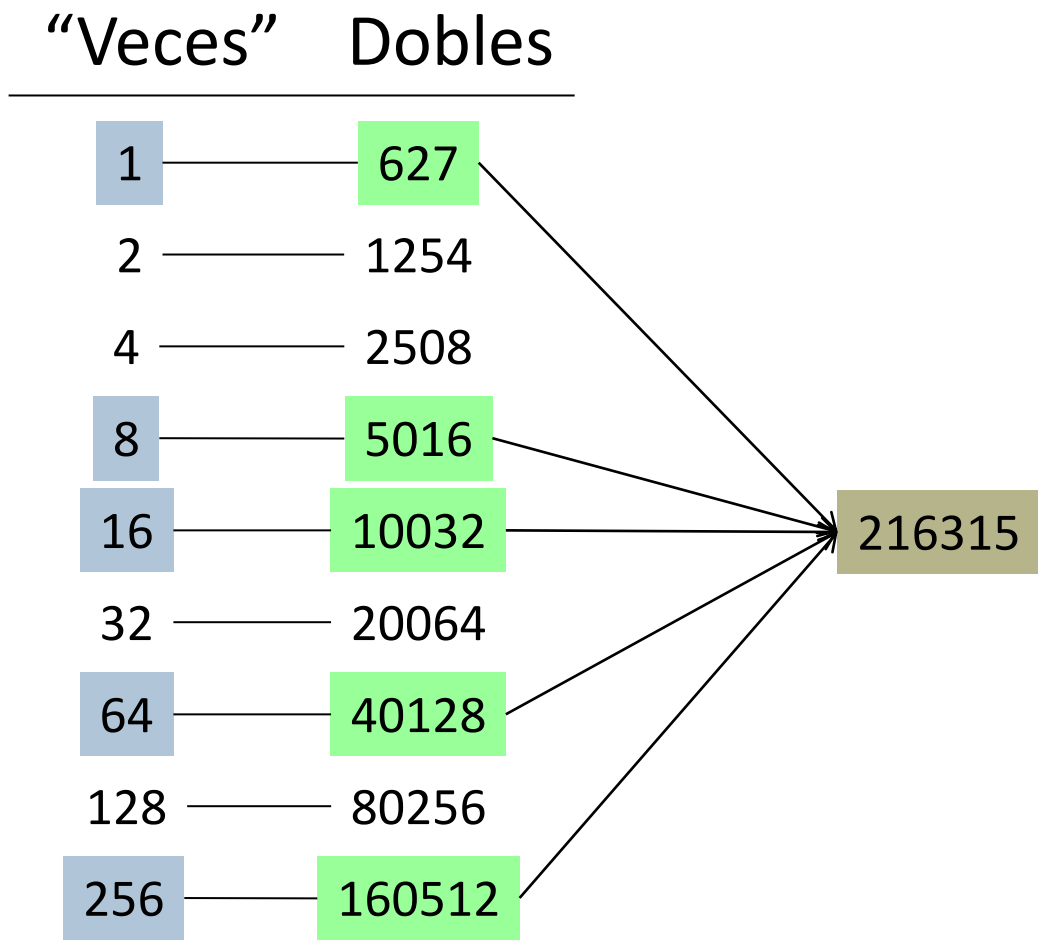
345 x 627

216315



## Retos:

- Explicar cómo funciona el algoritmo
- Justificarlo
- Adaptarlo a la división

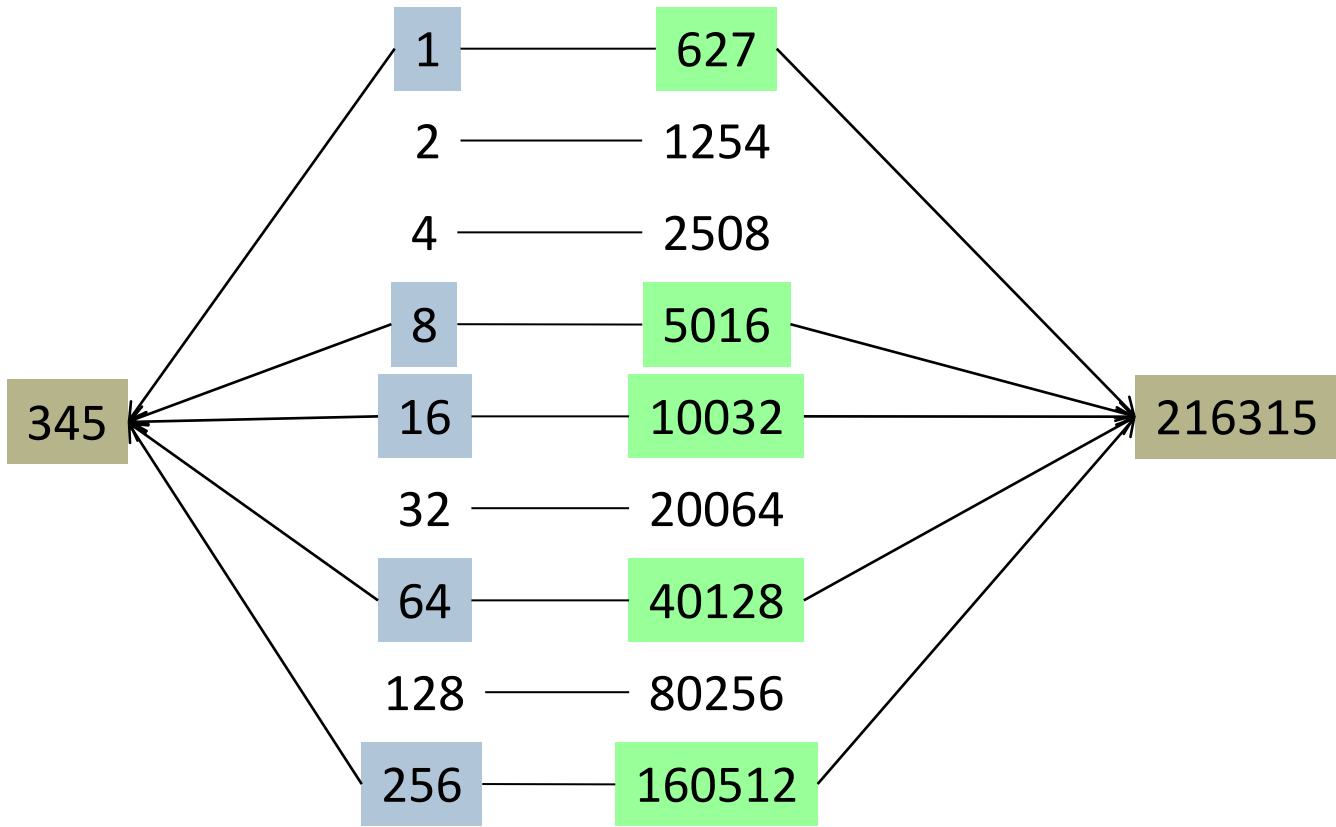


# Quien tiene un algoritmo, tiene un tesoro

Multiplicación egipcia

345 x 627

216315



# Quien tiene un algoritmo, tiene un tesoro

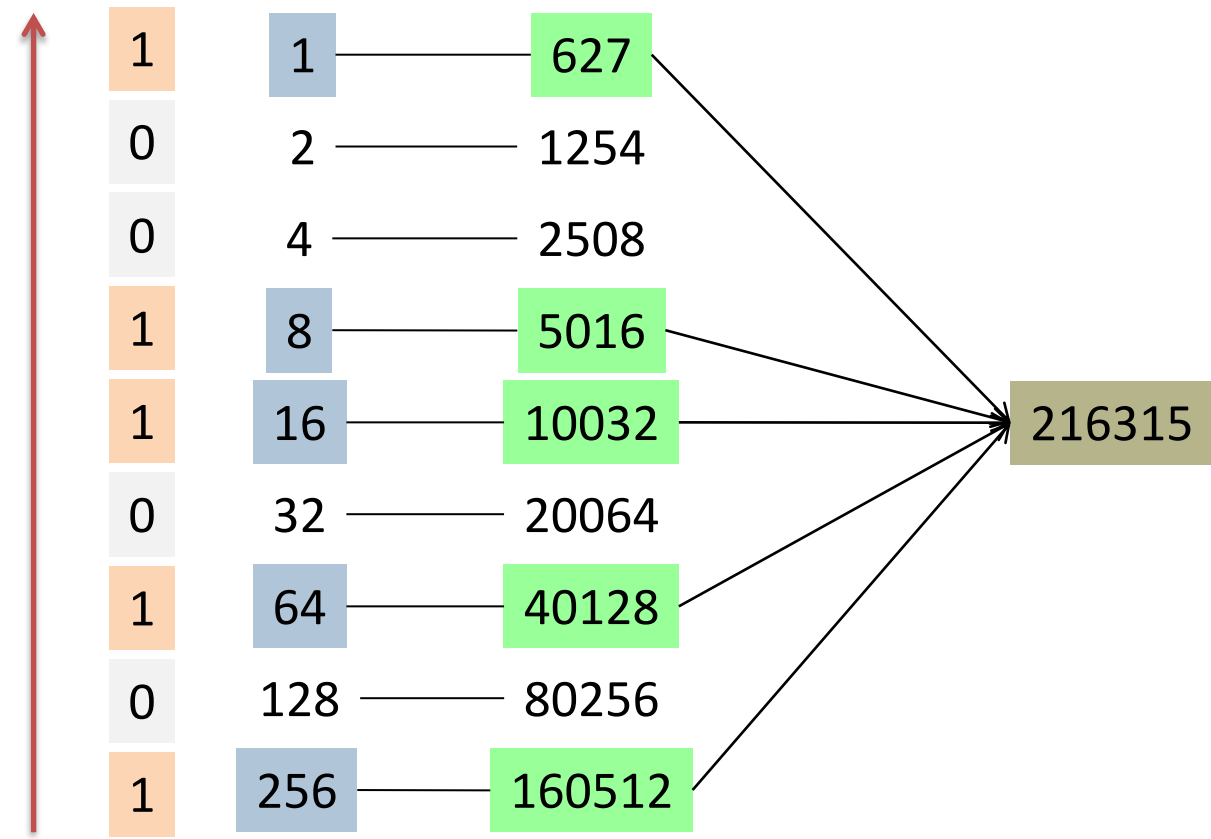
Multiplicación egipcia

345 x 627

216315



$$256 + 64 + 16 + 8 + 1 = 345_{10} = 101011001_2$$



# Quien tiene un algoritmo, tiene un tesoro

## Multiplicación egipcia

$$345 \times 627$$

$$216315$$



## Multiplicación egipcia

“Veces”      Dobles

1	—	627
2	—	1254
4	—	2508
8	—	5016
16	—	10032
32	—	20064
64	—	40128
128	—	80256
256	—	160512

## Multiplicación rusa

Mitades      Dobles

345	—	627
172	—	1254
86	—	2508
43	—	5016
21	—	10032
10	—	20064
5	—	40128
2	—	80256
1	—	160512

$$345 \times 627 = 627 + 5016 + 10032 + 40128 + 160512 = 216315$$

# Quien tiene un algoritmo, tiene un tesoro



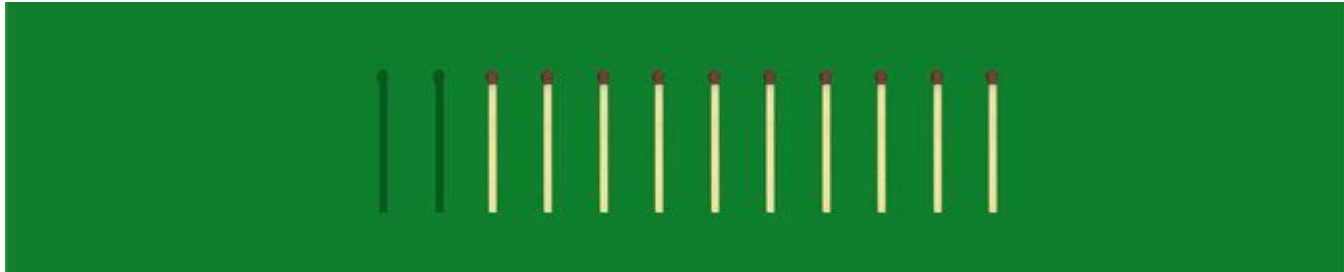
$$519 : 23$$

	“Veces”	Dobles	
	1	23	
	2	46	13
22	4	92	59
	8	184	
	16	368	151
		506	

Cociente: 22

Resto: 13

Quien tiene un algoritmo, tiene un tesoro



**Construimos**

## Juego de estrategia



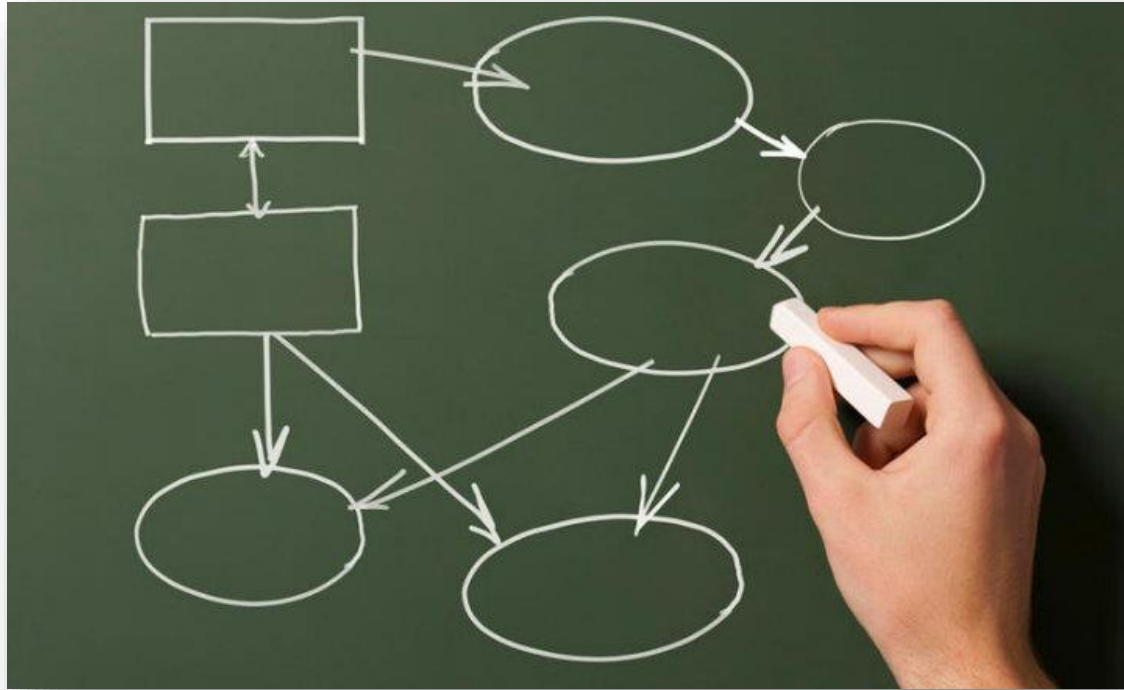
No interviene el azar  
ni la habilidad física.



Los jugadores disponen de  
información completa.

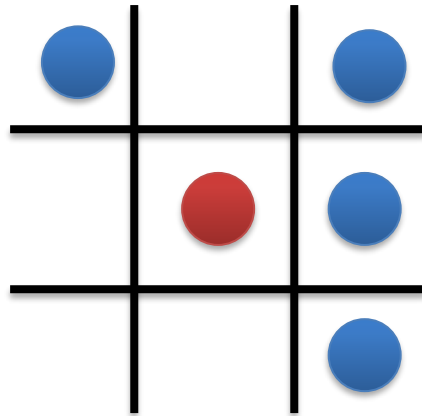


# Quien tiene un algoritmo, tiene un tesoro



Existe una estrategia para el primer jugador, para el segundo o para que uno de los dos provoque tablas.

Ta Te Ti Loco



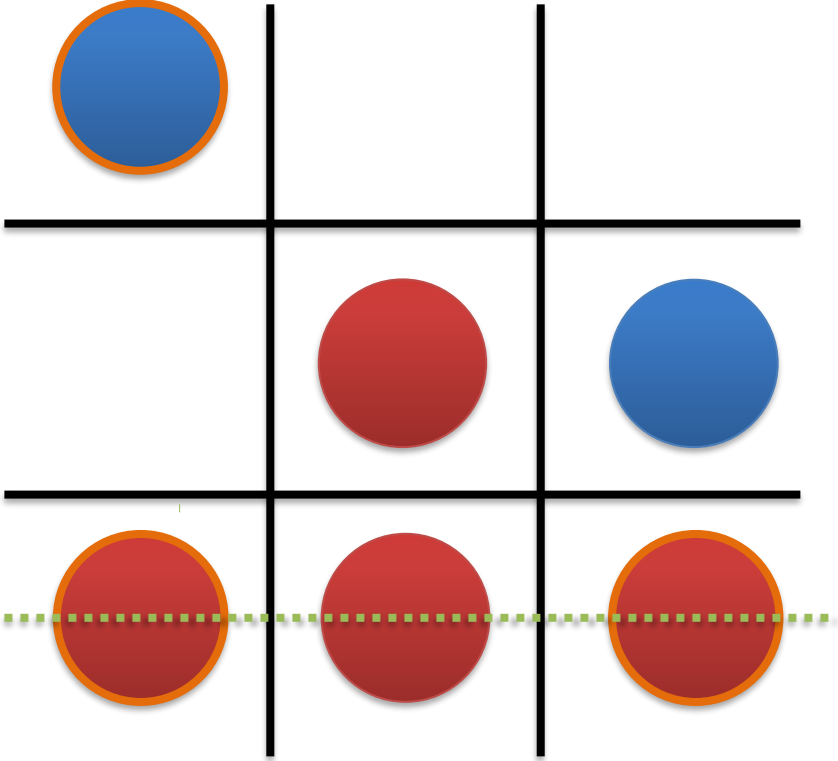
El objetivo es alinear tres fichas de un mismo color.

Los jugadores no tienen un color asignado.

En cada jugada deciden con cual juegan.

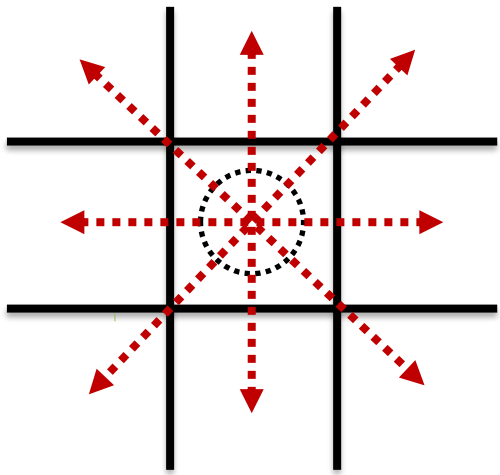
# Quien tiene un algoritmo, tiene un tesoro

Ta Te Ti Loco

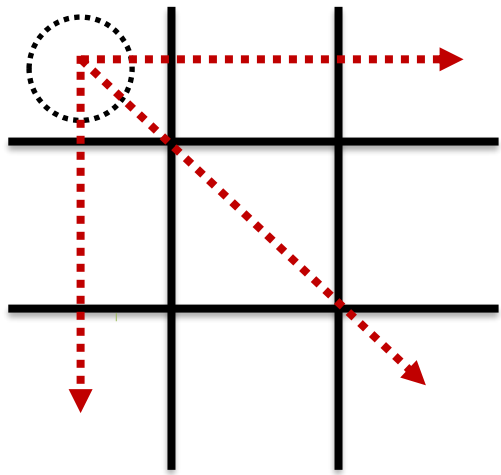


Buscar una estrategia ganadora

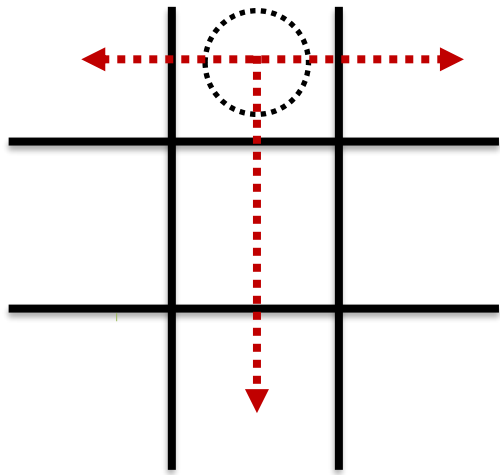
# Ta Te Ti Loco



4



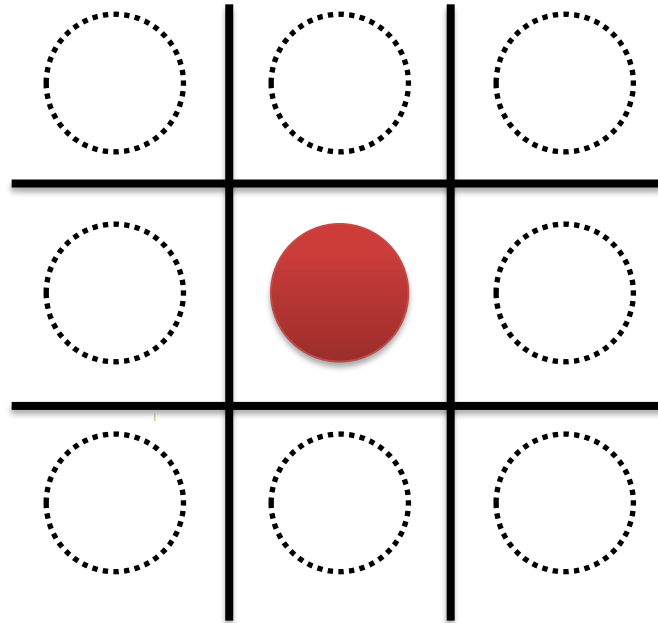
3



2

Buscar una estrategia ganadora

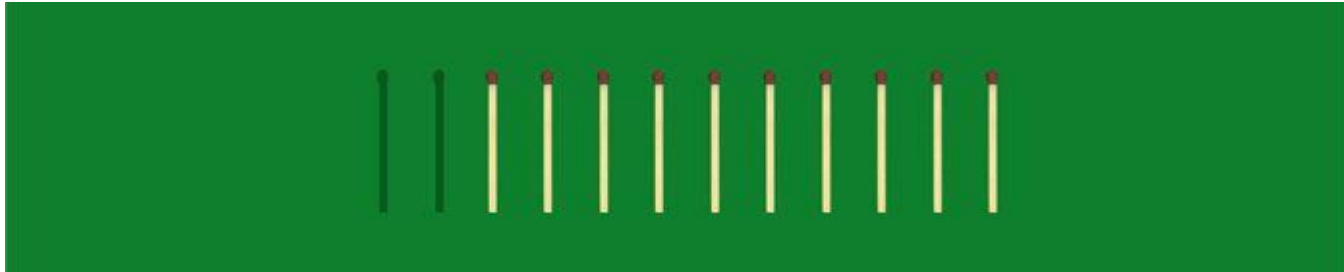
# Ta Te Ti Loco



Buscar una estrategia ganadora



Quien tiene un algoritmo, tiene un tesoro



**Autoconstrucció**

## NIM

- Se empieza con 7 fichas.
- Cada jugador , en su turno, puede coger 1 o 2 fichas.
- Gana el que deja la “mesa limpia”(el que coge la última ficha)



## NIM

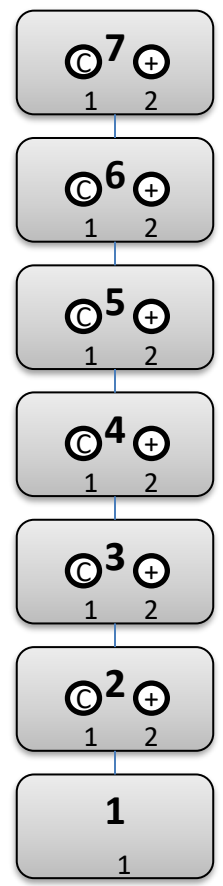
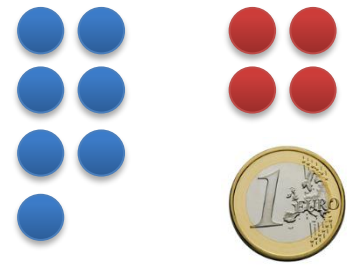
### La máquina que aprende

Grupo aconsejable: tres personas.

- Una persona gestiona la *máquina*.
- Una persona juega contra la *máquina*.
- Una persona controla que todo se haga bien.

#### Material

- Tablero
- 7 fichas de color (para jugar)
- 4 fichas de color diferente (control)
- Una moneda

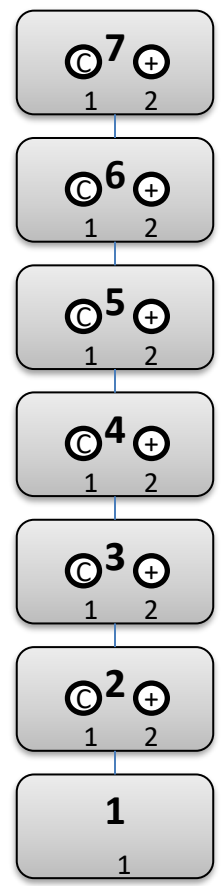
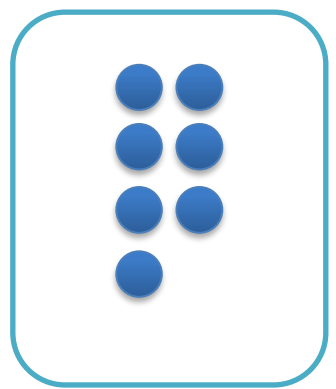




# NIM

## La máquina que aprende

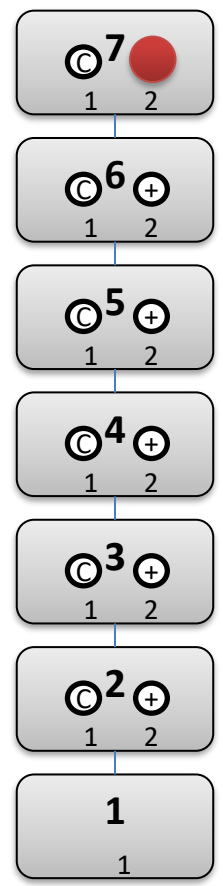
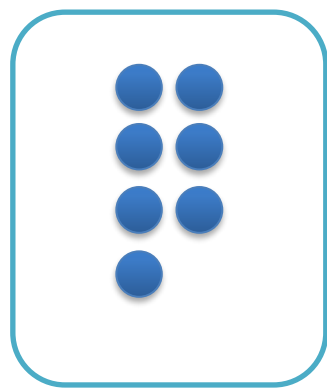
- Empieza el juego la *máquina* (casilla con el 7).
- Se tira la moneda.
- Si sale cara se retira una ficha del montón y si sale cruz se retiran dos.
- Se utiliza un marcador para indicar la jugada que se ha realizado (En el ejemplo una ficha de color rojo).



# NIM

## La máquina que aprende

- Imaginemos que sale cruz.
- Se marca cruz (2) en la casilla 7.
- Se retiran 2 fichas.
- Al jugador *humano* le quedan 5.

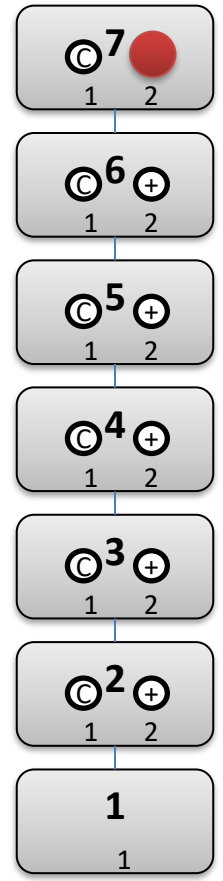
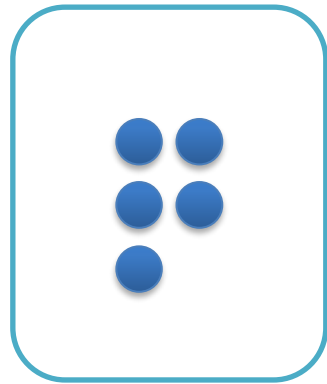


# NIM

## La máquina que aprende

A continuación juega el *humano*.

- Imaginemos que retira una ficha.
- En el montón de la mesa quedan cuatro fichas.

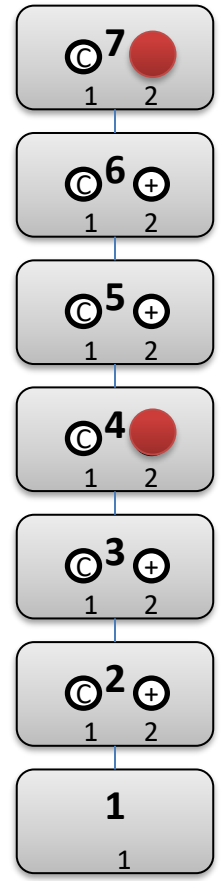
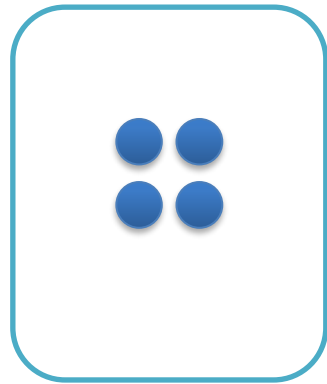


## NIM

### La máquina que aprende

Es el turno de la *máquina*.

- En la casilla del 4 tenemos dos opciones. Escogeremos una con la moneda.
- Imaginemos que vuelve a salir cruz.
- Marcaremos la jugada que ha salido y la realizaremos.
- En la casilla del 4 se marca que hemos retirado 2 fichas.
- Retiramos las dos fichas.
- Al *humano* le quedan dos fichas.

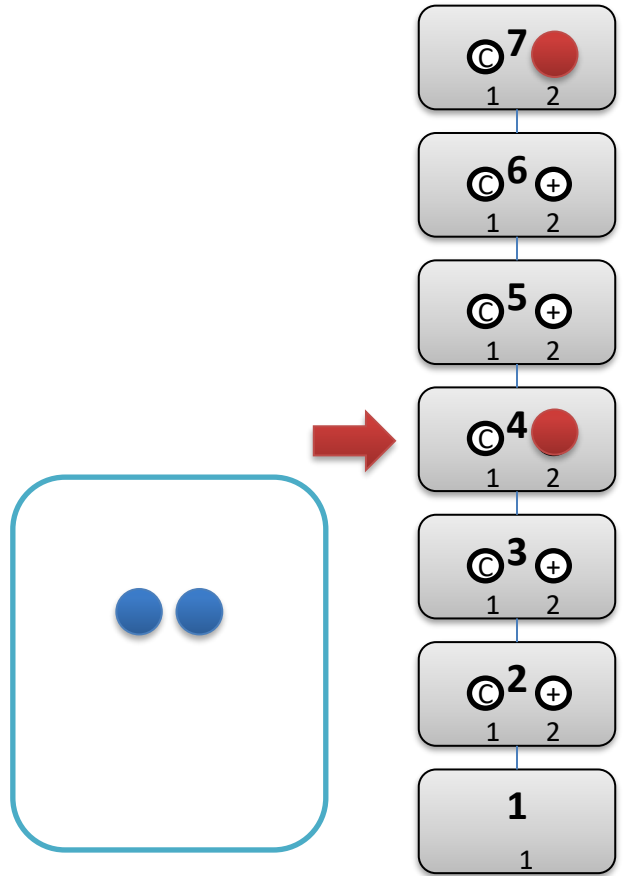


# NIM

## La máquina que aprende

Vuelve a jugar el *humano*.

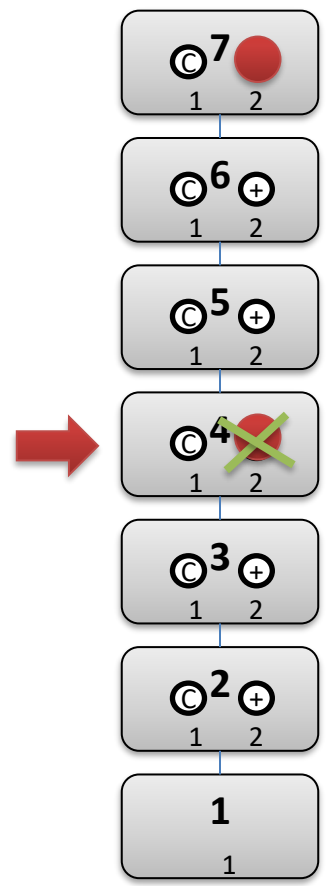
- Imaginemos que retira dos fichas y gana la partida.



# NIM

## La máquina que aprende

- En el caso de que la máquina pierda hemos de “castigar” la última jugada realizada.
- **En el ejemplo ha sido retirar dos fichas cuando quedaban 4.**
- Se sacan los marcadores i se tacha la jugada perdedora.
- **(Retirar 2 cuando hay 4 fichas).**
- Esta jugada no se volverá a realizar.
- **Si se vuelve a llegar a la casilla del 4 se escogerá la opción “retirar una ficha”.**

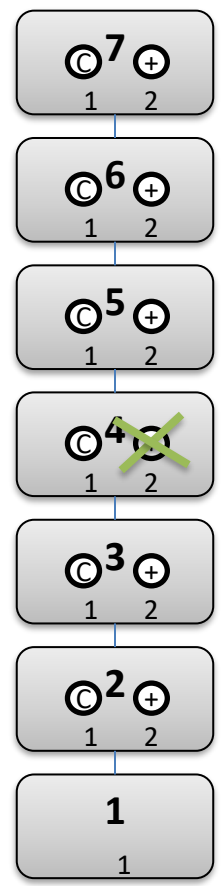


# NIM

## La máquina que aprende

A partir de aquí se van haciendo partidas nuevas.

Si la *máquina* gana no se elimina ninguna jugada. La *máquina* aprende sólo si pierde la partida.

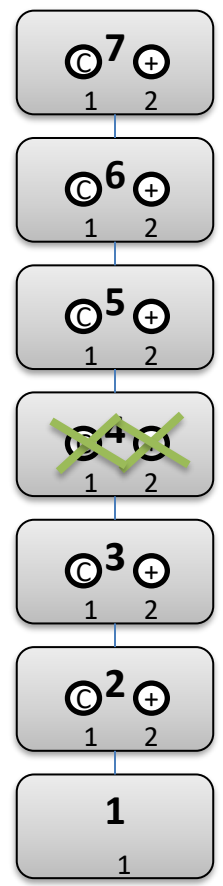


## NIM

### La máquina que aprende

#### Observaciones

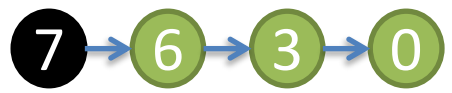
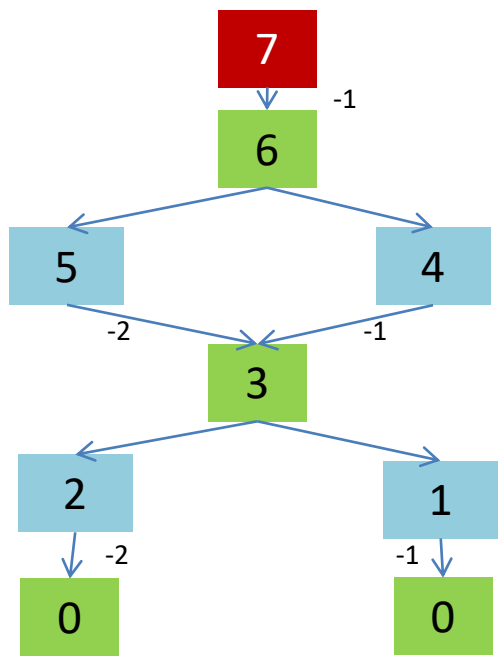
- Si en un momento dado hay una casilla (por ejemplo la del 4), que tiene las dos tiradas tachadas la *máquina se rinde* porque sabe que perderá. No es necesario continuar la partida. Se penaliza la jugada anterior.
- En la casilla del 1 la jugada es obligada: siempre se coge 1.
- La *máquina* no es inteligente. En la casilla del dos también juega al azar hasta que se elimine la jugada evidentemente perdedora: “retirar 1”



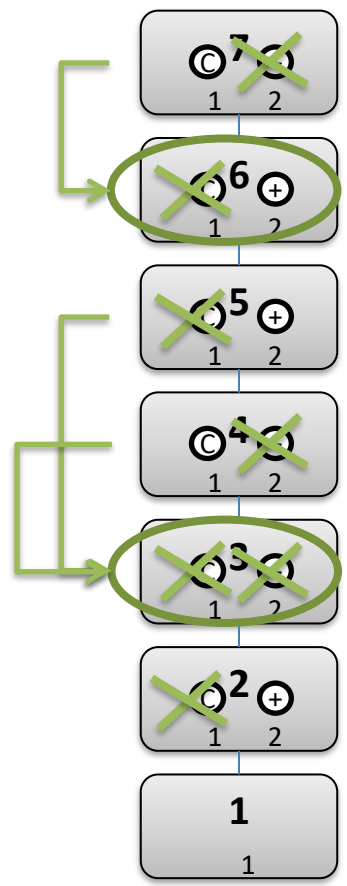


## NIM

### La máquina que aprende



- Retiramos una ficha al principio
- Si el otro jugador retira una ficha nosotros cogemos dos.
- Si el otro jugador retira dos nosotros cogemos una.



## NIM

### Modifiquemos el juego



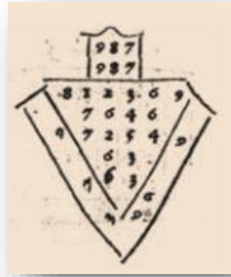
- ¿Qué pasa si comenzamos con 12 fichas?
- ¿Cómo varia la estrategia si cambiamos la cantidad total de fichas, por ejemplo 23?
- ¿Cómo seria la estrategia jugando con 32 fichas y pudiendo retorar 1, 2 o 3?
- ¿Y si pierde el que coge la última ficha?
- Etc.

# Quien tiene un algoritmo, tiene un tesoro

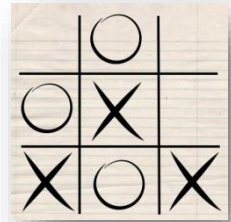
Aplicar



Interpretar-Justificar



Construir



Para saber más y materiales

- **Laberintos:**

[https://ja.cat/calaix\\_laberrints](https://ja.cat/calaix_laberrints)

- **Multiplicación egipcia:**

[https://ja.cat/calaix\\_multiplicacio\\_egipcia](https://ja.cat/calaix_multiplicacio_egipcia)

- **Ta-te-ti loco:**

[https://ja.cat/calaix\\_tateti\\_boig](https://ja.cat/calaix_tateti_boig)

- **Nim y “máquina que aprende”:**

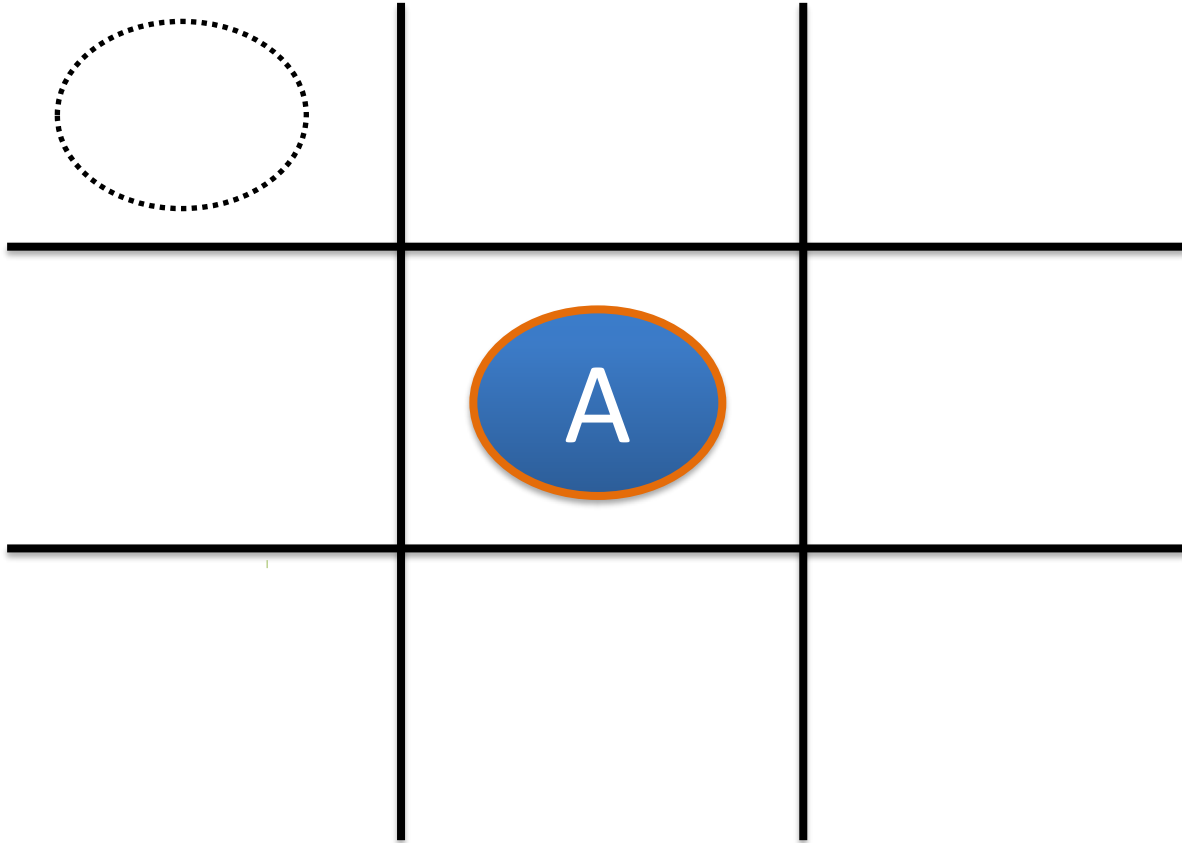
[https://ja.cat/arc\\_cesire\\_maquina\\_nim](https://ja.cat/arc_cesire_maquina_nim)

Encontraréis la presentación i el material fotocopiable en

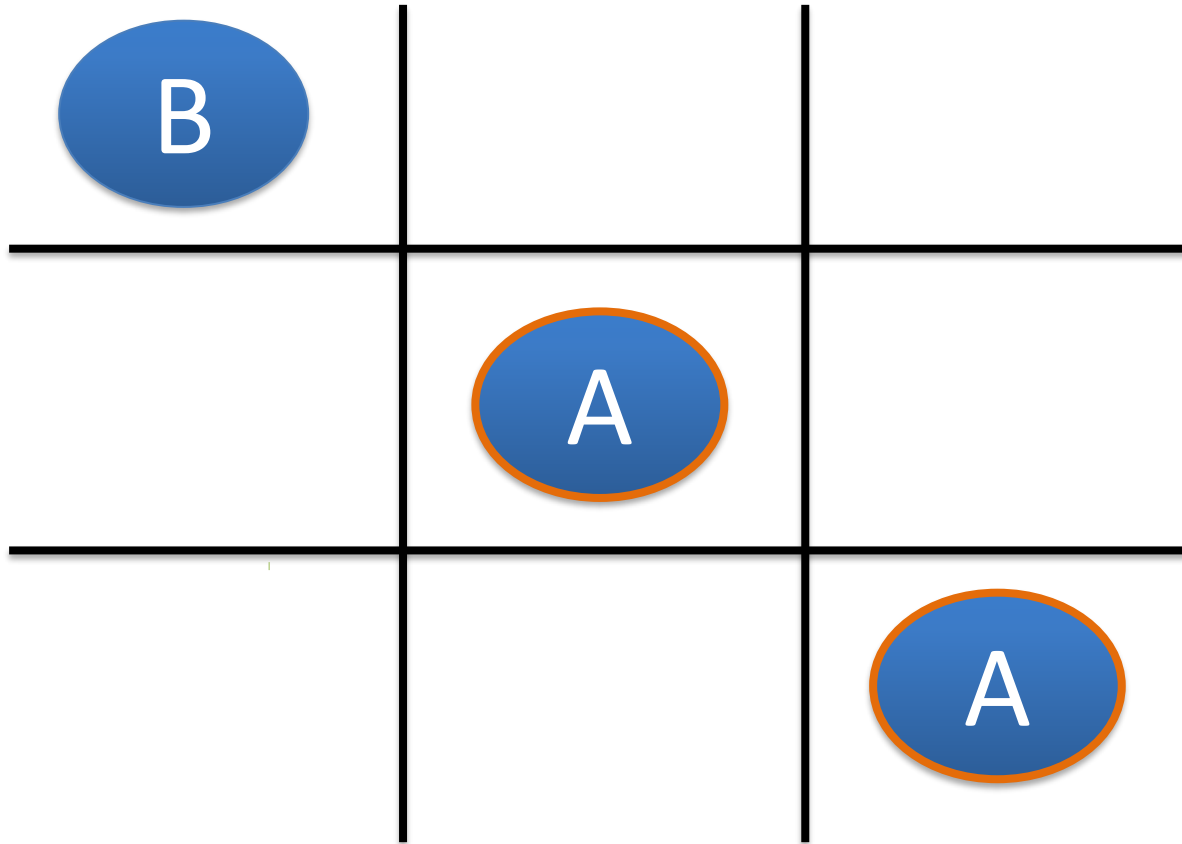
**<https://ja.cat/jaem-algoritmos>**



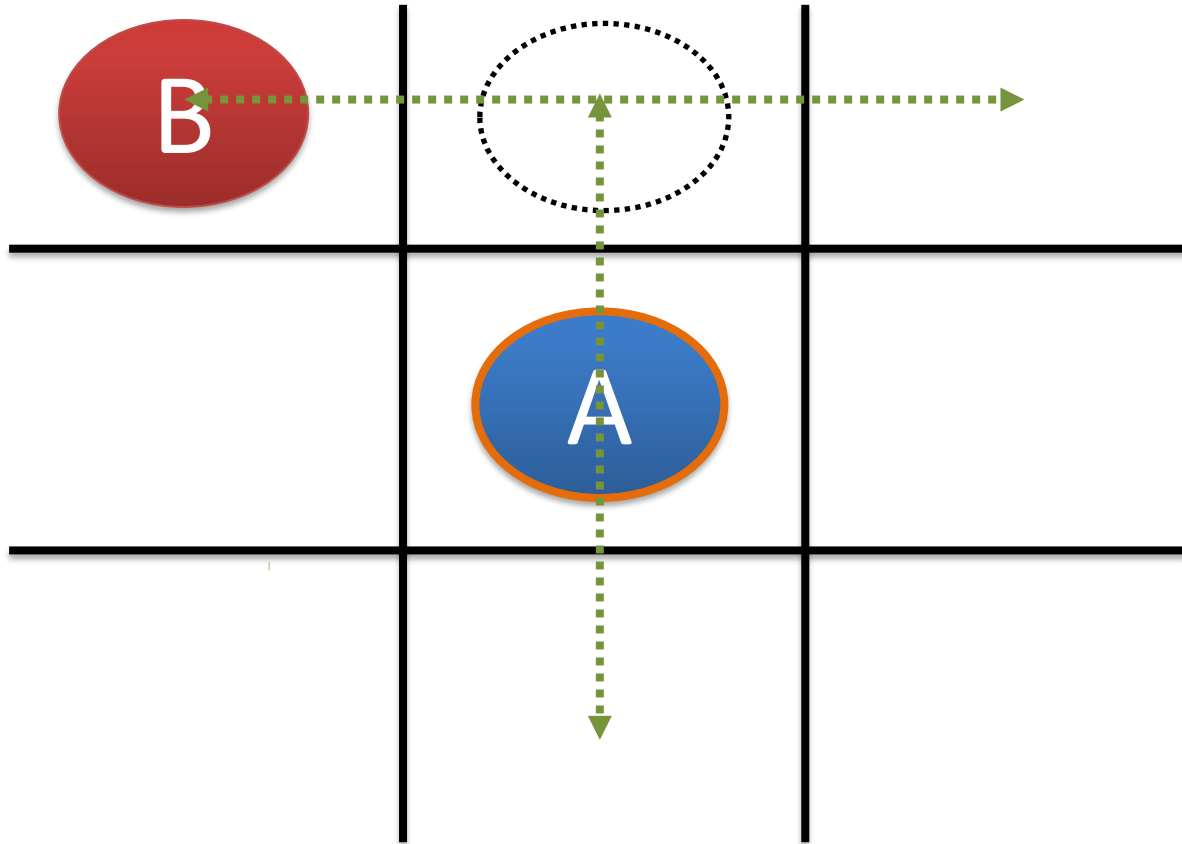
# Ta Te Ti Boig



# Ta Te Ti Boig

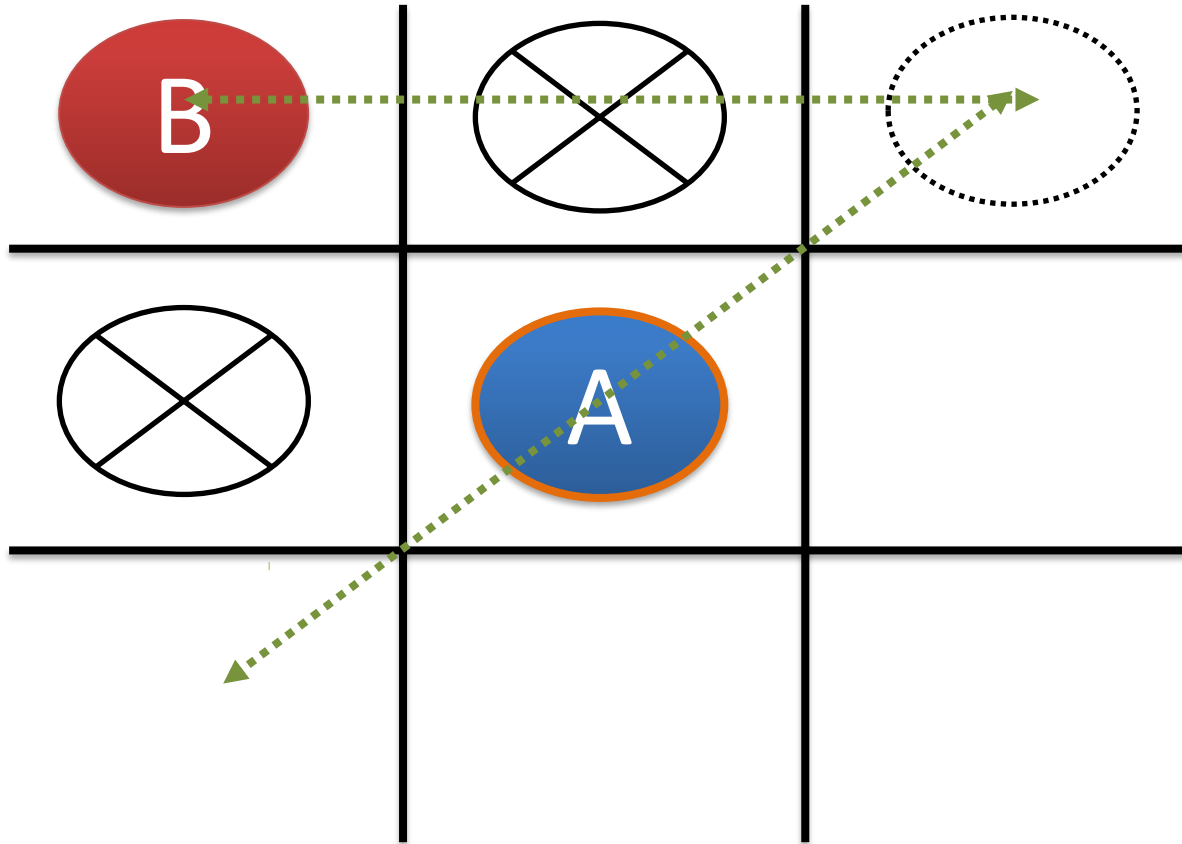


# Ta Te Ti Boig

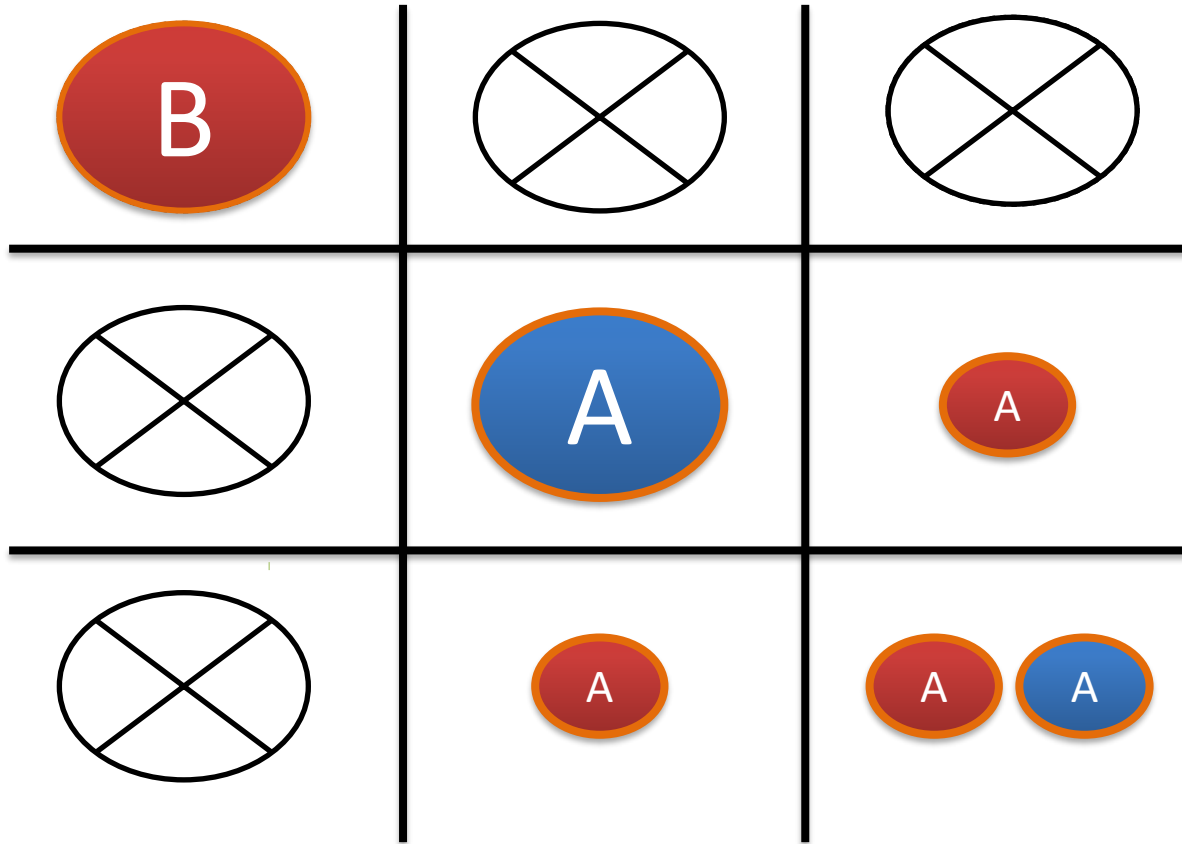




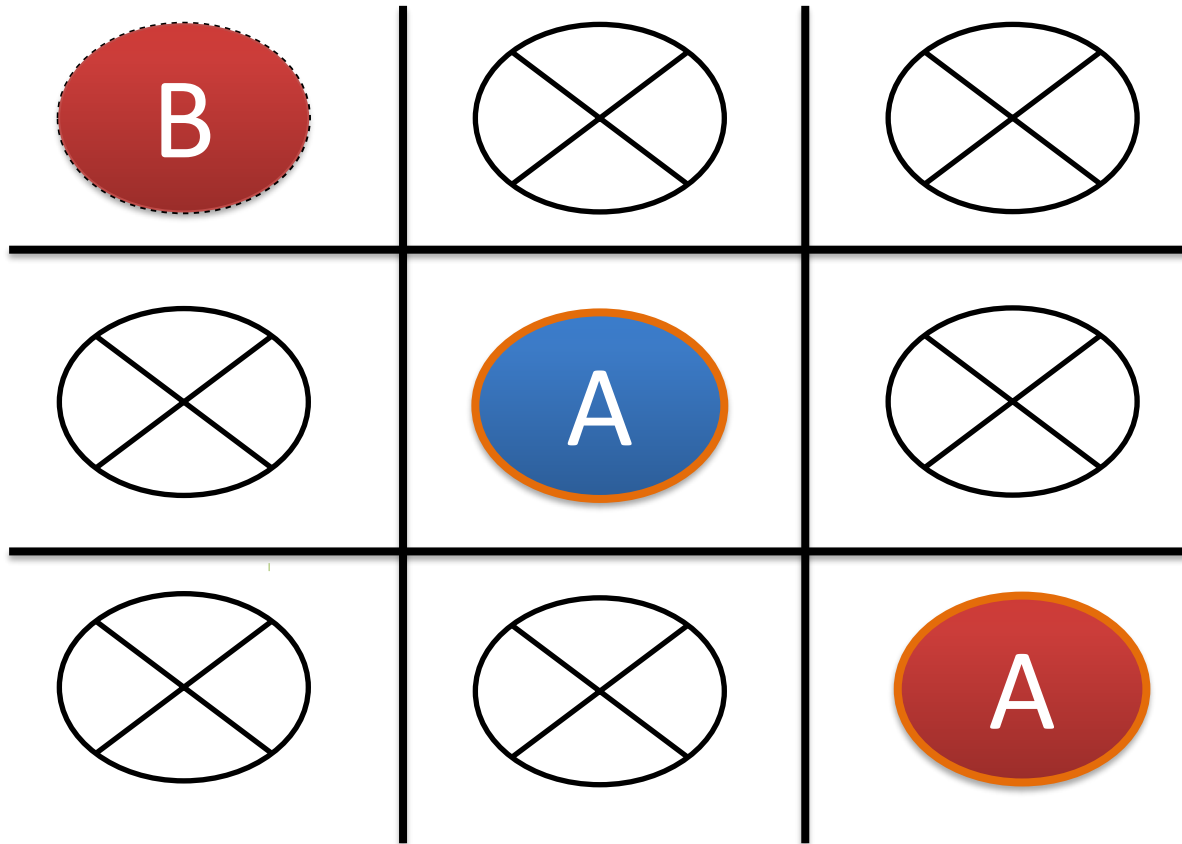
# Ta Te Ti Boig



# Ta Te Ti Boig

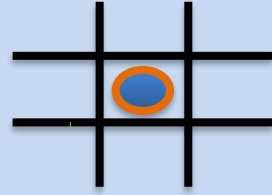


# Ta Te Ti Boig

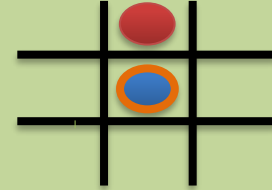
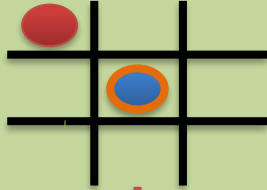


# Ta Te Ti Boig

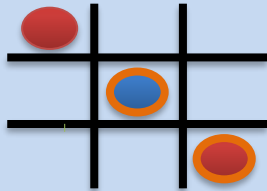
A



B

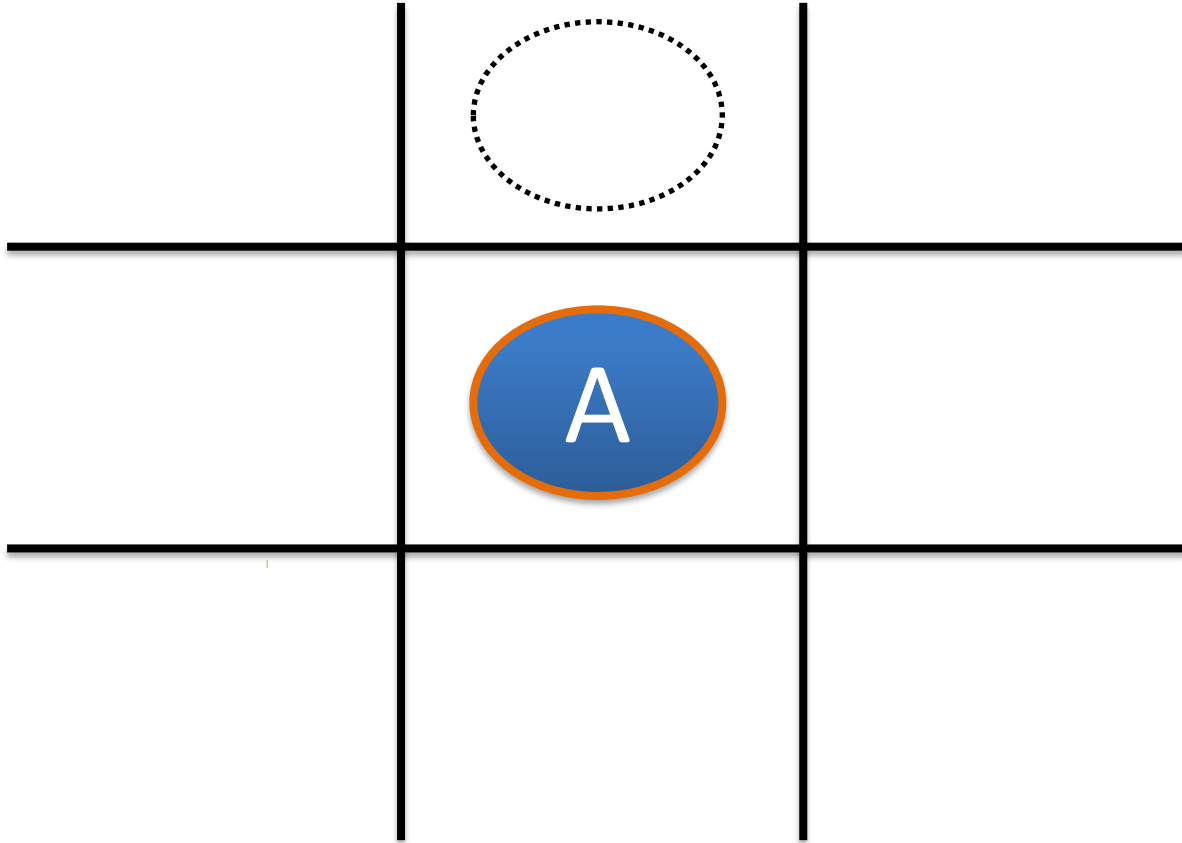


A

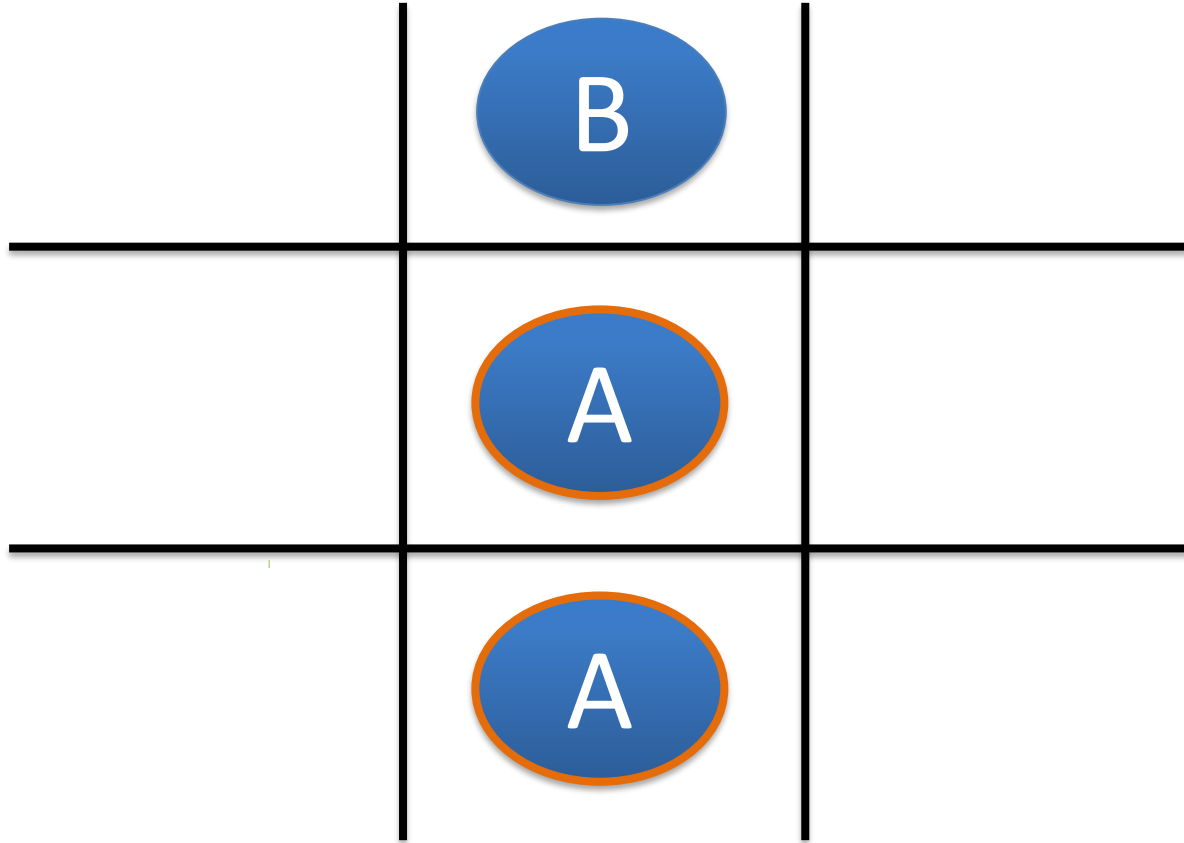


Fer 3 tres en ratlla  
segons resposta

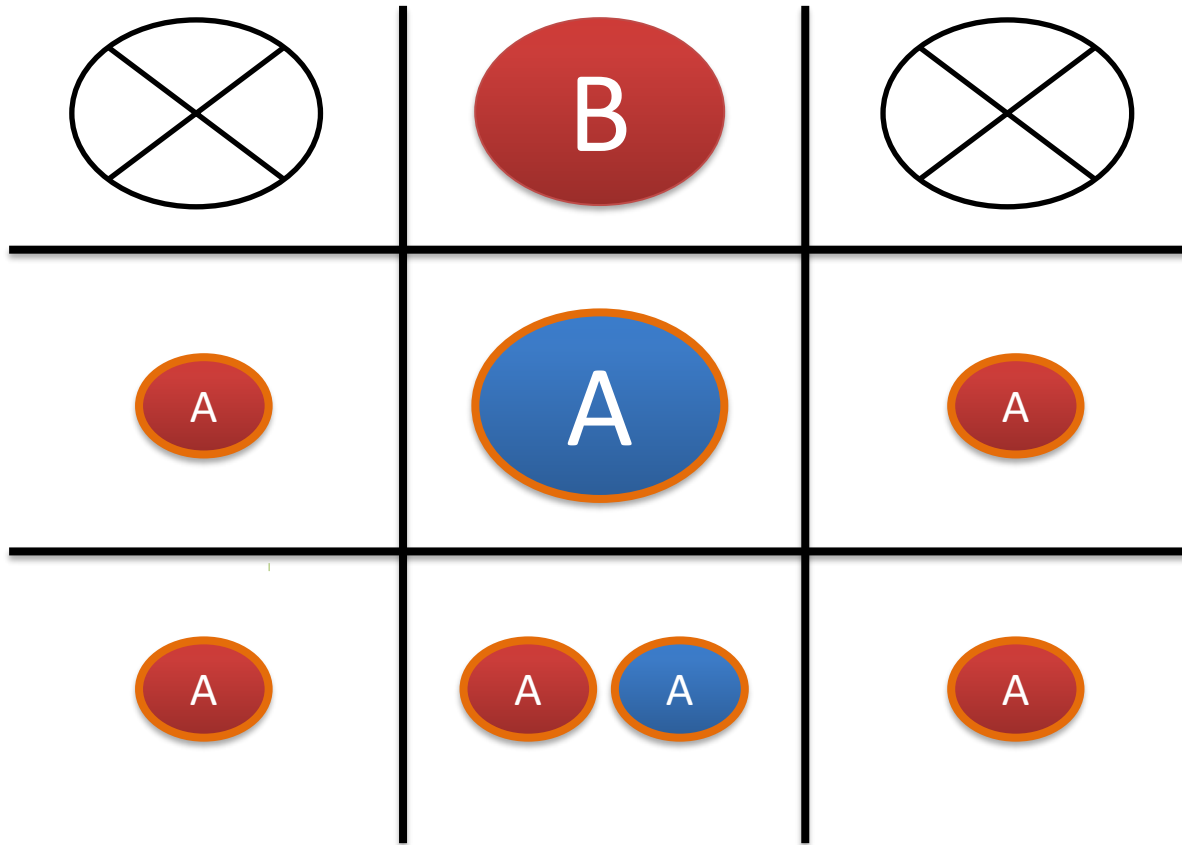
# Ta Te Ti Boig



# Ta Te Ti Boig

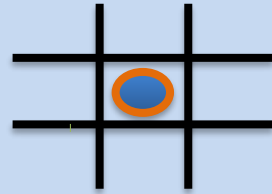


# Ta Te Ti Boig

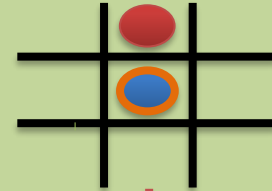
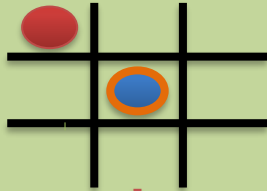


# Ta Te Ti Boig

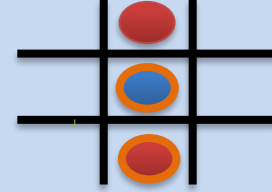
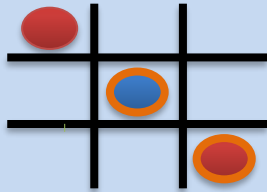
A



B



A



Fer 3 tres en ratlla  
segons resposta

?

