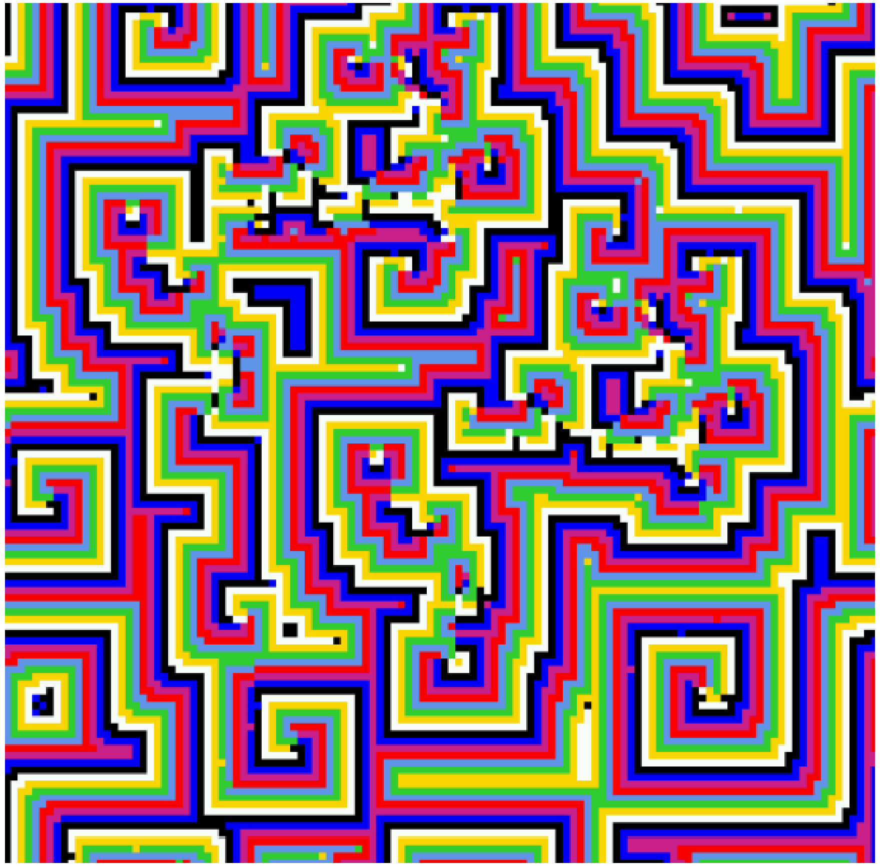


Autómatas Celulares

Ismael Acosta
(iacosta@fcien.edu.uy)
Facultad de Ciencias, UdelaR

Contenido de la clase

- Introducción
- Autómata de Conway: Game of Life
- Autómata de Wolfram

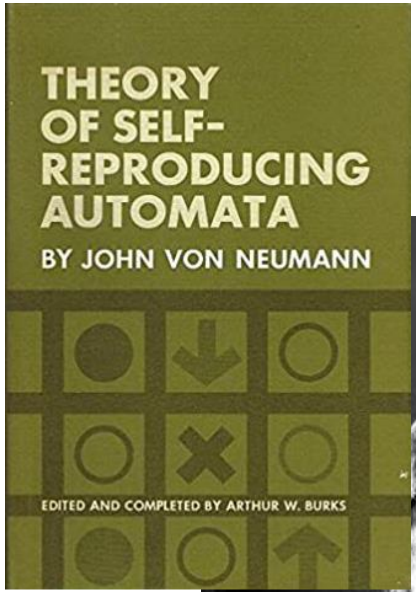
Contenido de la clase

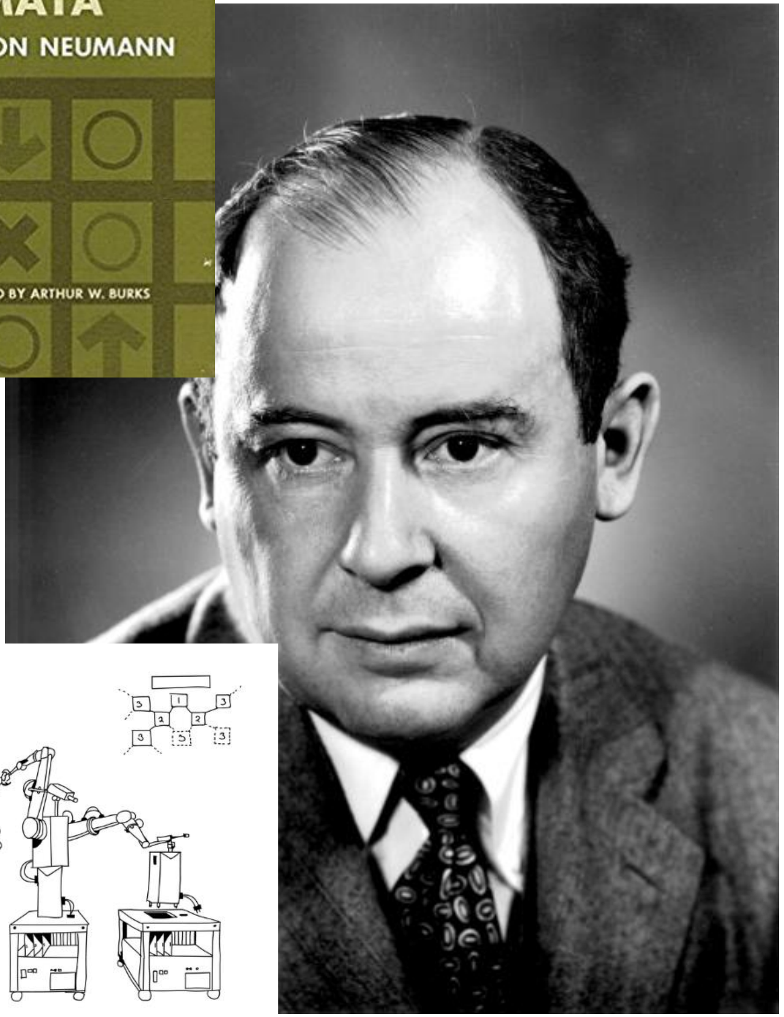
- **Introducción**
- Autómata de Conway: Game of Life
- Autómata de Wolfram

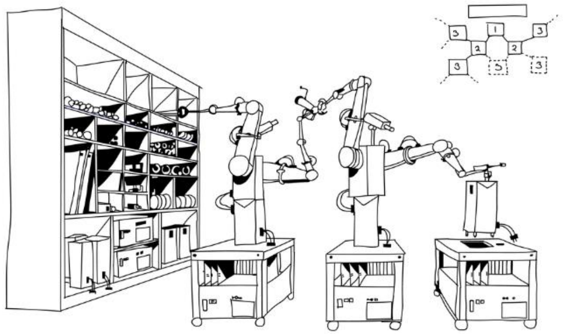
El concepto original de "*autómata celular*" surgió con John von Neumann en la década de los 40s.

Teóricamente, consistía en una máquina capaz de **autoreplicarse** y realizar cálculos complejos.

Estas ideas serían las precursoras de los ordenadores modernos.

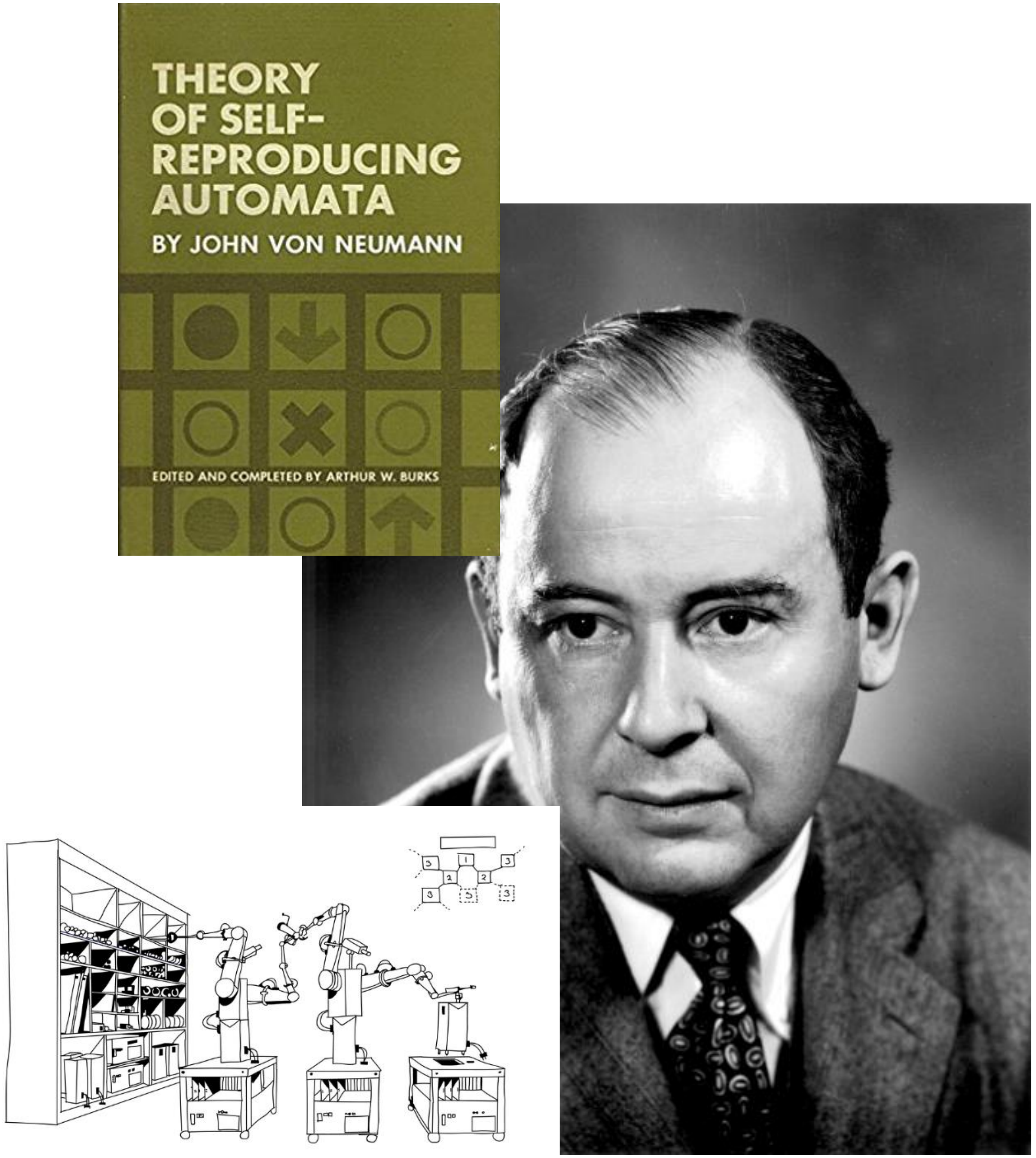






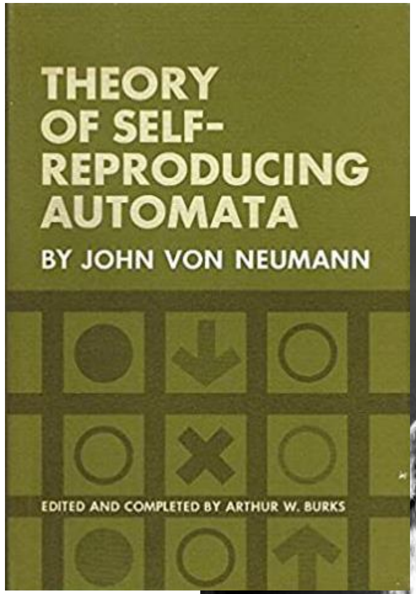
Por definición, un autómata celular es un **modelo matemático-computacional de un sistema dinámico** que evoluciona en *pasos discretos*.

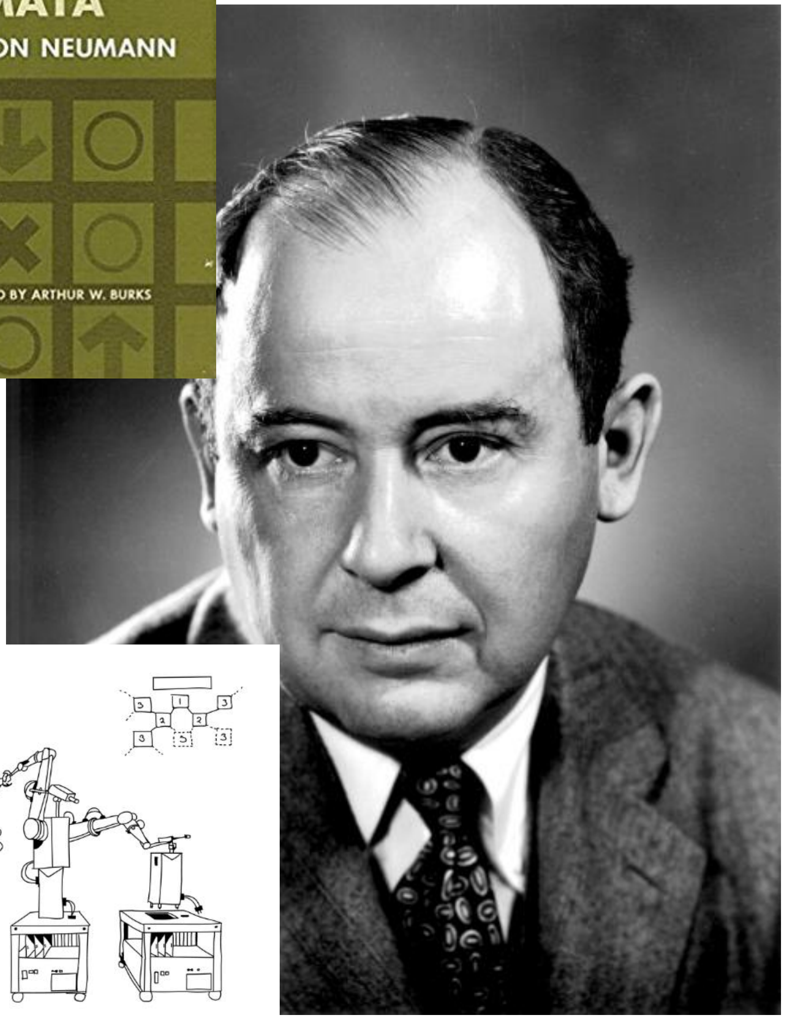
Está definido por **celdas** que interactúan entre sí en el tiempo y pueden cambiar de estado.

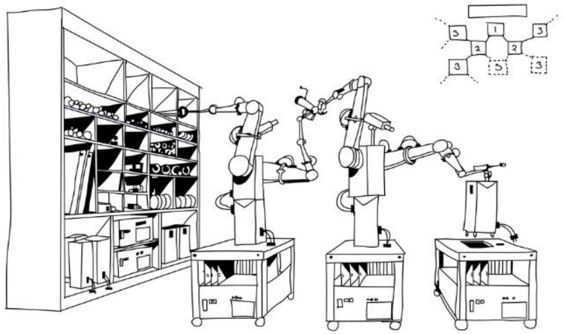


Todos los autómatas celulares poseen al menos 5 elementos básicos:

1) **Espacio.**
2) **Estados.**
3) **Configuración inicial.**
4) **Vecindad.**
5) **Reglas.**

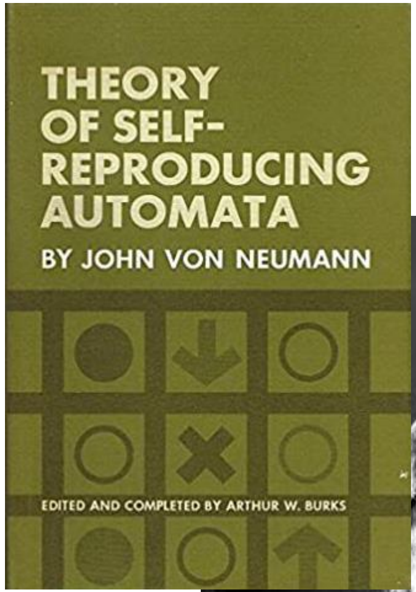


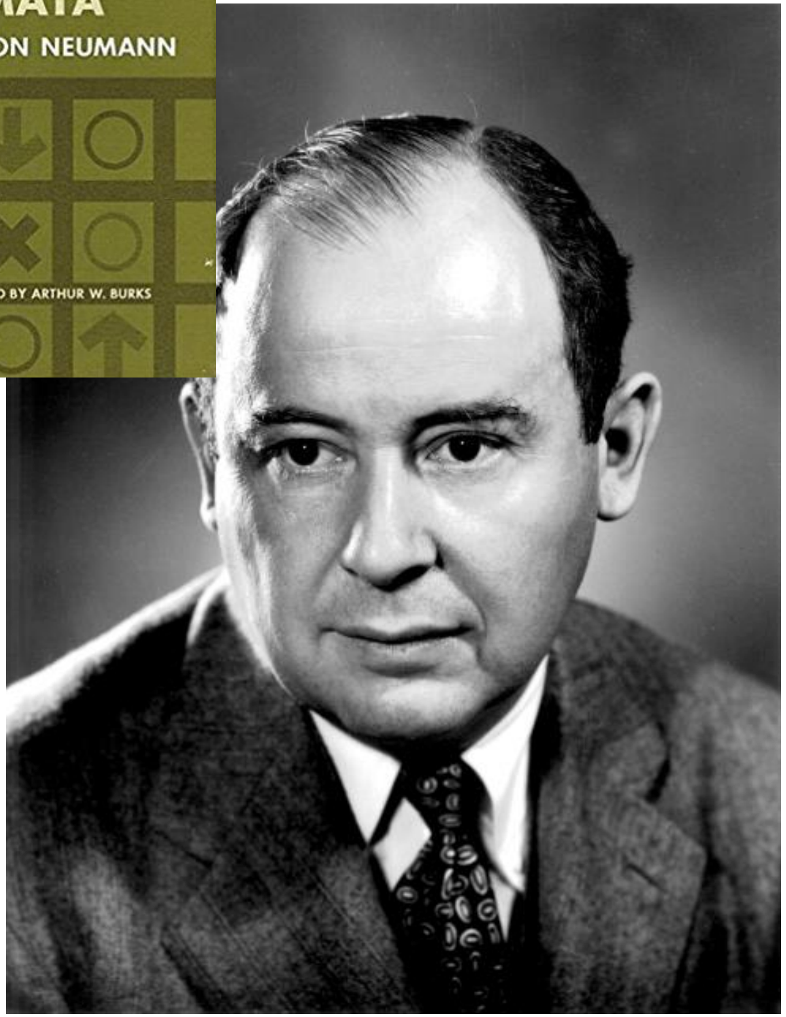




El **espacio** de un autómata celular es cualquier región n-dimensional, dividida en *células* (o celdas), generalmente homogéneas, lo que constituye una *grilla* (o *lattice*).

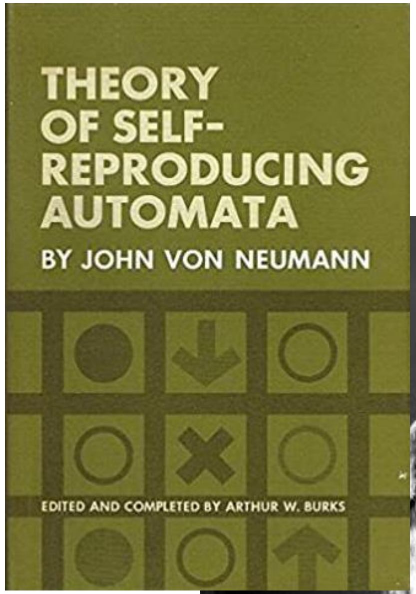
Cada celda tiene un conjunto de **estados posibles** (finito) y solo puede tomar un estado por iteración. Puede ser representado por valores numéricos, colores, letras, etc.

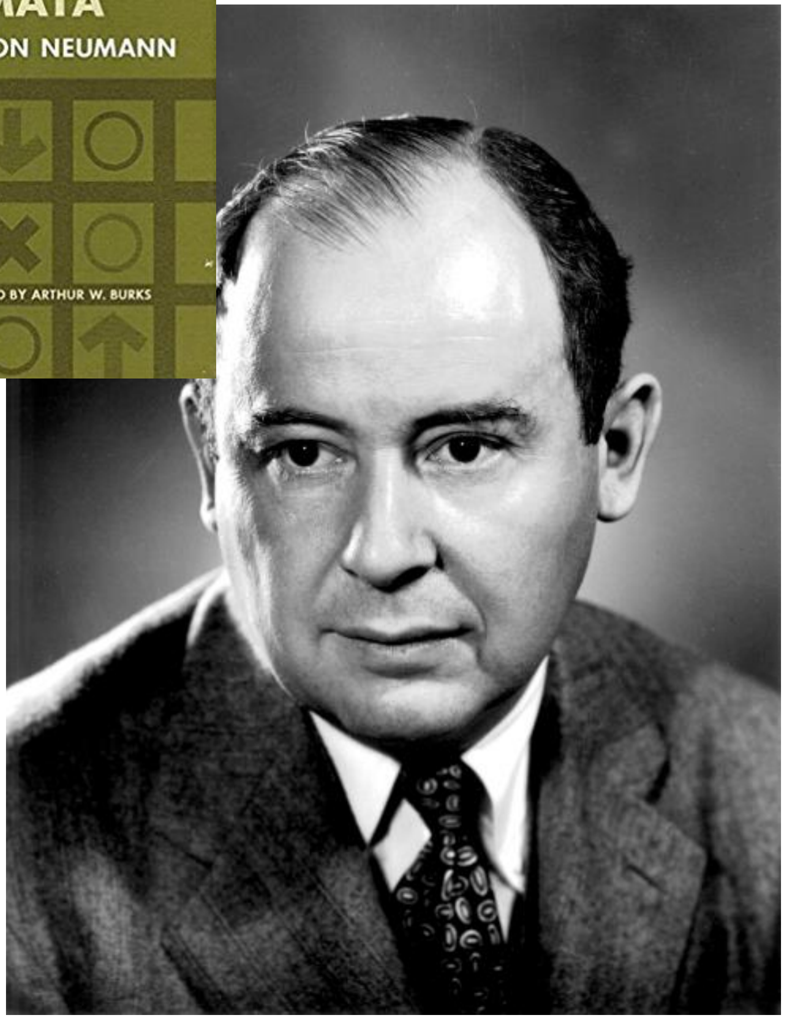




La **vecindad** de un autómata se define como el conjunto de celdas que se consideran adyacentes a una celda dada, así como su posición/estado relativa a ella.

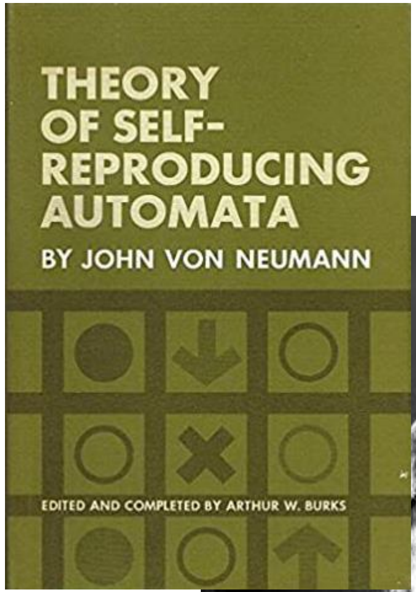
Este concepto es relativo a cada autómata.

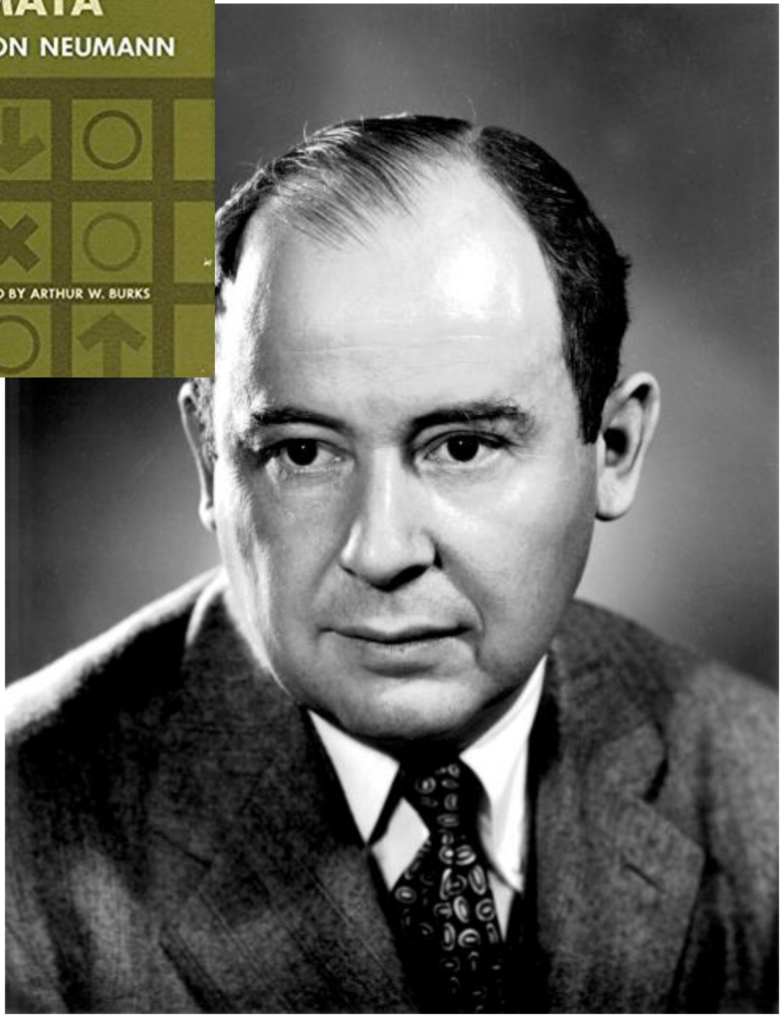




Cada celda evoluciona a pasos discretos de tiempo de acuerdo a un **conjunto de reglas** (*funciones*) que definen el comportamiento del autómata.

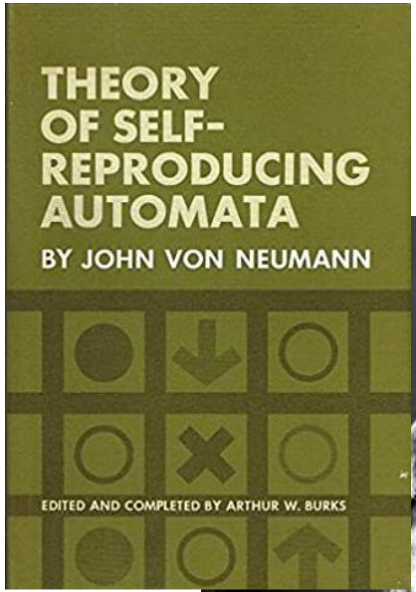
A partir del estado de la celda y su vecindad se calcula el estado de la celda en la siguiente iteración.

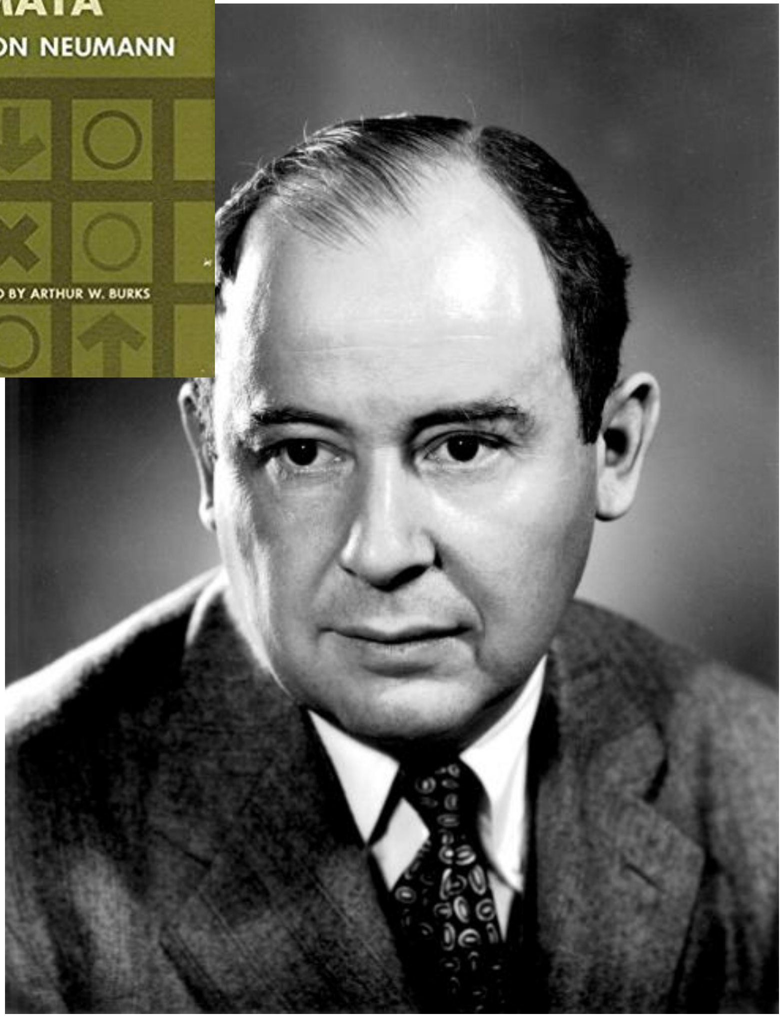




Los autómatas celulares más comunes (i.e.: los que trabajaremos en este práctico) decimos que son *triplemente discretos* porque:

1) Evolucionan en intervalos de tiempo discreto.
2) El conjunto de estados posibles es discreto.
3) El espacio de celdas es discreto.





Contenido de la clase

- Introducción
- Autómata de Conway: Game of Life
- Autómata de Wolfram

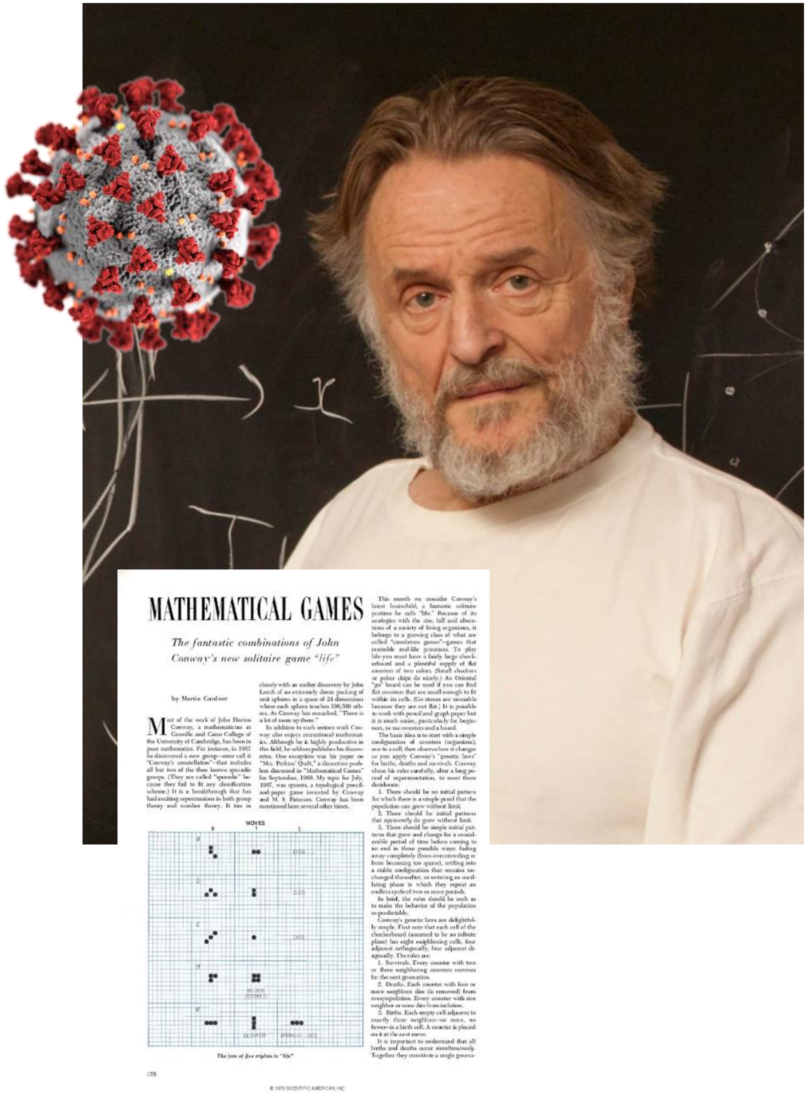
Contenido de la clase

- Introducción
- **Autómata de Conway: Game of Life**
- Autómata de Wolfram

En 1970, John H. Conway publica su autómata celular conocido como ***Game of Life*** (‘*El Juego de la Vida*’)[1,2].

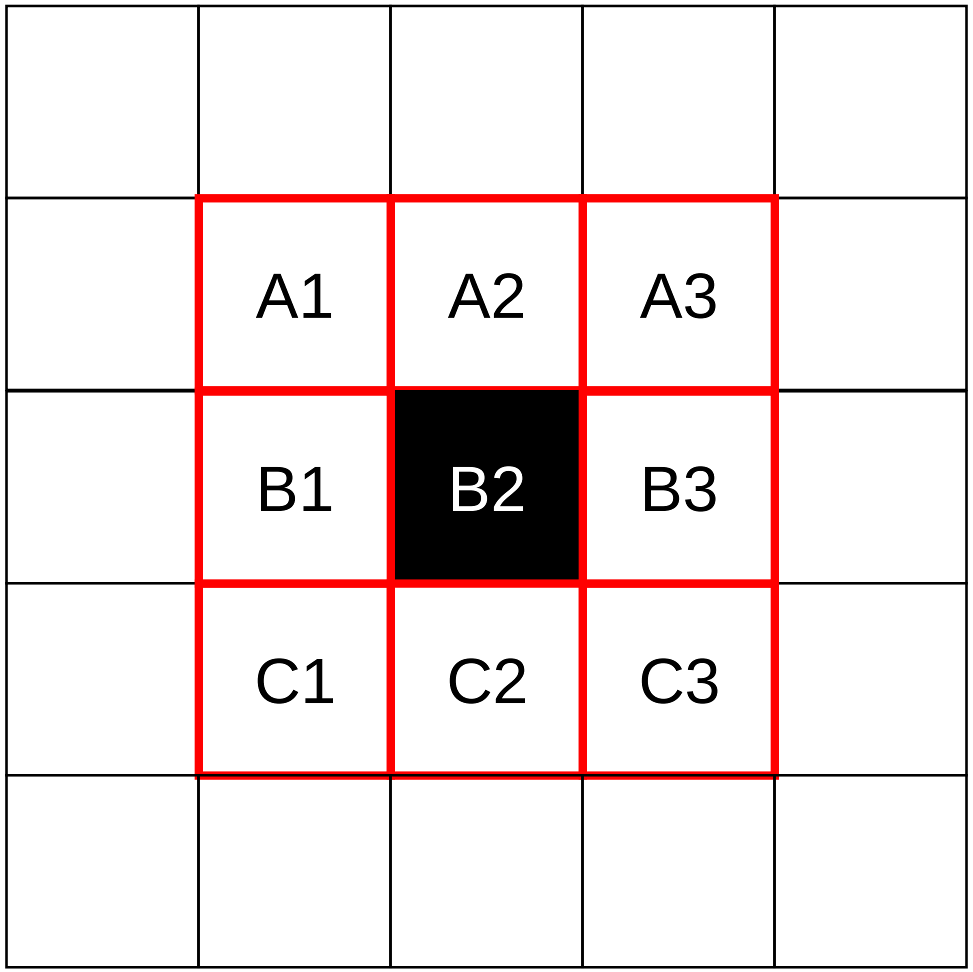
Consiste en una grilla **bidimensional**, formadas por celdas capaces de tener 2 estados (**viva o muerta**).

La evolución de cada celda depende de su propio estado y de los estados de su vecindad *inmediatamente* adyacente.



[1] Gardner, M. (1970). Mathematical Games: The fantastic combinations of John Conway’s new solitaire game ’life’. *Scientific American*.
[2] Bhargava, M. (2020). John Horton Conway (1937-2020). *Nature*, *582*(7810), 27-28.

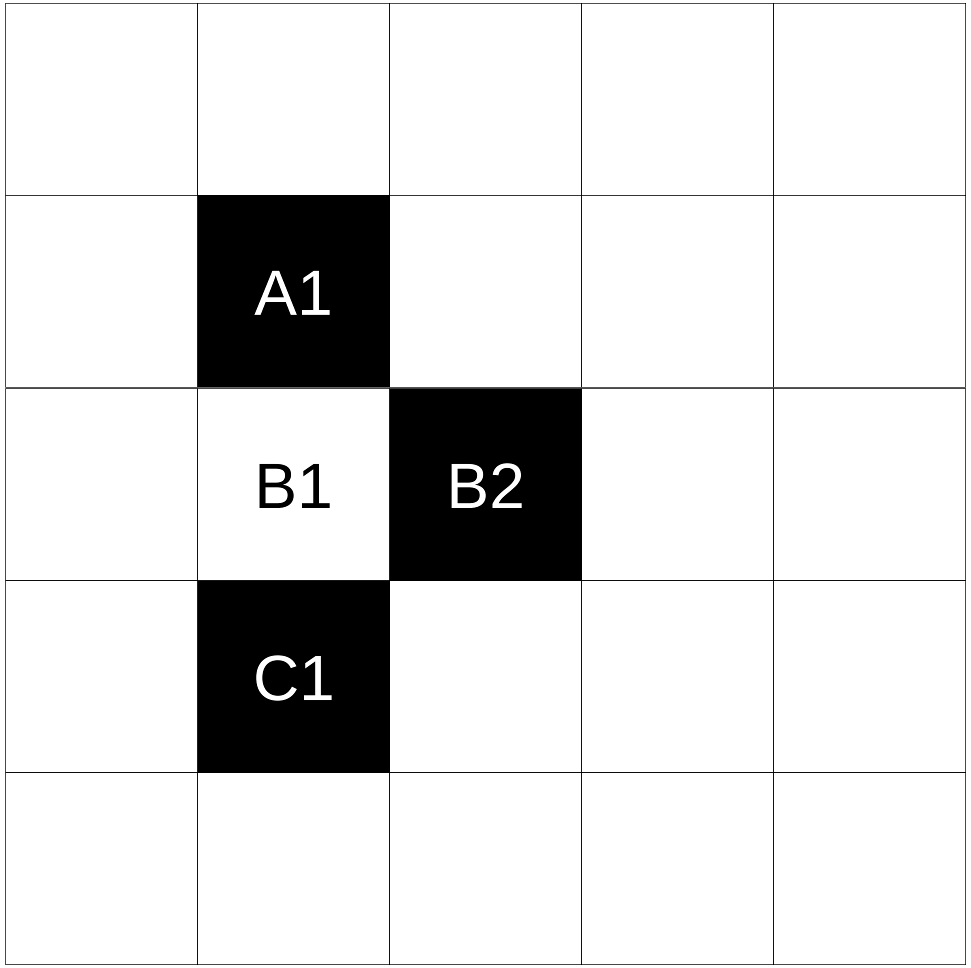
A partir de un espacio bidimensional infinito (un plano), la evolución de una celda en tiempo $t + 1$ depende de su propio estado en tiempo t y de los estados de la vecindad adyacente (**Vecindad de Moore**), es decir, de las 8 celdas con las que comparte una arista o un vértice (celdas bordeadas en rojo).



Regla de Nacimiento

Una celda pasará de estado muerto (blanco) a vivo (negro) en tiempo $t + 1$, si en tiempo t está rodeada por 3 celdas vivas.

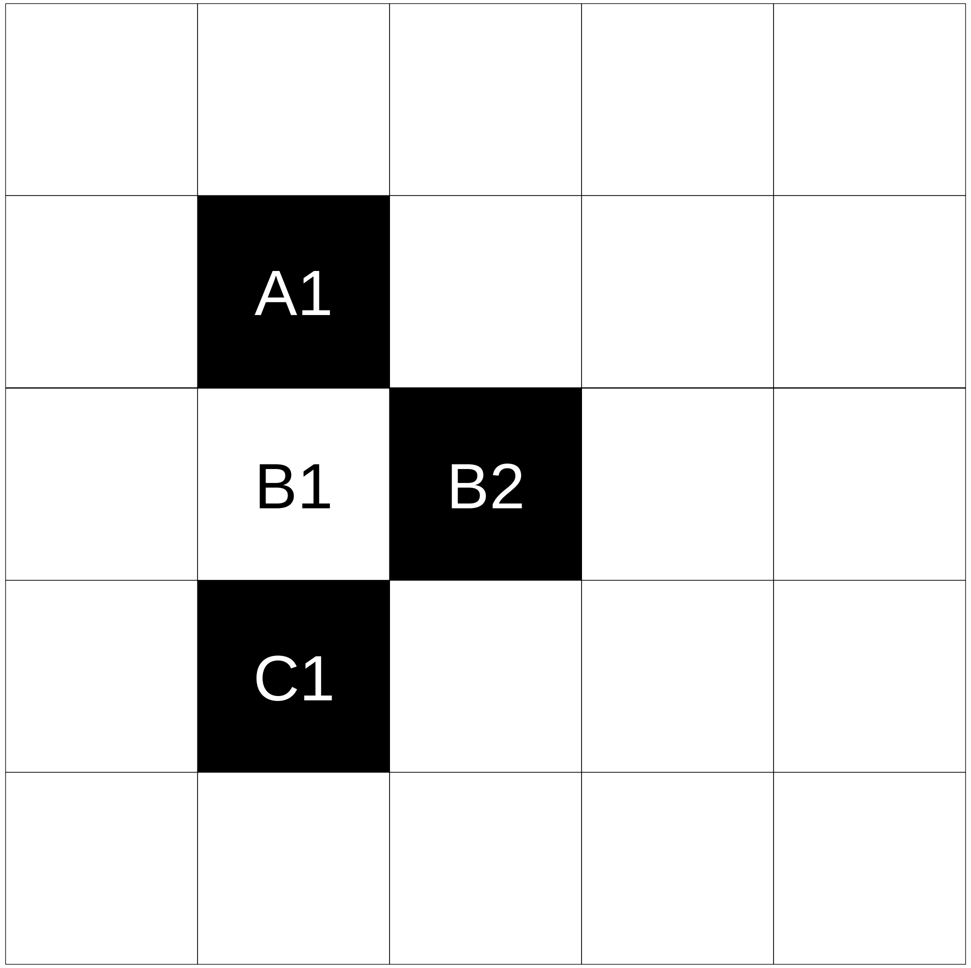
En este ejemplo, la celda B1 estará viva (negro) en tiempo $t + 1$ por tener tres vecinas vivas (negras) en tiempo t.



Regla de Nacimiento

Podemos interpretar esta regla como la necesidad de tener una población mínima para que surja (nazca) una celda.

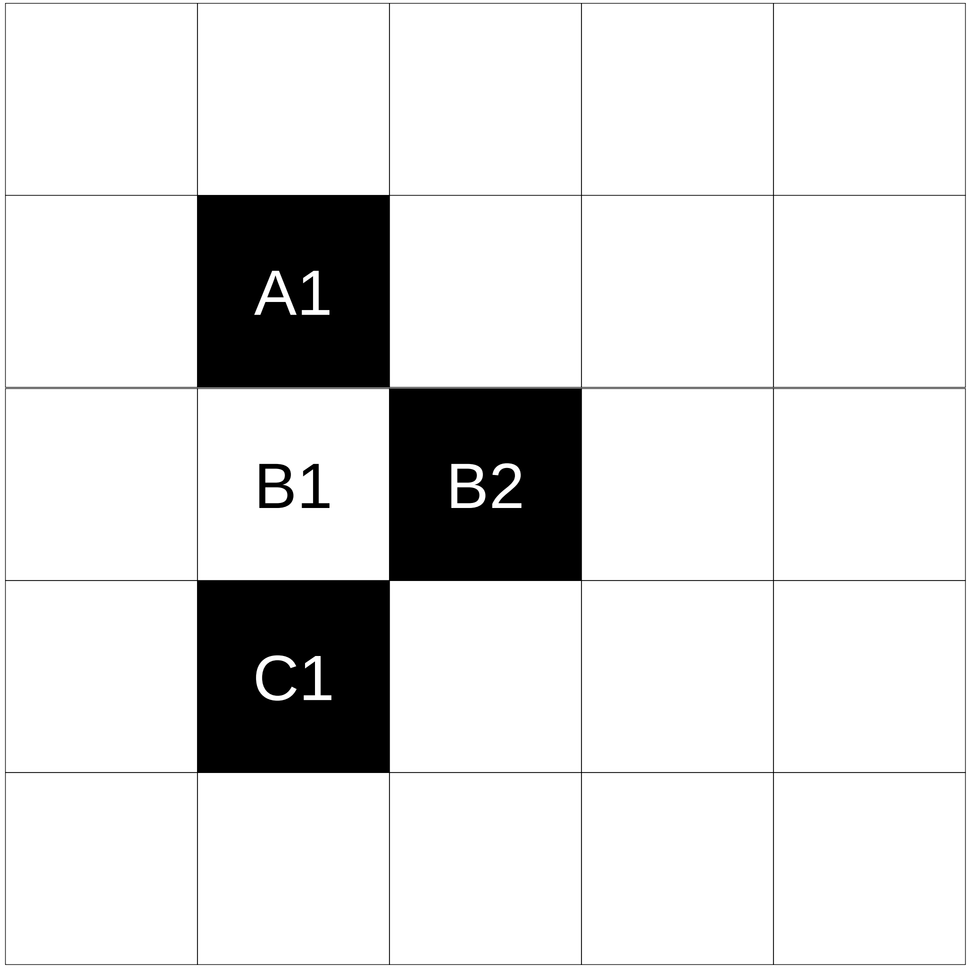
Quitando las celdas de la configuración inicial (impuestas), no existe la generación espontánea de celdas vivas rodeadas de celdas muertas.



Regla de Muerte

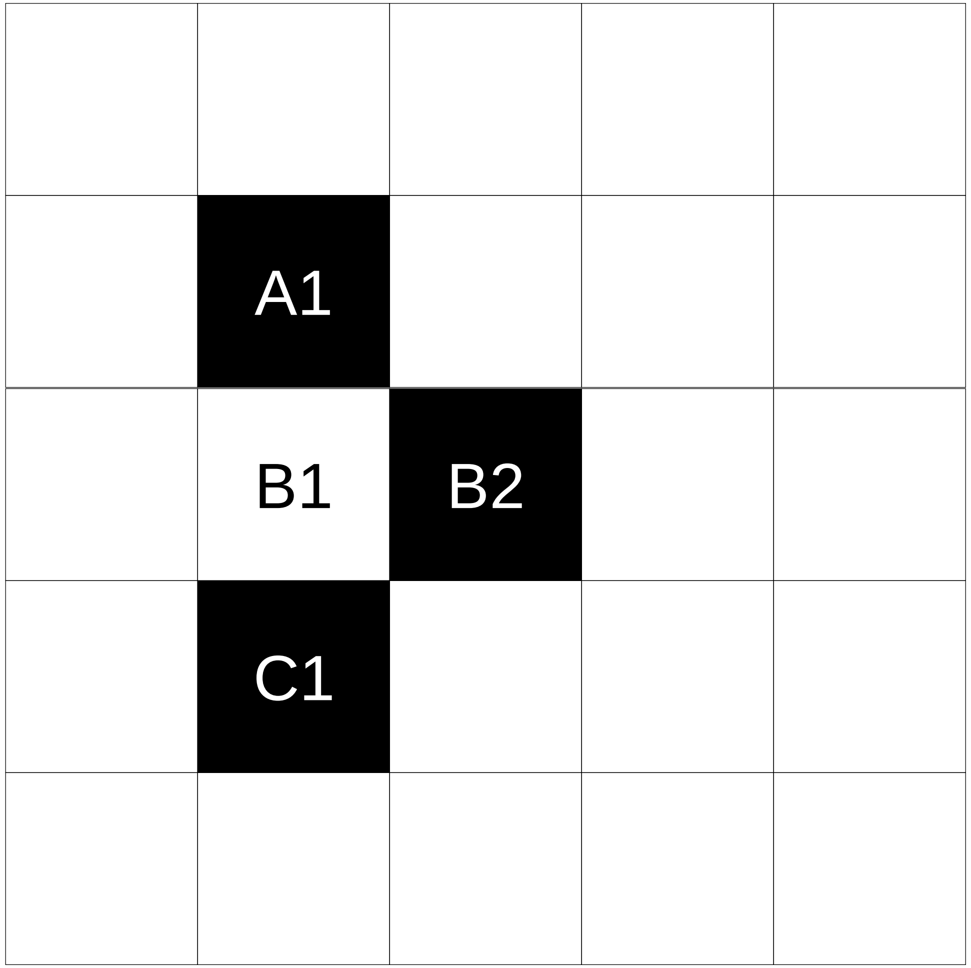
Una celda pasará de estado vivo (negro) a muerto (blanco) si está rodeada por 1 o menos celdas vivas o más de 3 celdas vivas.

Podemos interpretar esta regla como **muerte por aislamiento** (≤ 1) o **muerte por sobrepoblación** (> 3).



Regla de Muerte

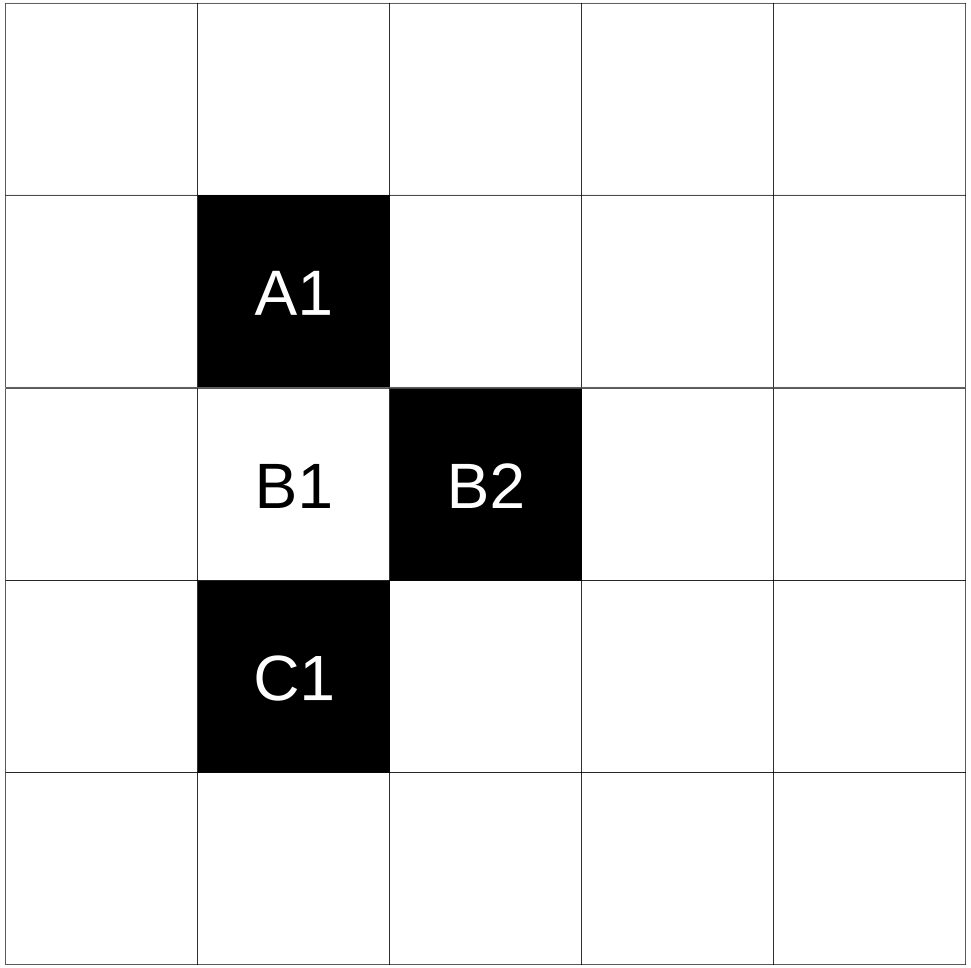
En este ejemplo, las celdas A1 y C1 que están vivas (negras) en tiempo t, pasarán a estar muertas (blancas) en tiempo $t + 1$, ya que solamente tienen una única celda viva adyacente (celda B2).



Regla de Supervivencia

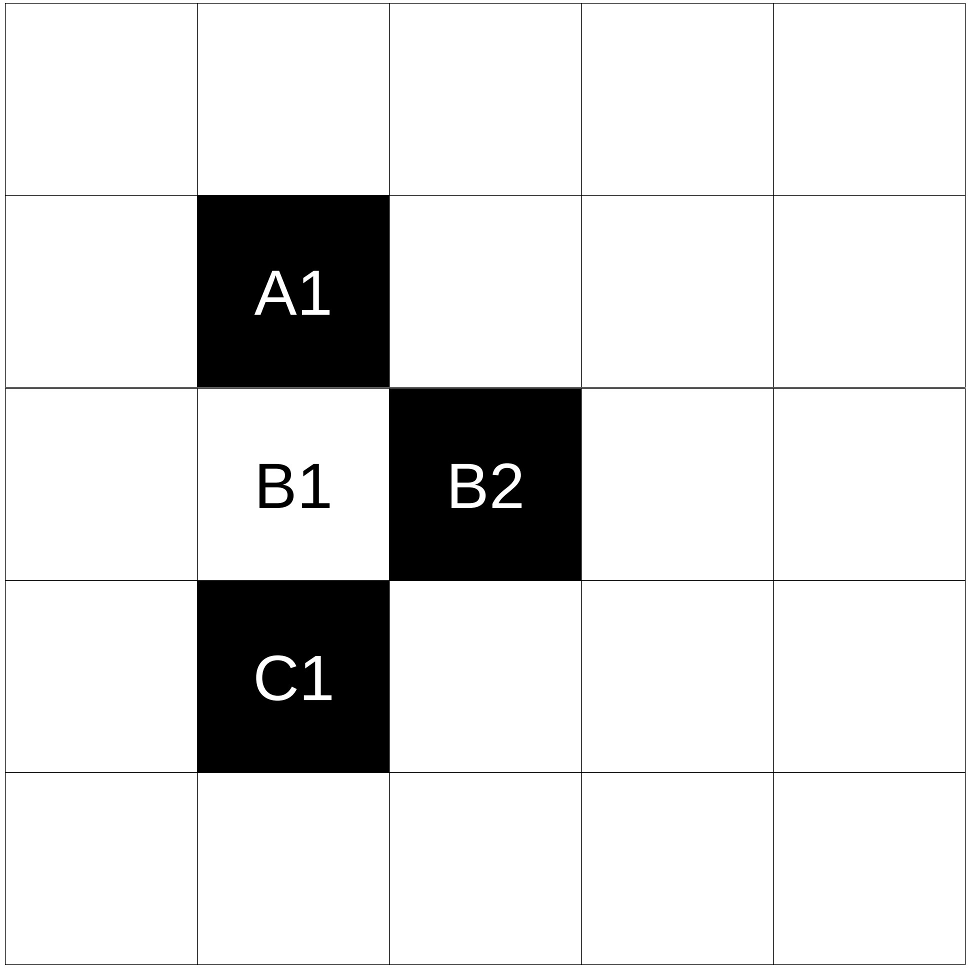
Una celda viva permanecerá en su estado vivo (negro) si tiene 2 o 3 vecinos vivos.

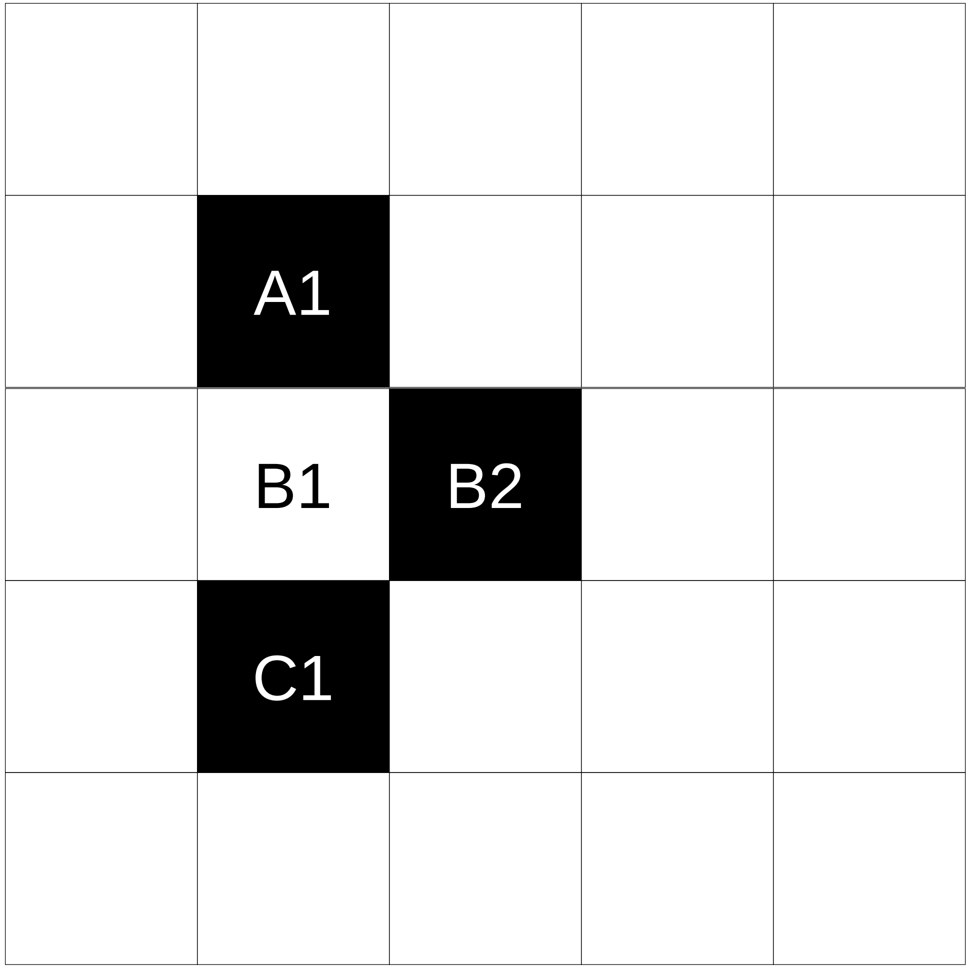
Por contraposición, una celda permanecerá muerta (blanca) si tiene un número distinto de 3 celdas vivas adyacentes.



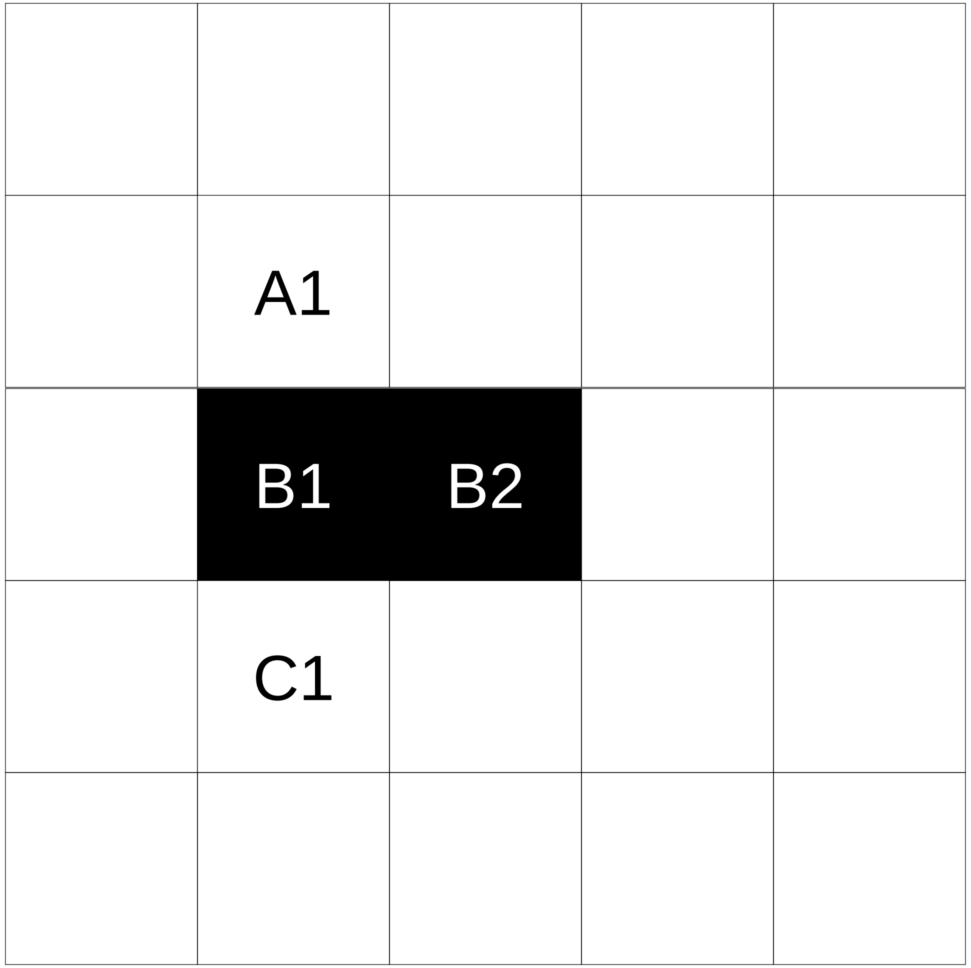
Regla de Supervivencia

En este ejemplo, la celda B2 que está viva en tiempo t, permanecerá viva (negra) en tiempo $t + 1$ porque está rodeada de 2 celdas vivas (negras) A1 y C1.



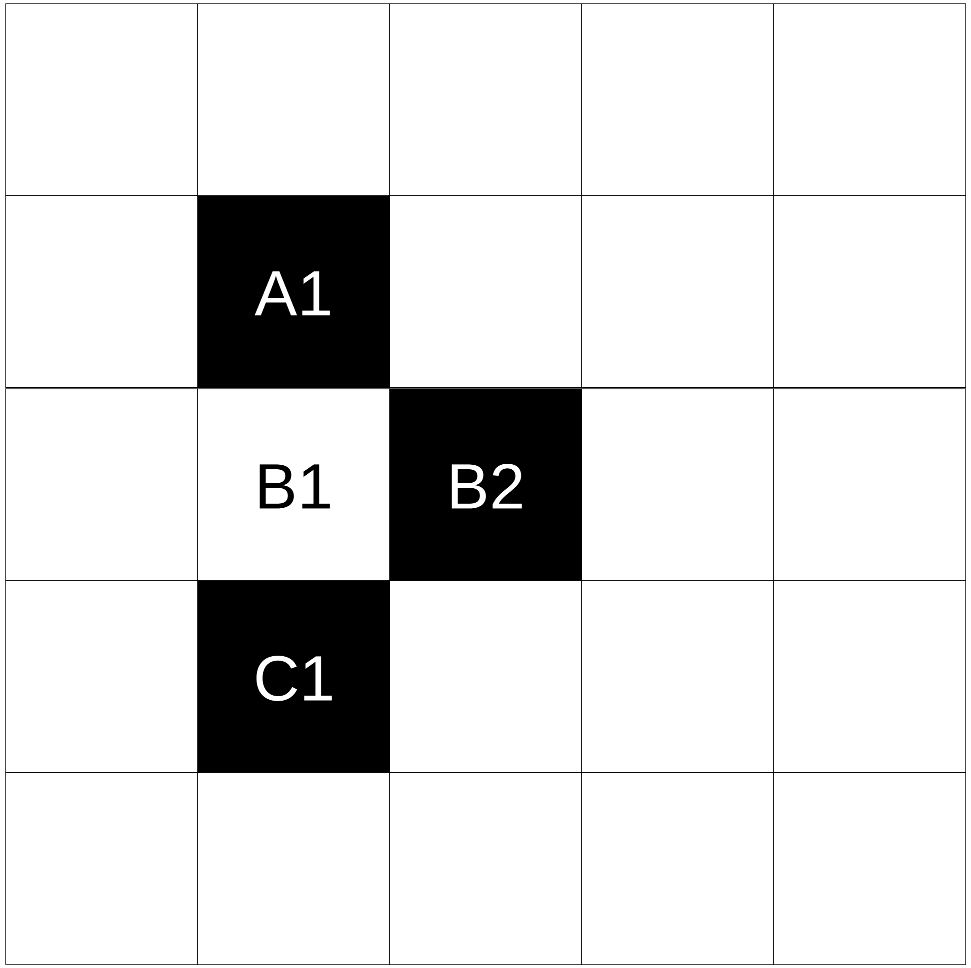


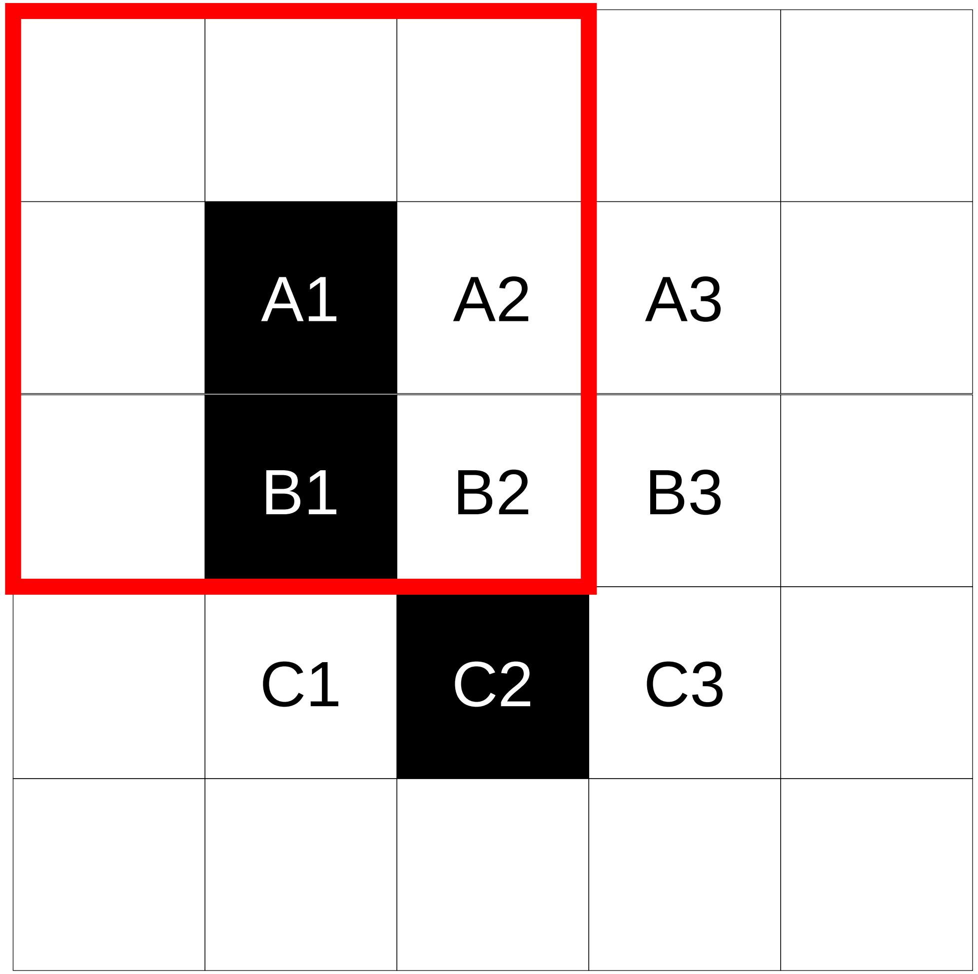
$$t \rightarrow t+1$$



Es importante tener en cuenta que todas las celdas se analizan en simultáneo.

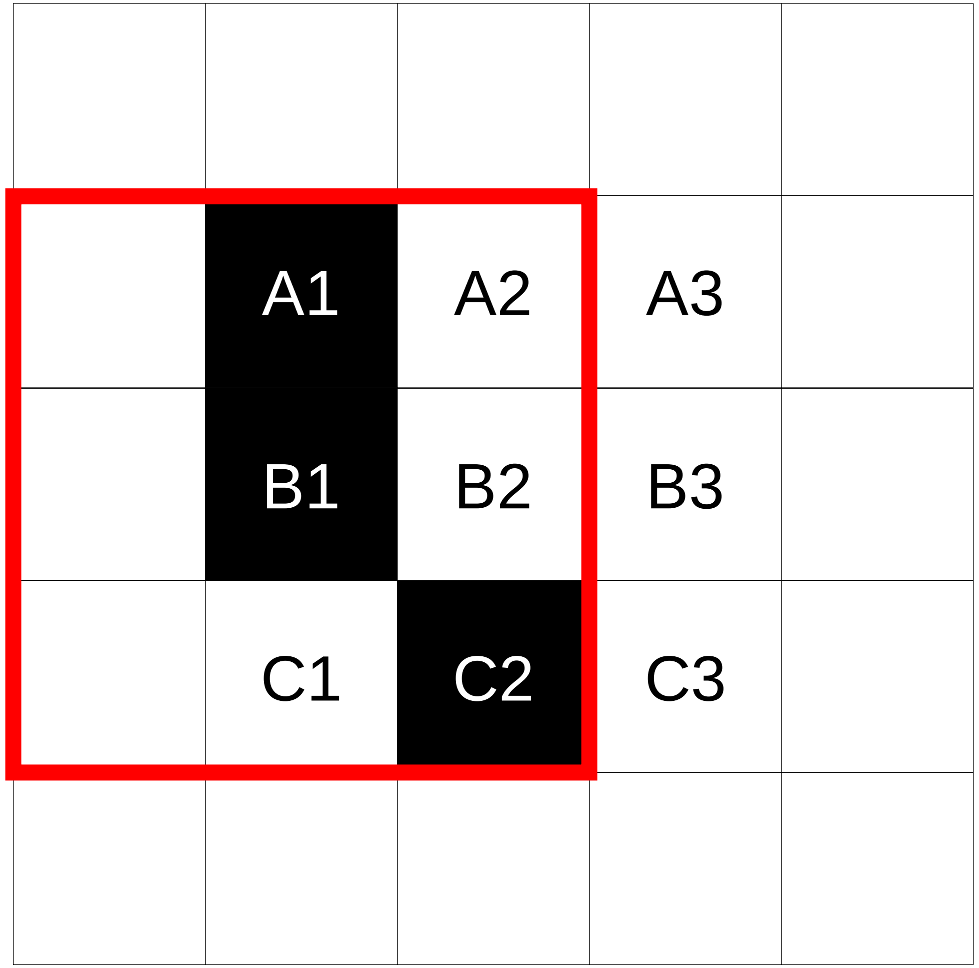
Las reglas del autómata de Conway son tales que manifiestan patrones de complejidad variados.





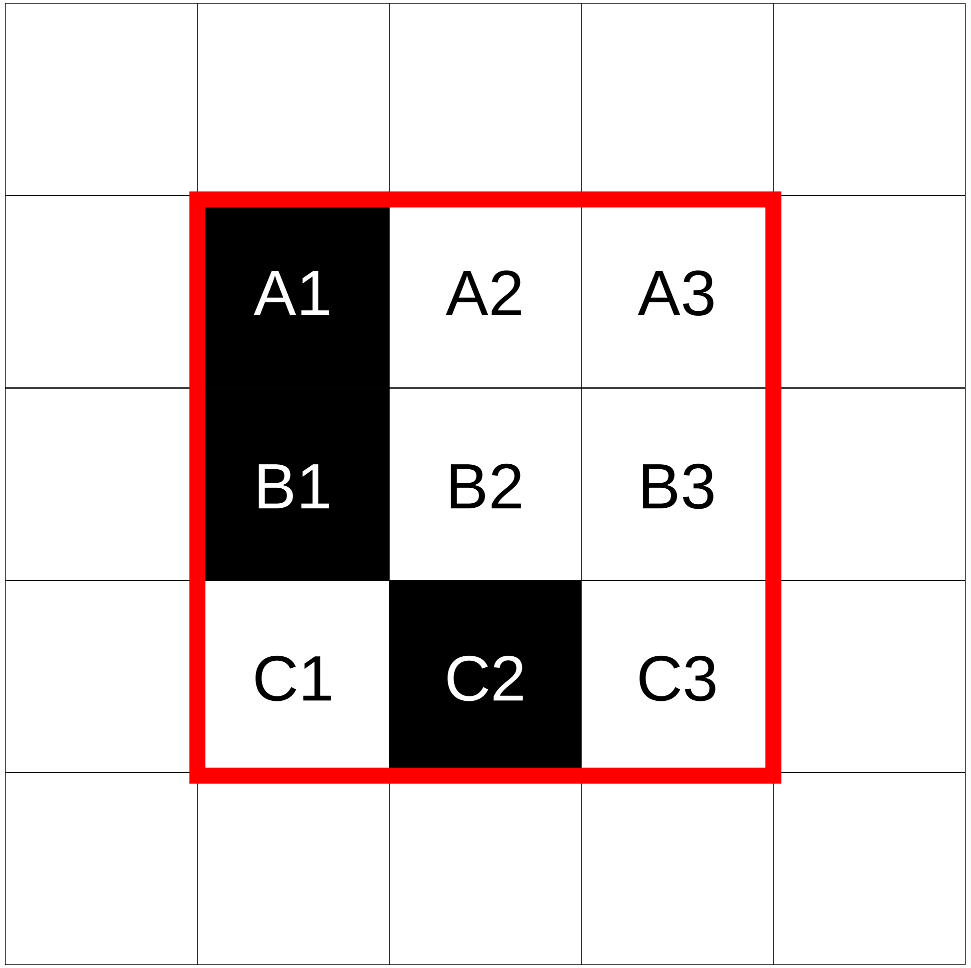
Analicemos la configuración inicial del Ejercicio 1ª. Debemos analizar celda por celda de esta configuración para establecer la configuración de estados en el tiempo $t + 1$.

Para la celda A1, tenemos 1 sola celda viva adyacente, y 7 celdas muertas, por lo que en la siguiente iteración morirá.



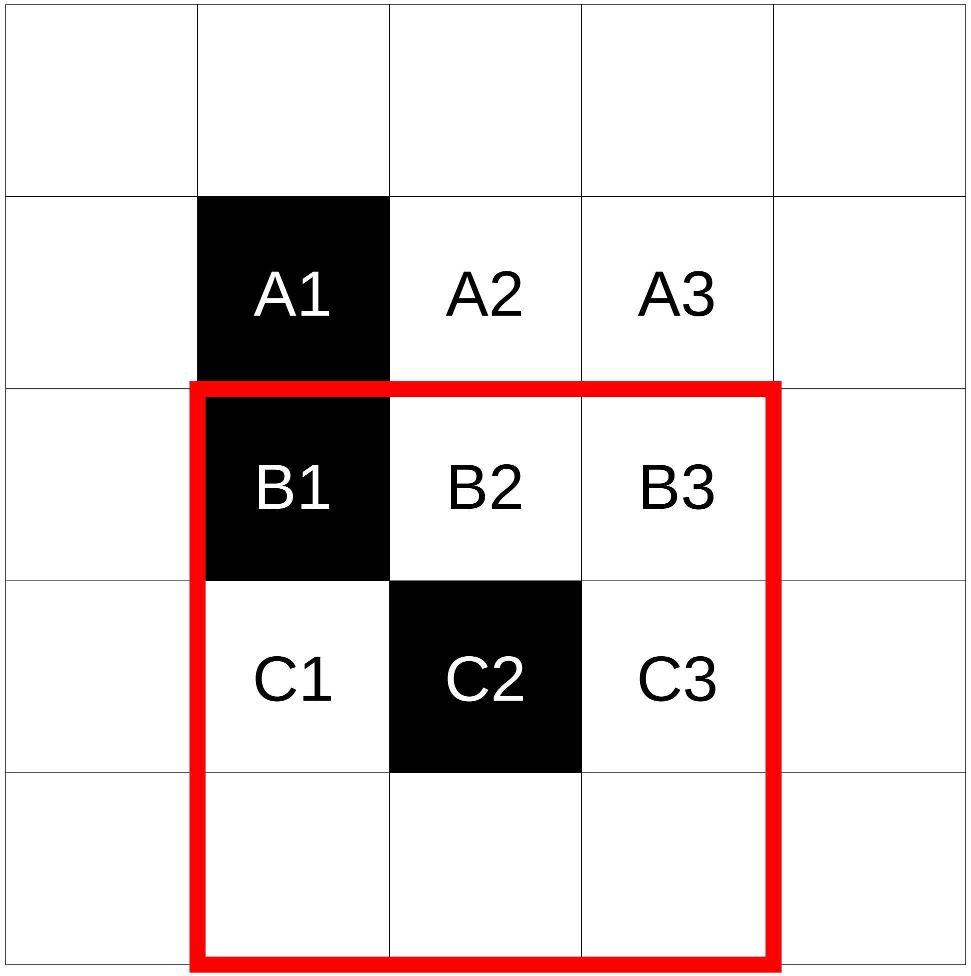
Las celdas A2, A3 y B3 están rodeadas por 1 o ninguna celda viva, por lo que en tiempo $t + 1$ permanecerán muertas.

La celda B1 está rodeada por 2 celdas vivas y 6 celdas muertas por lo que en tiempo $t + 1$ permanecerá viva.



La celda B2 está rodeada por 3 celdas vivas por lo que en tiempo $t + 1$ estará viva.

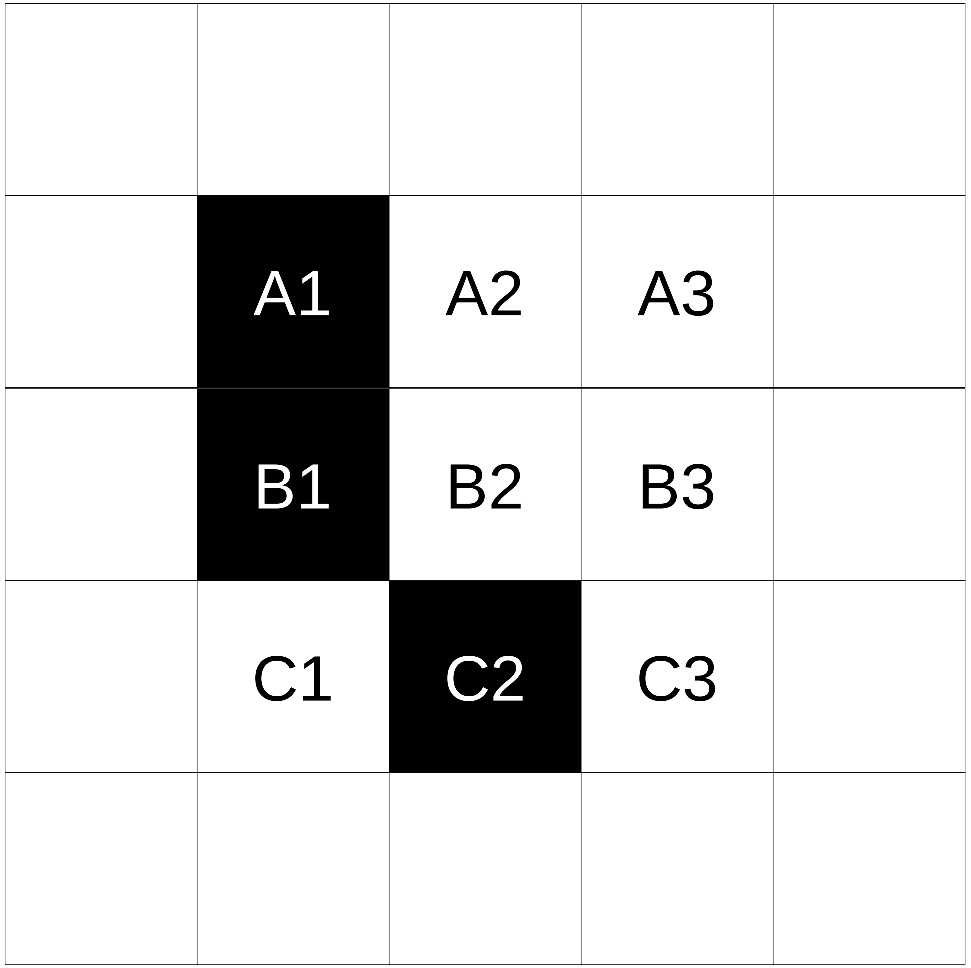
Las celdas C1 y C3 están rodeadas por 2 o 1 celda viva por lo que en tiempo $t + 1$ permanecerán muertas.



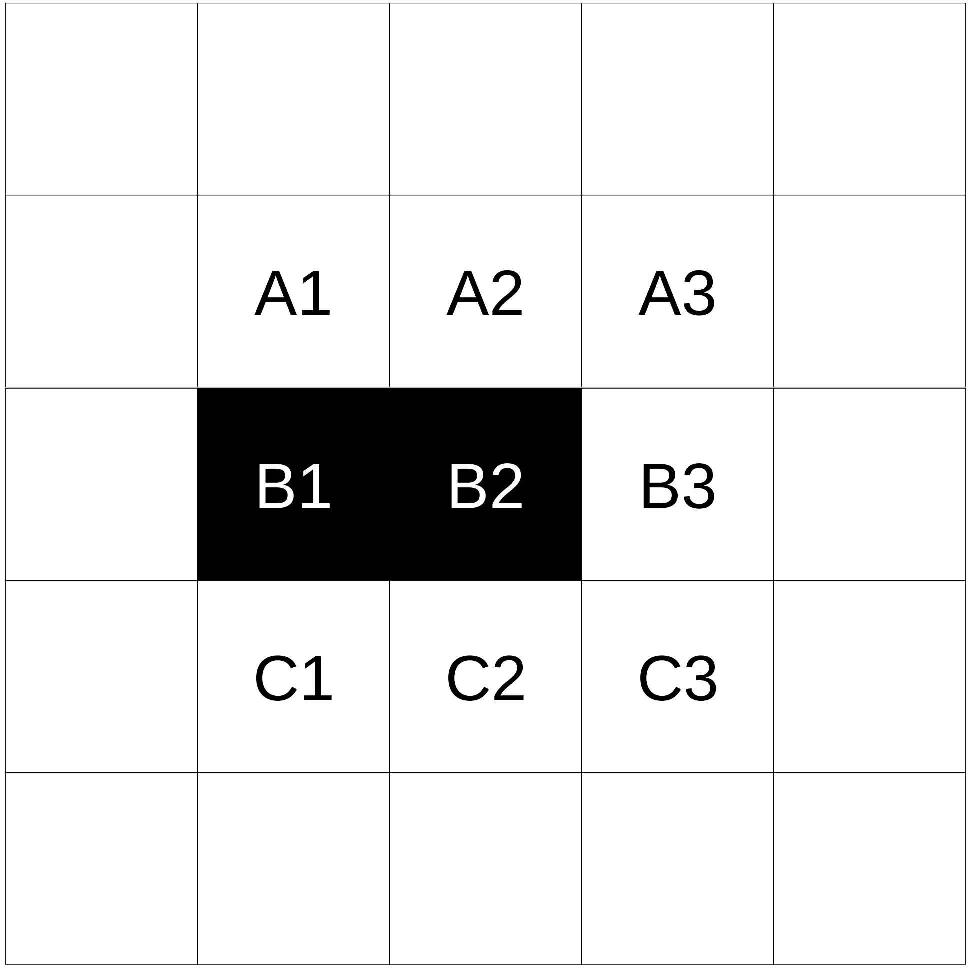
Finalmente, la celda C2 está rodeada por 1 celda viva y 7 celdas muertas adyacentes, por lo que en tiempo $t+1$ morirán.

En conclusión, en tiempo $t+1$ las celdas vivas son:

- B2 por nacimiento.
- B1 por supervivencia.



$$t \rightarrow t + 1$$



Notar que para tiempo $t+2$ ninguna de las celdas vivas en tiempo $t+1$ sobrevive, ya que están rodeadas únicamente por 1 sola celda (esto es: B1 está rodeada por 1 celda viva que es B2 y viceversa).

Por lo tanto, en tiempo $t+2$ todas las celdas estarán muertas (**aniquilación**).

	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

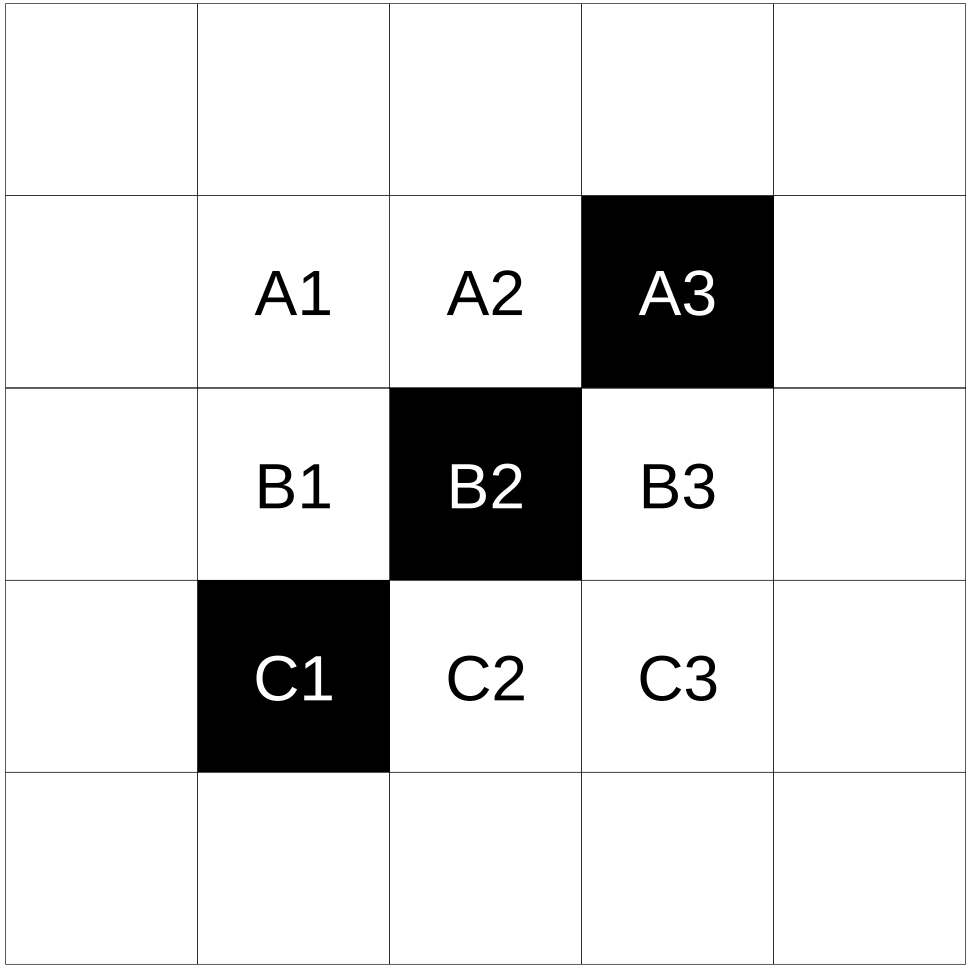
$$t + 1 \rightarrow t + 2$$

	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

Ejercicio 1C

En este configuración, las celdas C1 y A3 están rodeadas por 1 sola celda viva por lo que en la siguiente iteración estarán muertas.

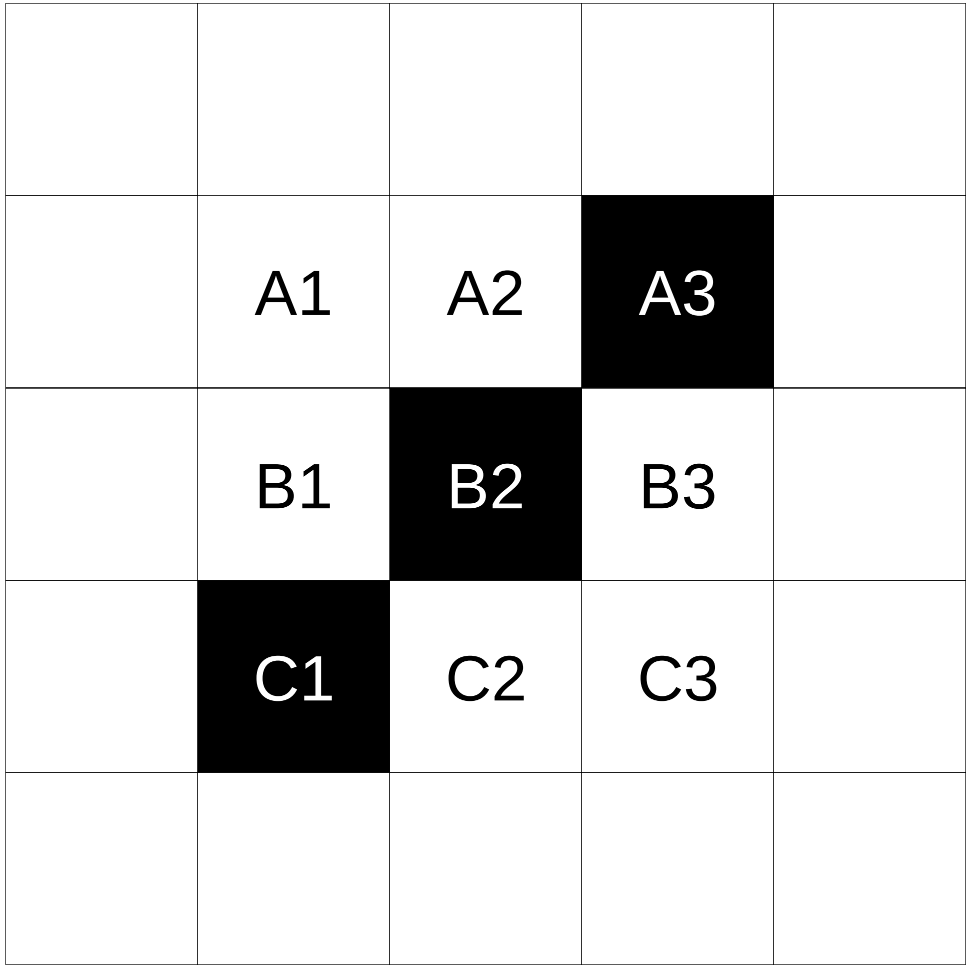
Por otro lado, la celda B2 está rodeada por dos celdas vivas (C1 y A3) por lo que sobrevive la siguiente iteración.

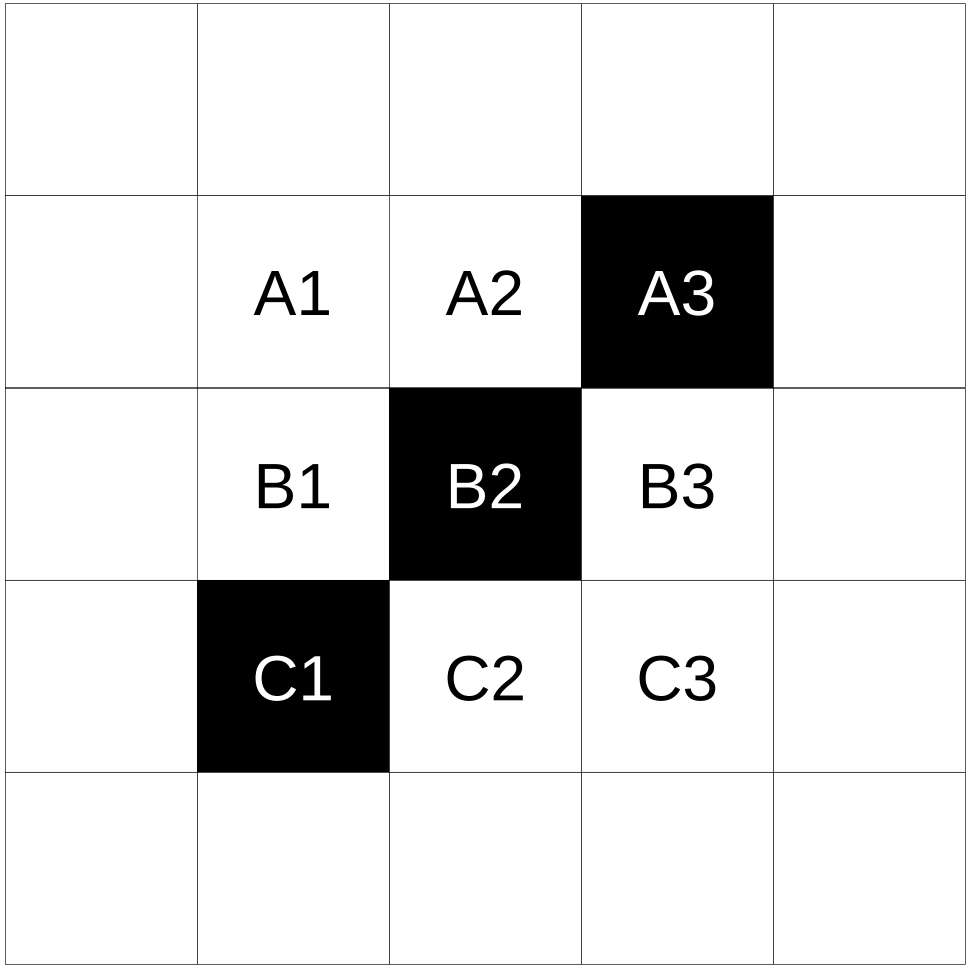


Ejercicio 1C

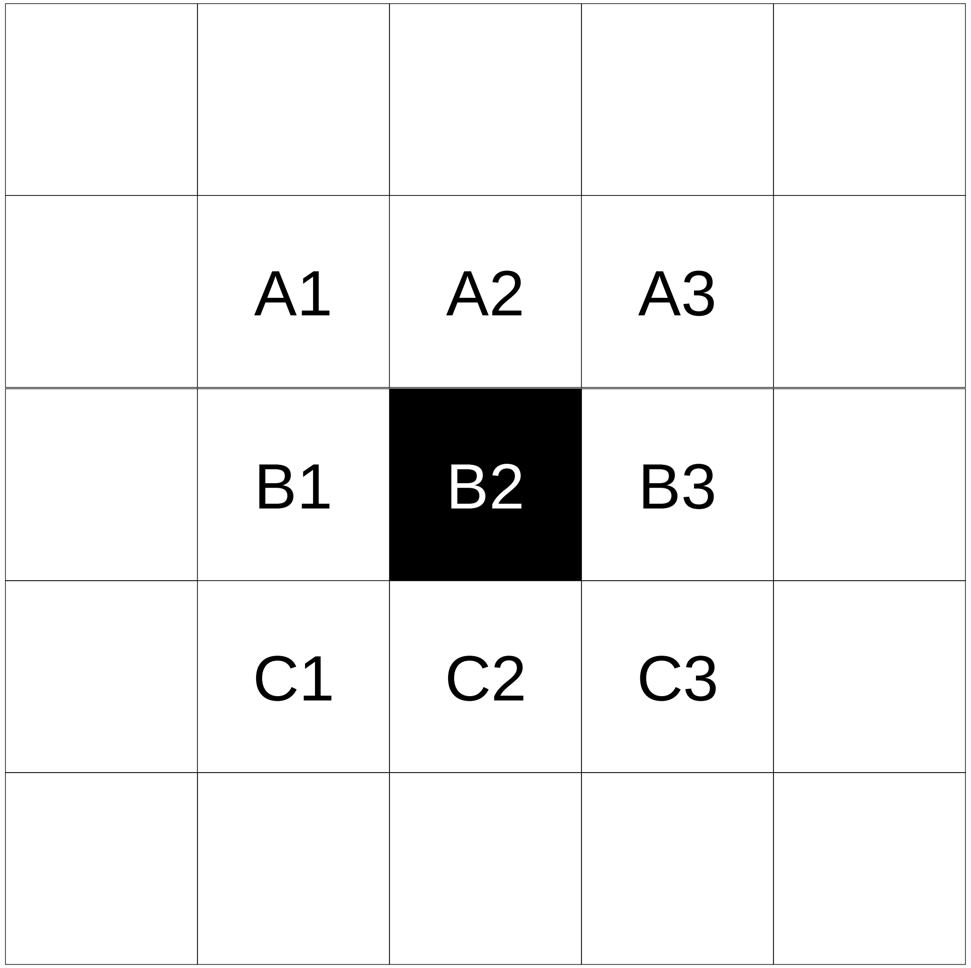
Las demás celdas que están muertas están rodeadas por 2 o menos celdas vivas por lo que no son suficientes para que se forme una nueva celda viva.

Por lo tanto, todas las celdas que están muertas, continuarán estándolo en la siguiente iteración.





$$t \rightarrow t+1$$



Ejercicio 1C

La única celda que sobrevivió fue la celda B2. Al estar rodeada por 8 celdas muertas, no sobrevivirá para la siguiente iteración y morirá.

Este es un ejemplo de una configuración inicial que lleva a la **aniquilación**.

	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

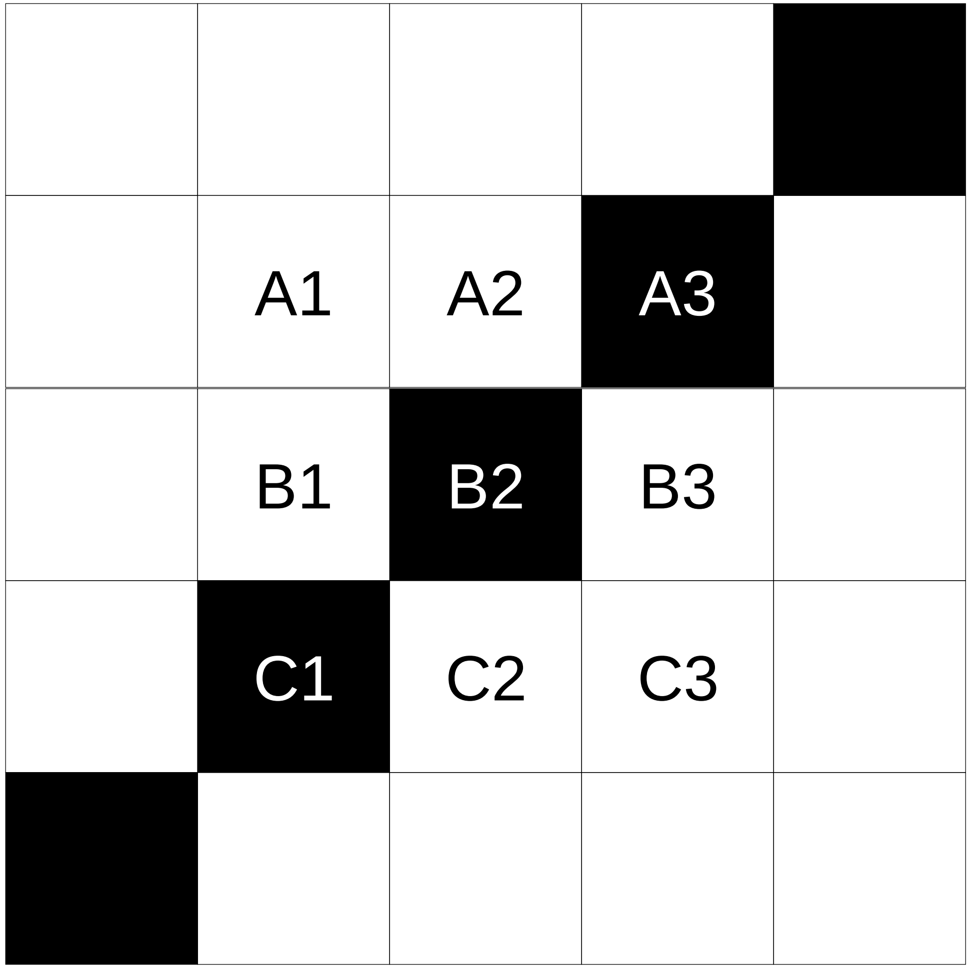
	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

$$t + 1 \rightarrow t + 2$$

	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

Ejercicio 1C

Notar que si extendemos la longitud de la configuración hacia n celdas diagonales. Lo que se observa es la aniquilación de los extremos en cada iteración hasta que todas las celdas alcanzan el estado de muerte.

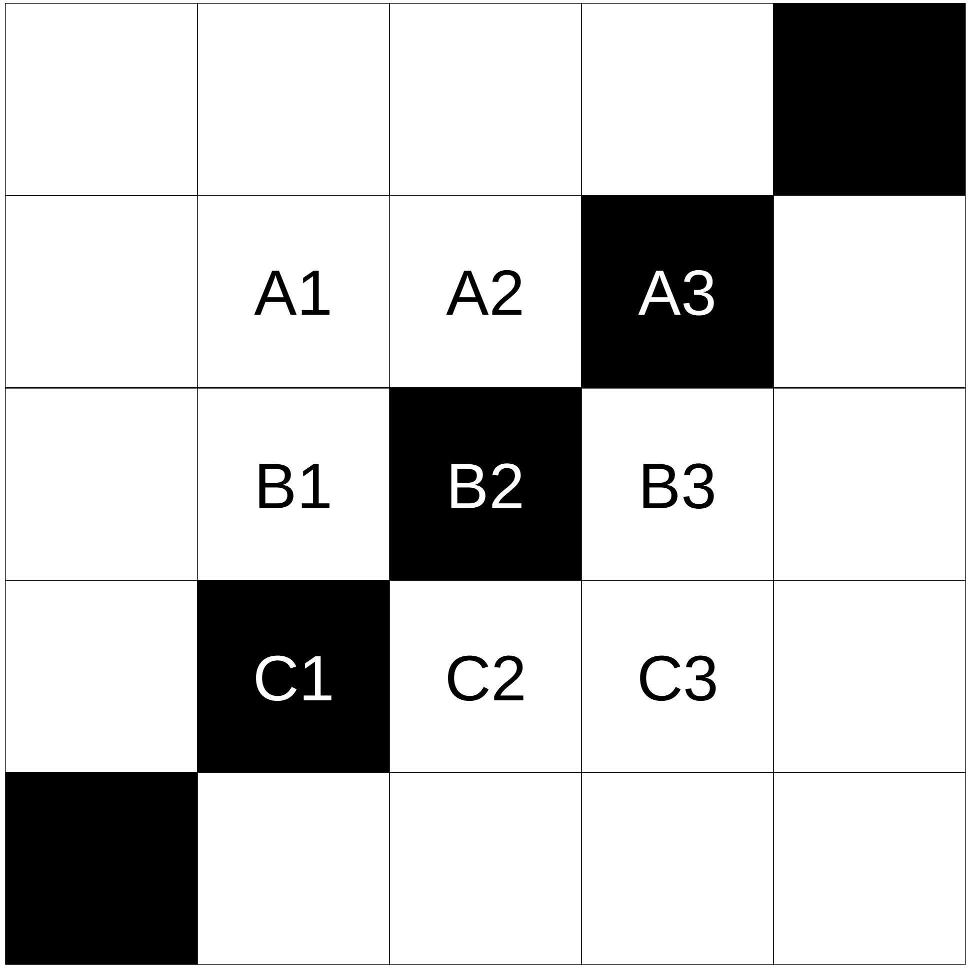


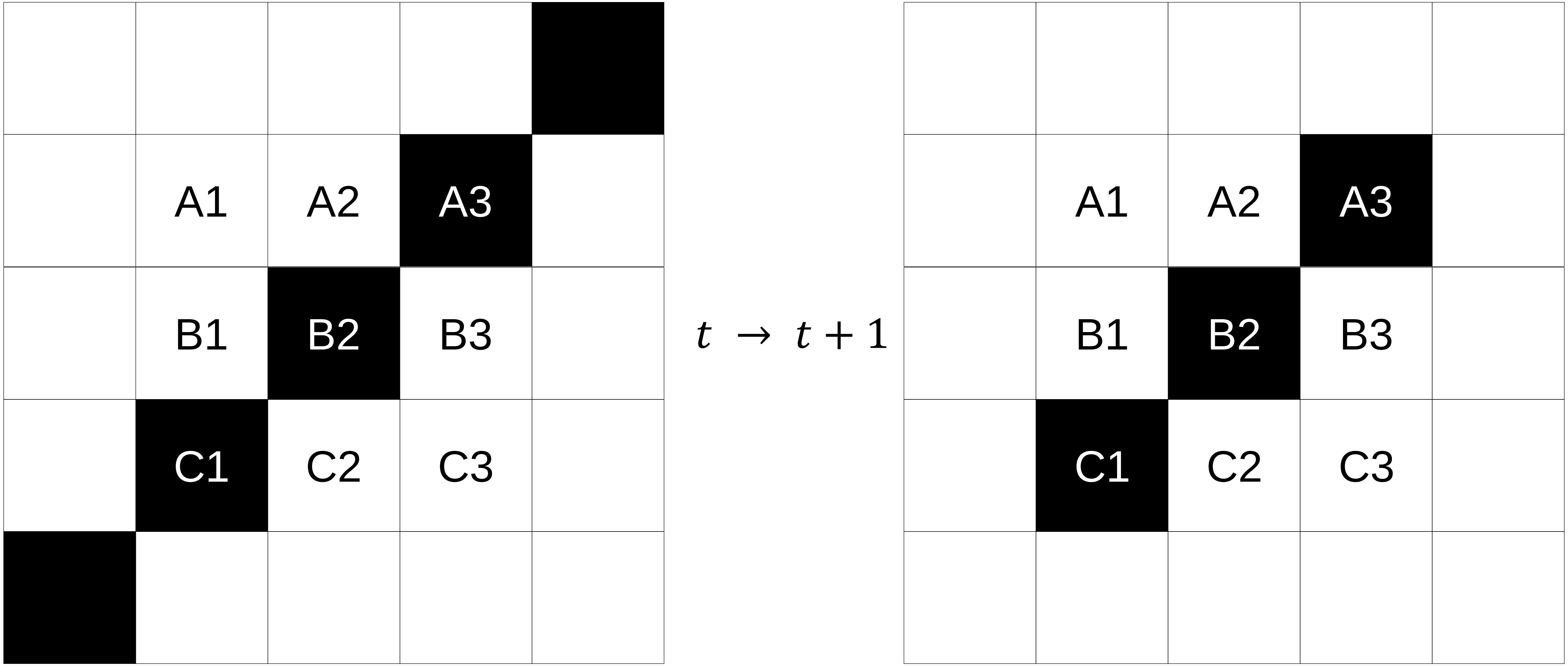
Ejercicio 1C

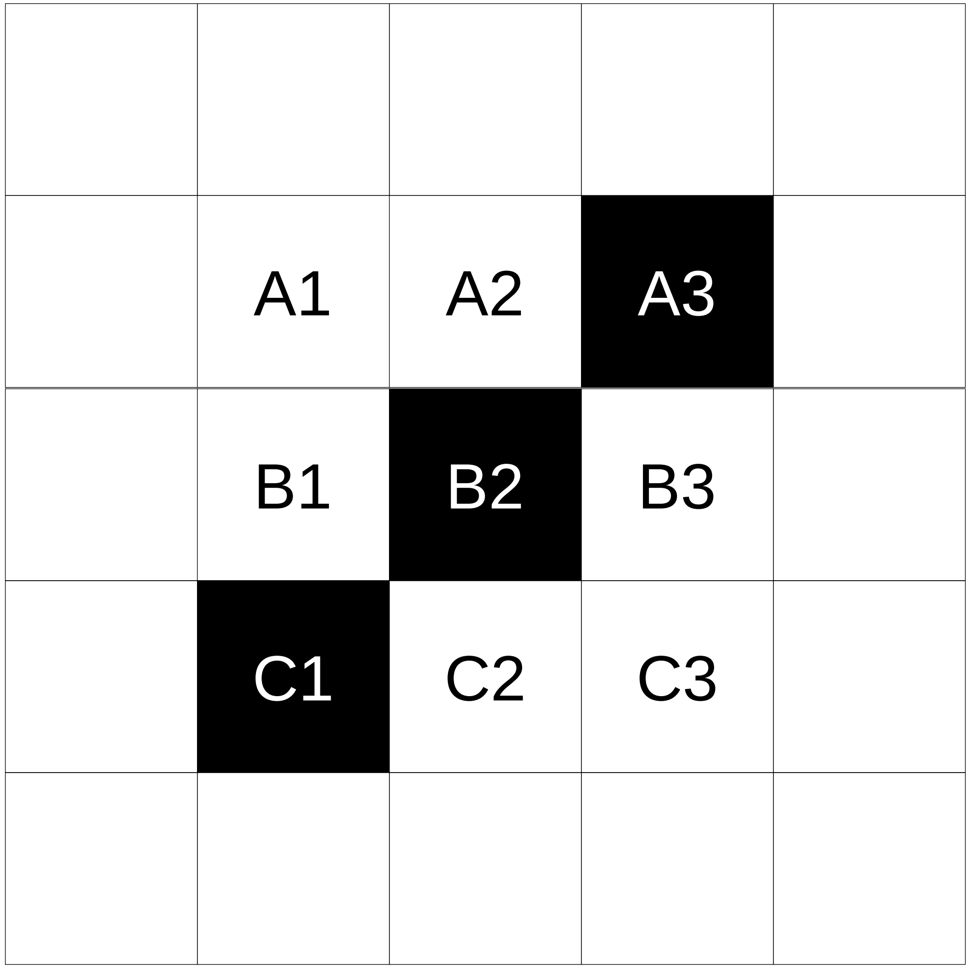
Si la cadena consta de n celdas, el número de iteraciones necesarias para aniquilar la estructura es el número entero superior más cercano a $n/2$.

Ej.: $n = 7,\ t = 4$
Ej.: $n = 16, t = 8$

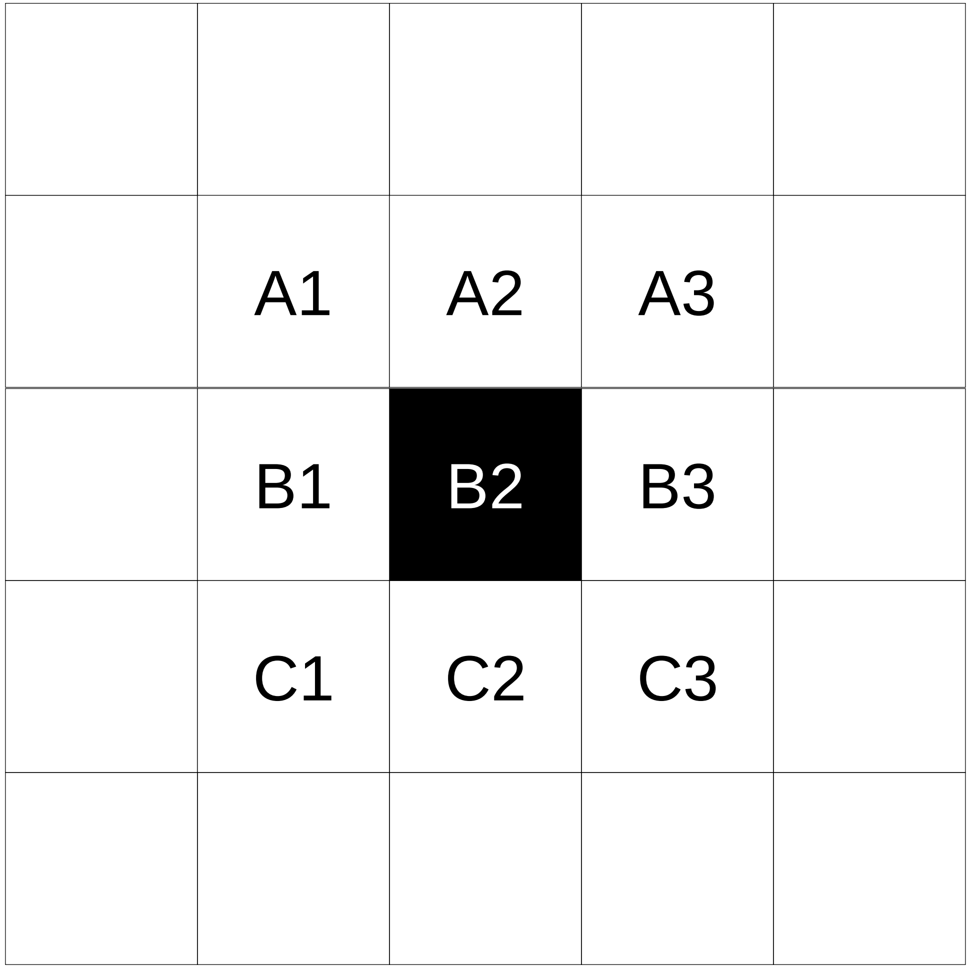
Ej.: $n = 5, t = 3$




A1 A2 A3
B1 B2 B3
C1 C2 C3
$t \rightarrow t+1$
A1 A2 A3
B1 B2 B3
C1 C2 C3



$t + 1 \rightarrow t + 2$



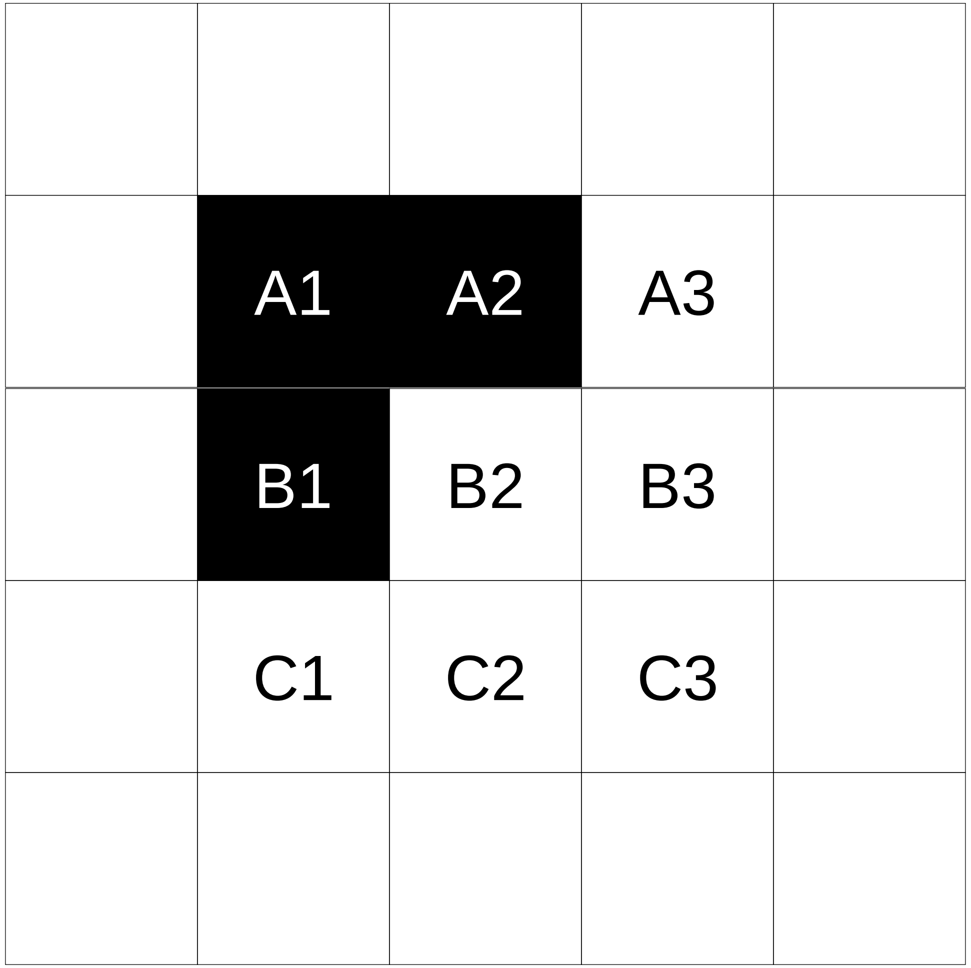
	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

$$t + 2 \rightarrow t + 3$$

	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

Ejercicio 1D

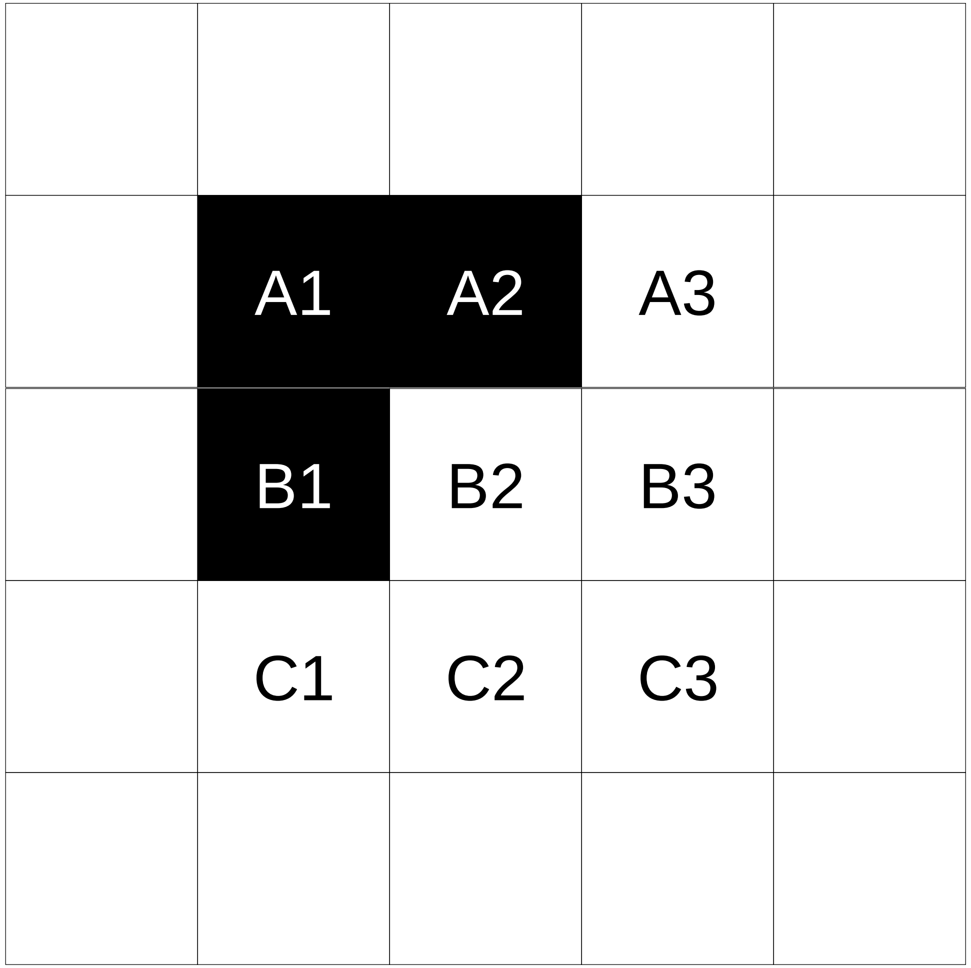
La configuración mostrada indica que todas las celdas vivas están rodeadas por dos celdas vivas adyacentes, por lo tanto, las tres celdas vivas vivirán en la siguiente iteración.

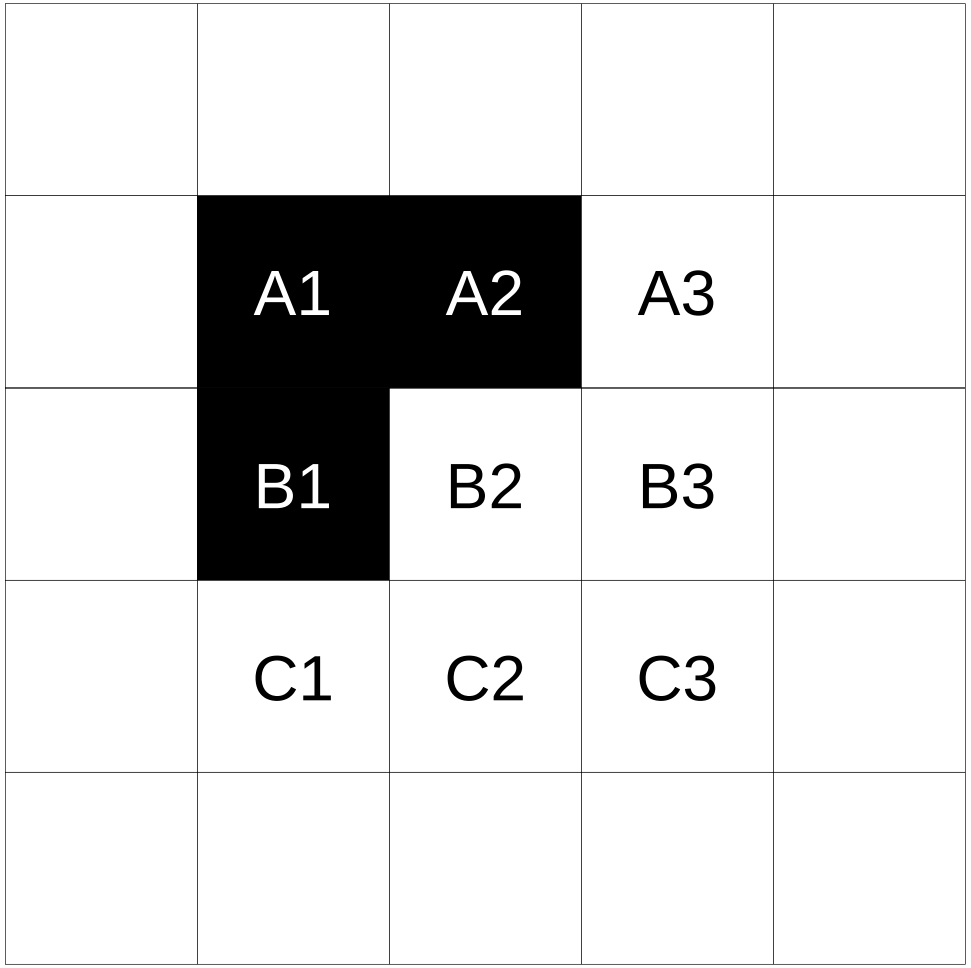


Ejercicio 1D

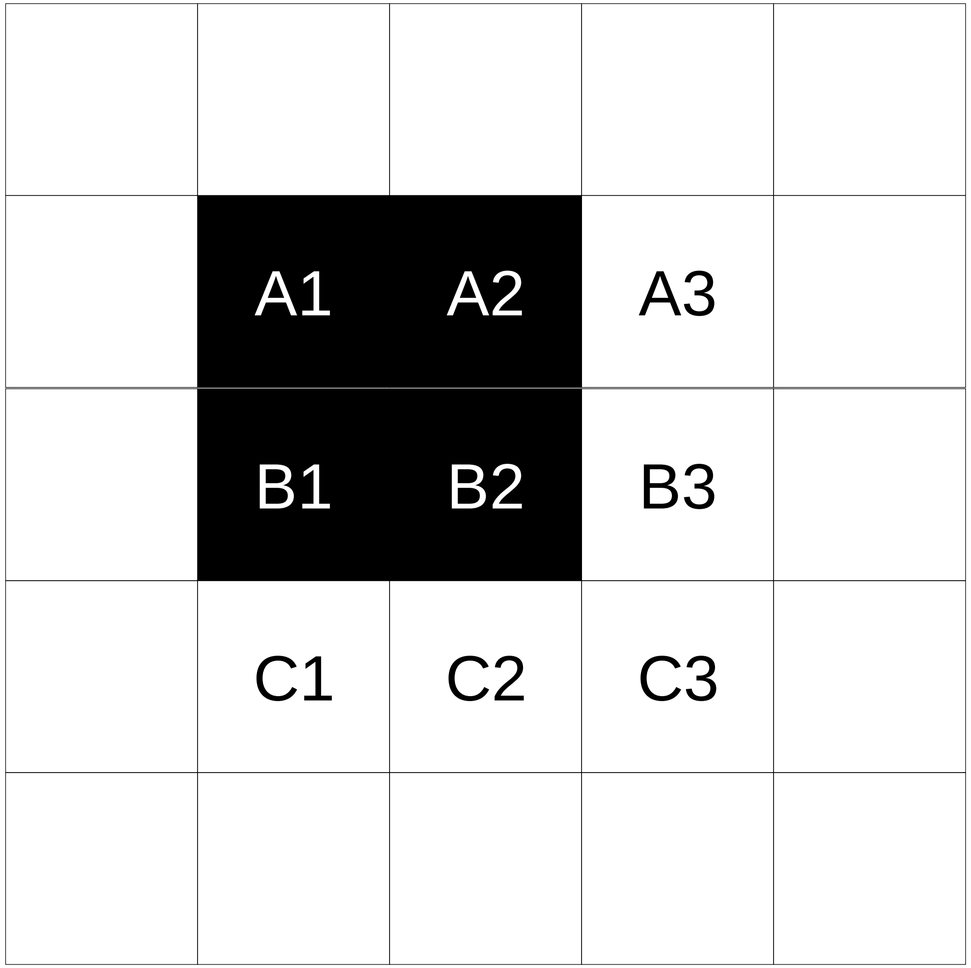
La observación notable es que la celda B2, que está muerta en tiempo t, estará viva en tiempo $t+1$ al estar rodeadas por tres celdas vivas (A1, A2 y B1).

Las demás celdas muertas (A3, B3, C1, C2 y C3) están rodeadas por 1 o ninguna celda viva por lo que permanecerán muertas en la siguiente iteración.





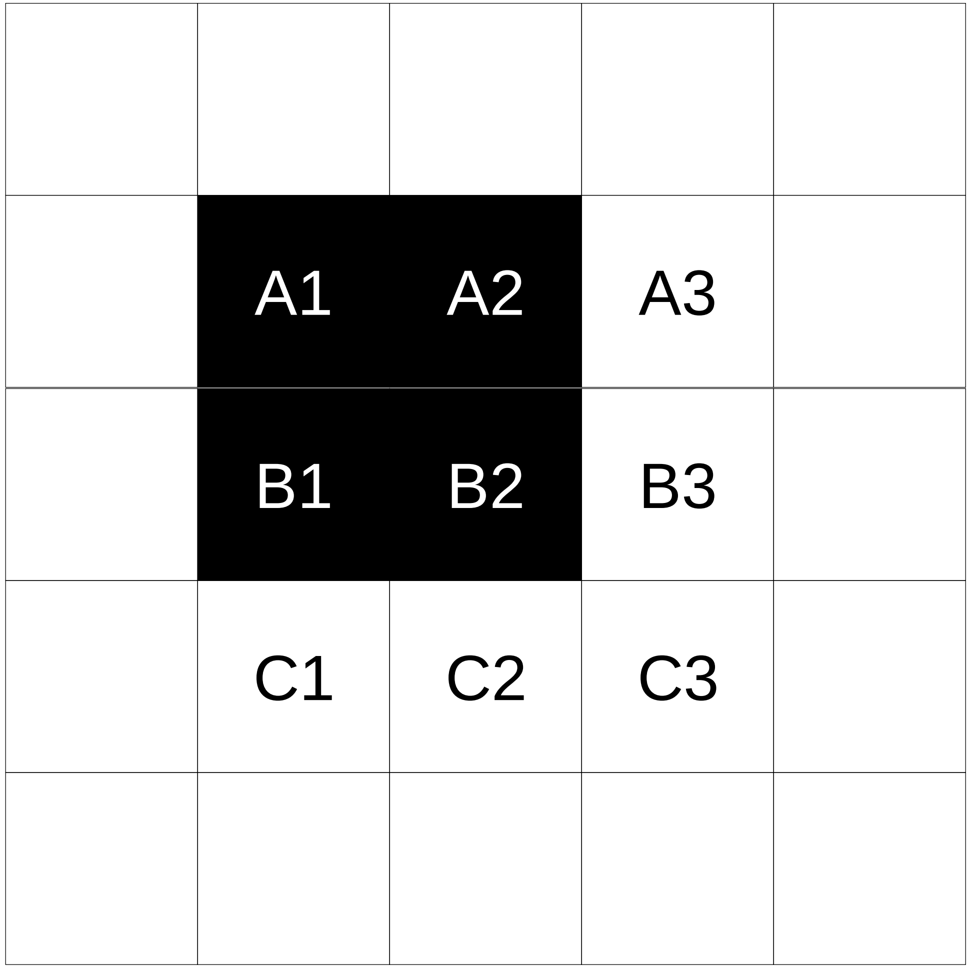
$$t \rightarrow t + 1$$



Ejercicio 1D

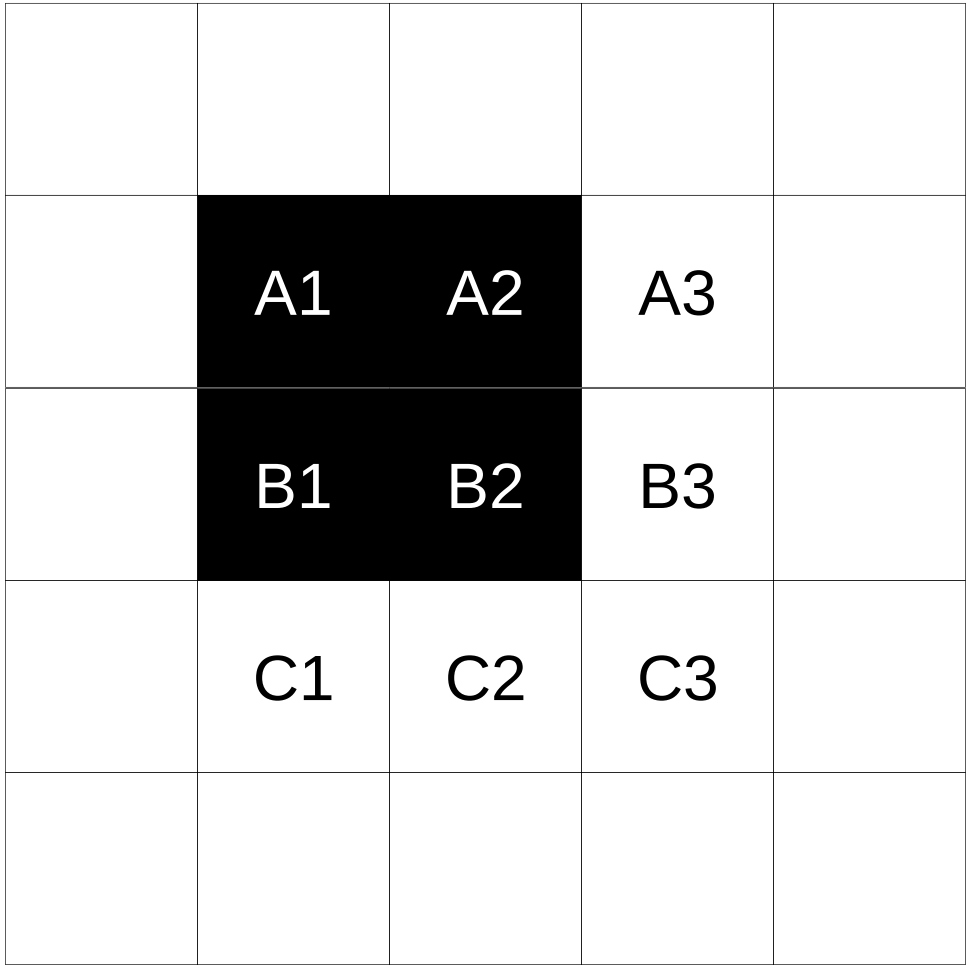
La configuración obtenida en tiempo $t + 1$ es **estable**. Esto quiere decir que permanecerá incambiada en las siguientes iteraciones.

Todas las celdas vivas están rodeadas por 3 celdas vivas adyacentes por lo que permanecerán vivas.



Ejercicio 1D

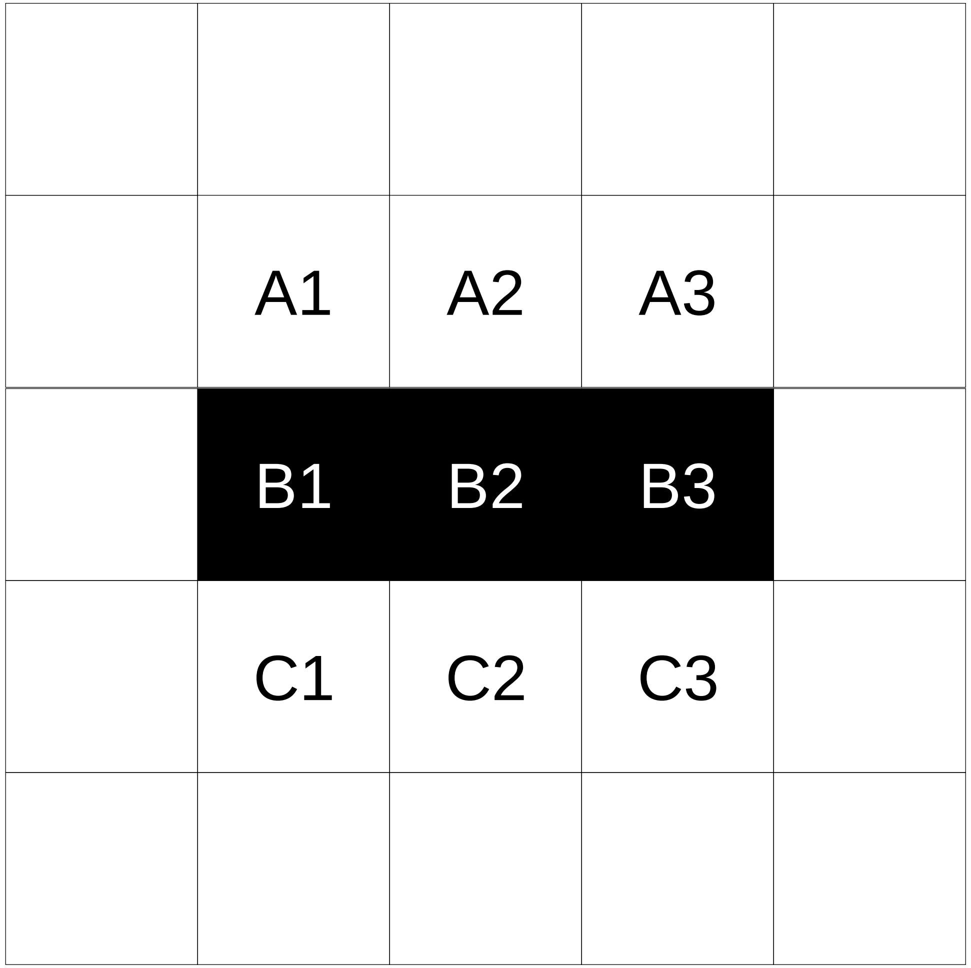
Las celdas muertas de esta configuración permanecerán muertas ya que están rodeadas por 2 o menos celdas vivas y no son capaces de nacer.



Ejercicio 1E

En esta configuración inicial tenemos 2 celdas (B1 y B3) rodeadas por 1 celda viva (B2) por lo que no logran sobrevivir a la siguiente iteración y mueren.

La celda B2 al estar rodeada por 2 celdas vivas (B1 y B3) logra sobrevivir a la siguiente iteración.

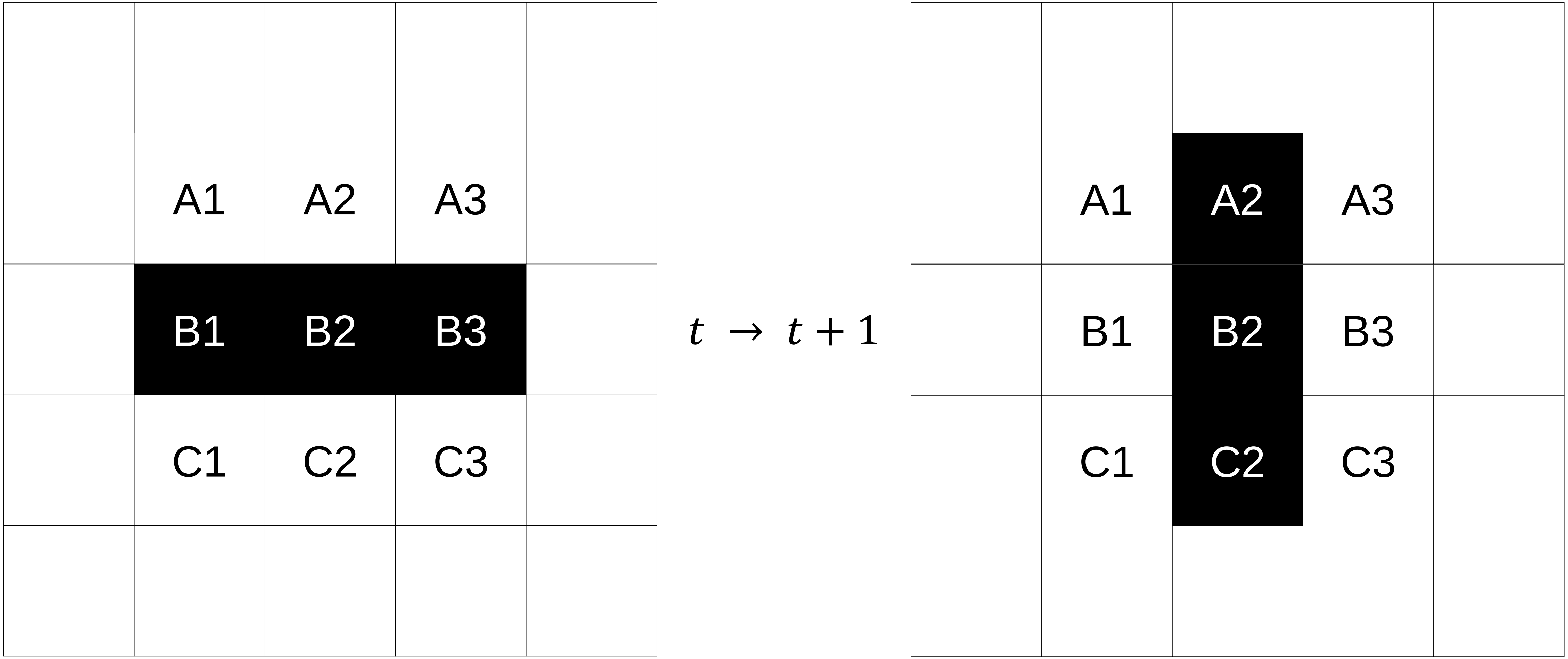


Ejercicio 1E

A su vez, las celdas A2 y C2 están rodeadas por 3 celdas vivas (B1, B2 y B3) por lo que nacen en la siguiente iteración.

Notar que las celdas A1, A3, C1 y C3 están rodeadas por 2 celdas vivas por lo que no logran tener suficientes celdas vivas adyacentes para nacer en la siguiente iteración.

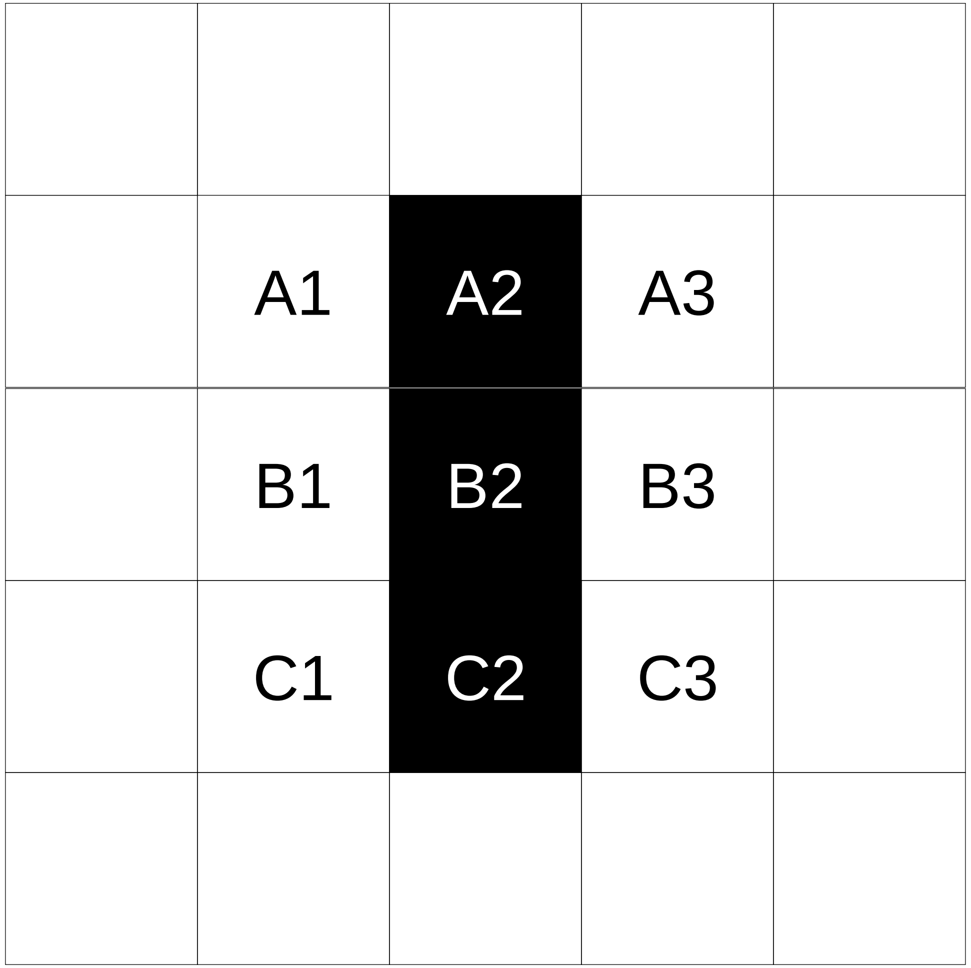
	A1	A2	A3	
	B1	B2	B3	
	C1	C2	C3	

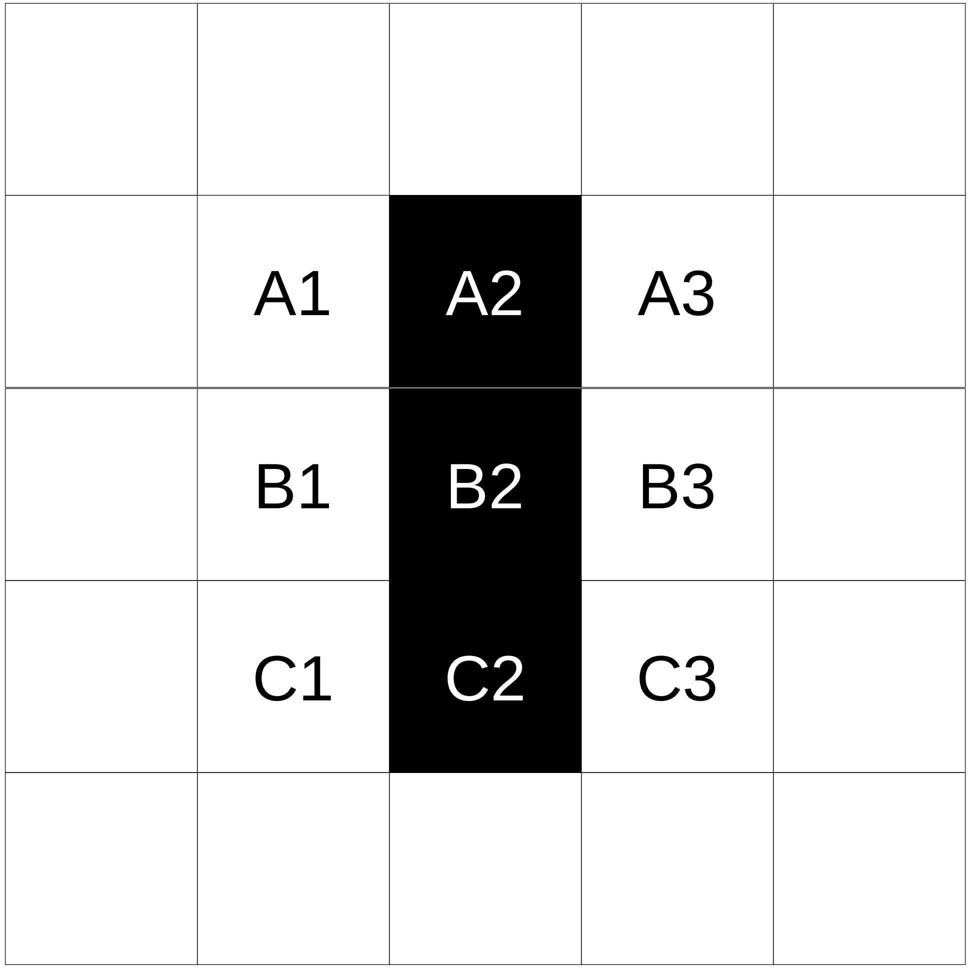

A1 A2 A3
B1 B2 B3
C1 C2 C3
$t \to t+1$
A1 A2 A3
B1 B2 B3
C1 C2 C3

Ejercicio 1E

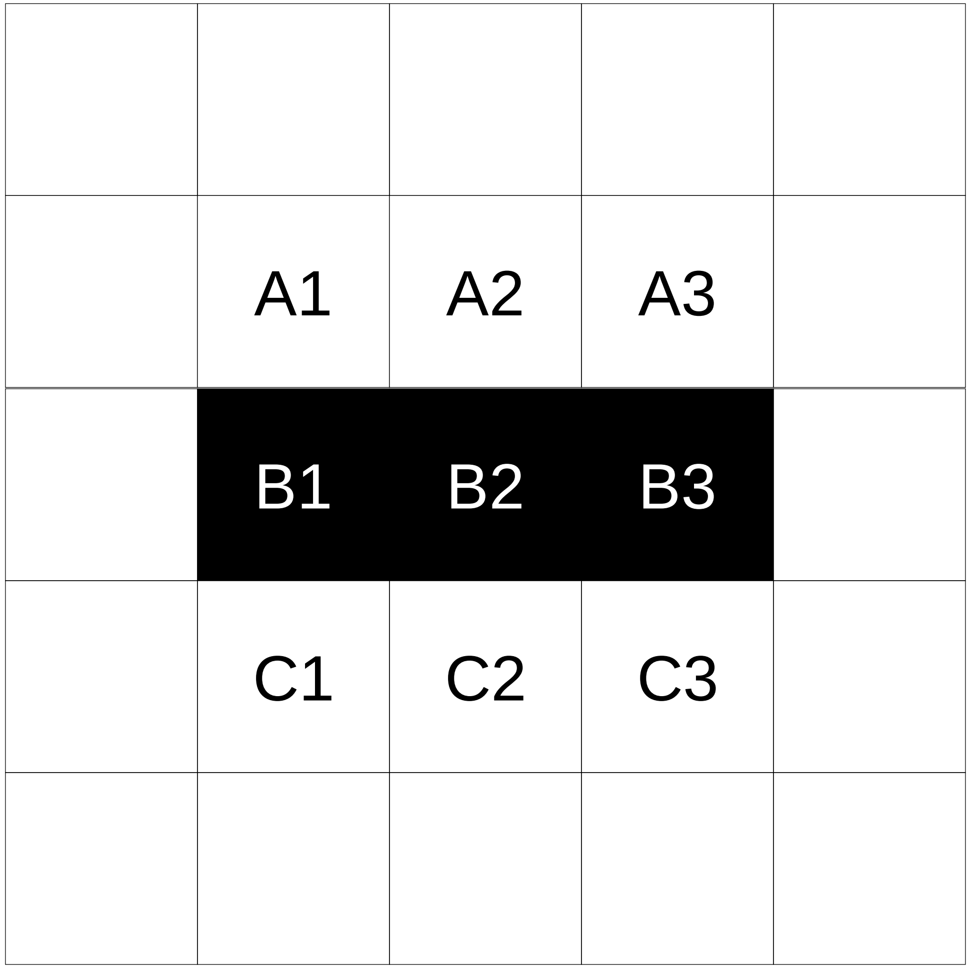
Notar que la nueva configuración es una rotación de 90° de la configuración original.

Las celdas B1 y B3 vuelven a nacer al estar rodeadas de 3 celdas vivas (A2, B2 y C2), y las celdas A2 y C2 vuelven a morir en la siguiente iteración al estar rodeadas solamente por 1 celda viva (B2).





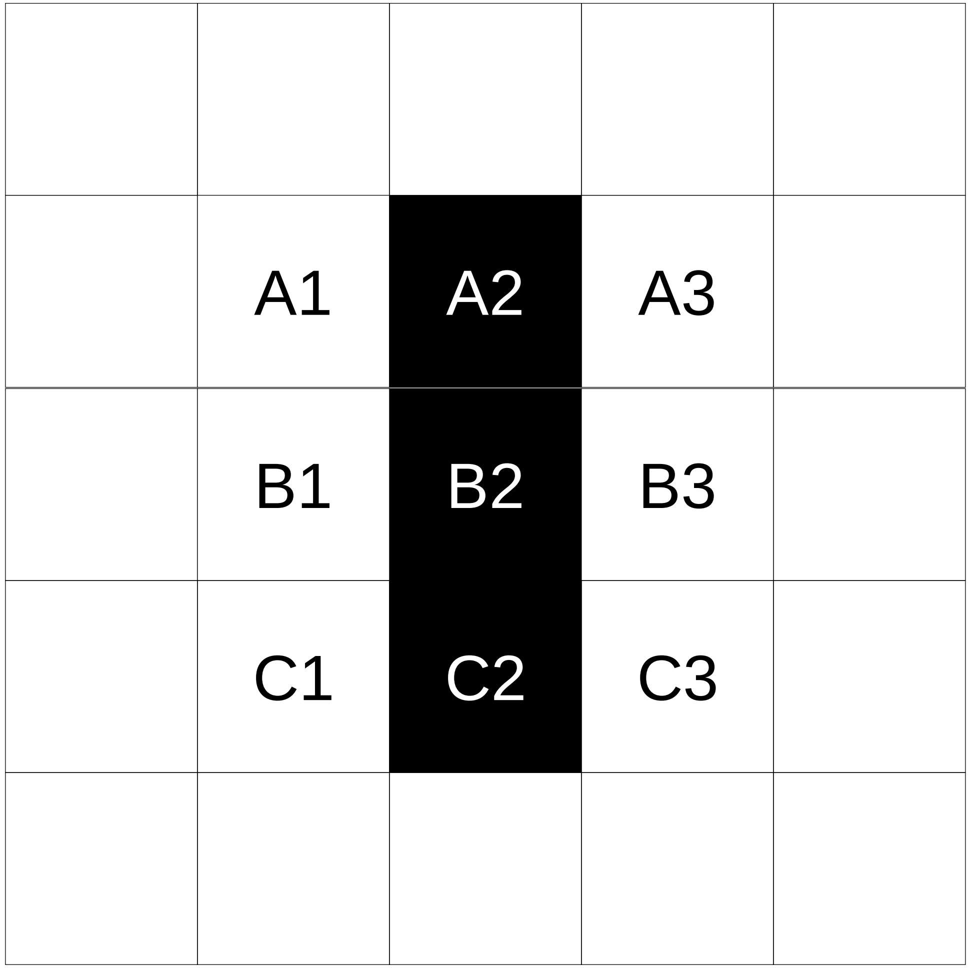
$t + 1 \rightarrow t + 2$



Ejercicio 1E

A este tipo de configuraciones que retornan al estado original se les conoce como **Osciladores**.

Notar que la evolución temporal de *celdas consecutivas ortogonales* es diferente a la evolución temporal de *celdas consecutivas diagonales* (Ej.1C)



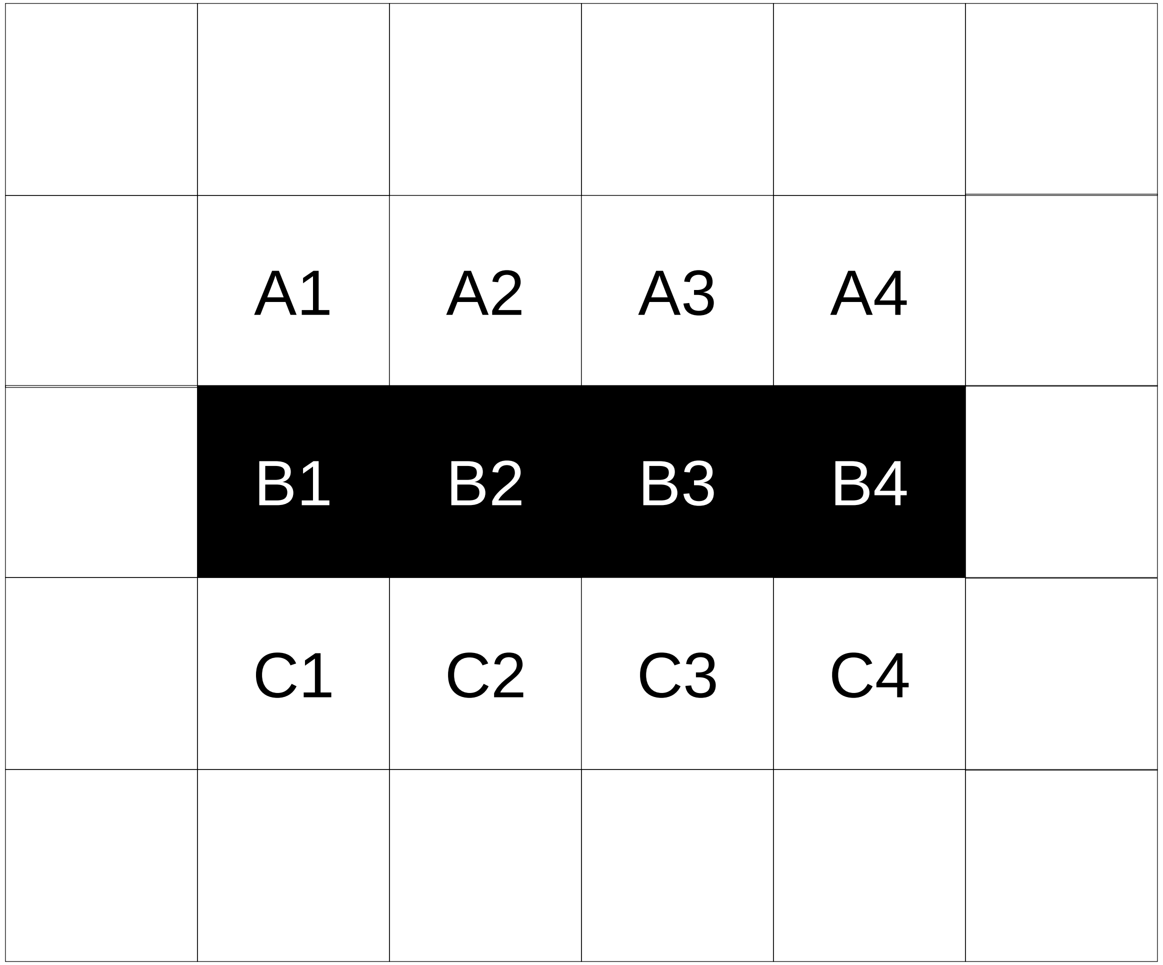
Para la segunda parte de este ejercicio conviene modificar la enumeración de la grilla para poder abarcar una mayor cantidad de celdas.

En este caso trabajaremos con una grilla de 3 filas y 5 columnas.

	A1	A2	A3	A4	
	B1	B2	B3	B4	
	C1	C2	C3	C4	

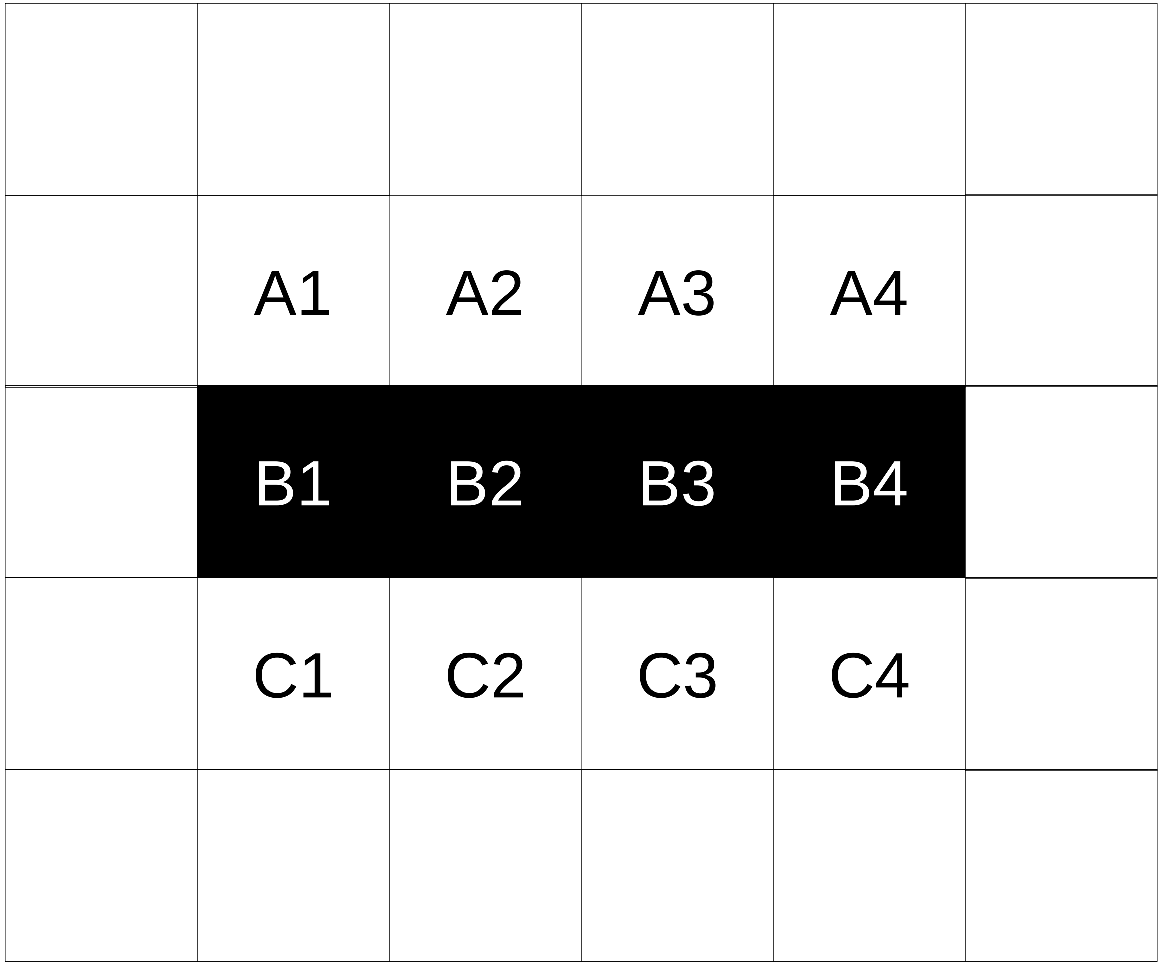
Tetrómino 1B

Para esta estructura notamos que las celdas del centro (B2 y B3) están rodeadas por 2 celdas vivas cada una por lo que vivirán en la siguiente iteración.



Tetrómino 1B

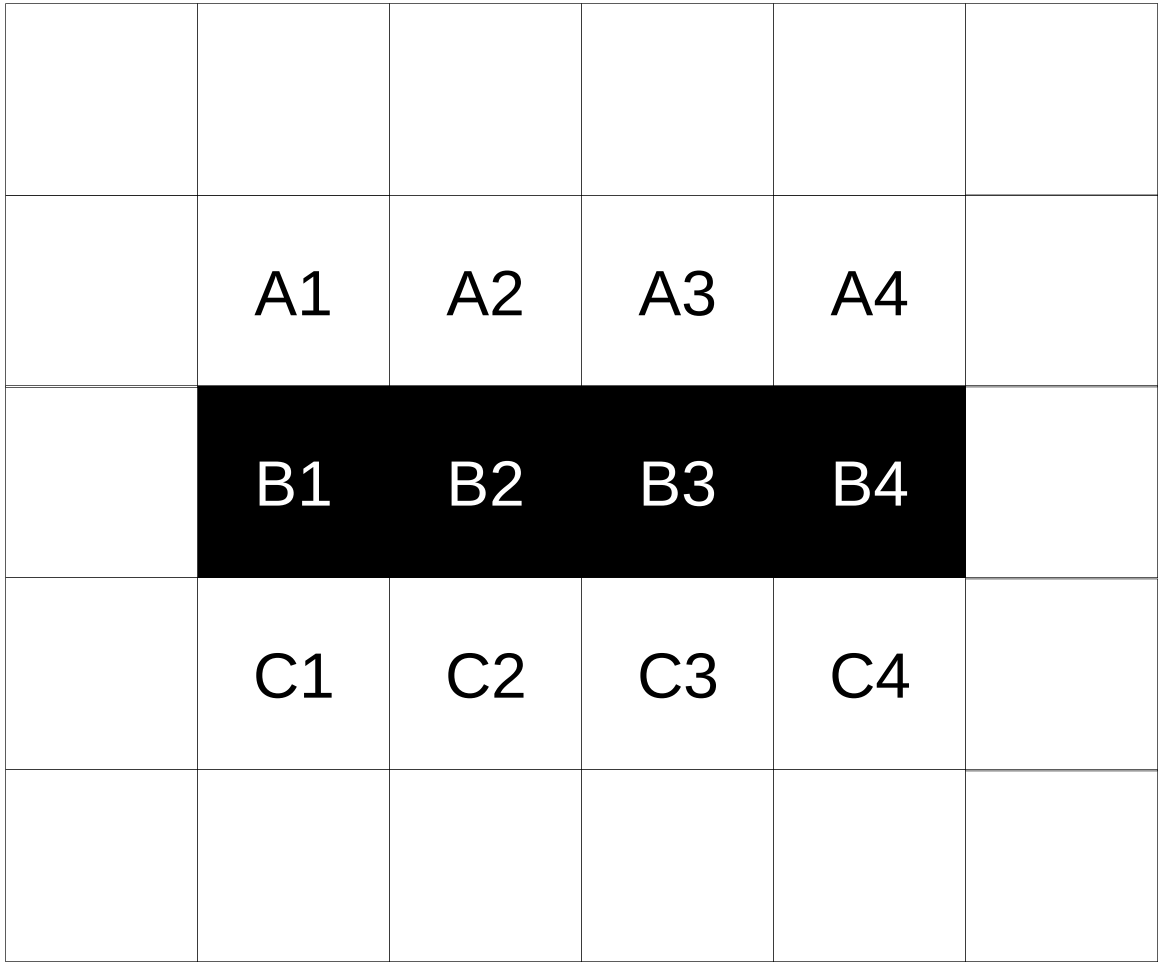
Las celdas B1 y B4 de los extremos están rodeadas solamente por 1 celda viva cada una por lo que no vivirán en la siguiente iteración.

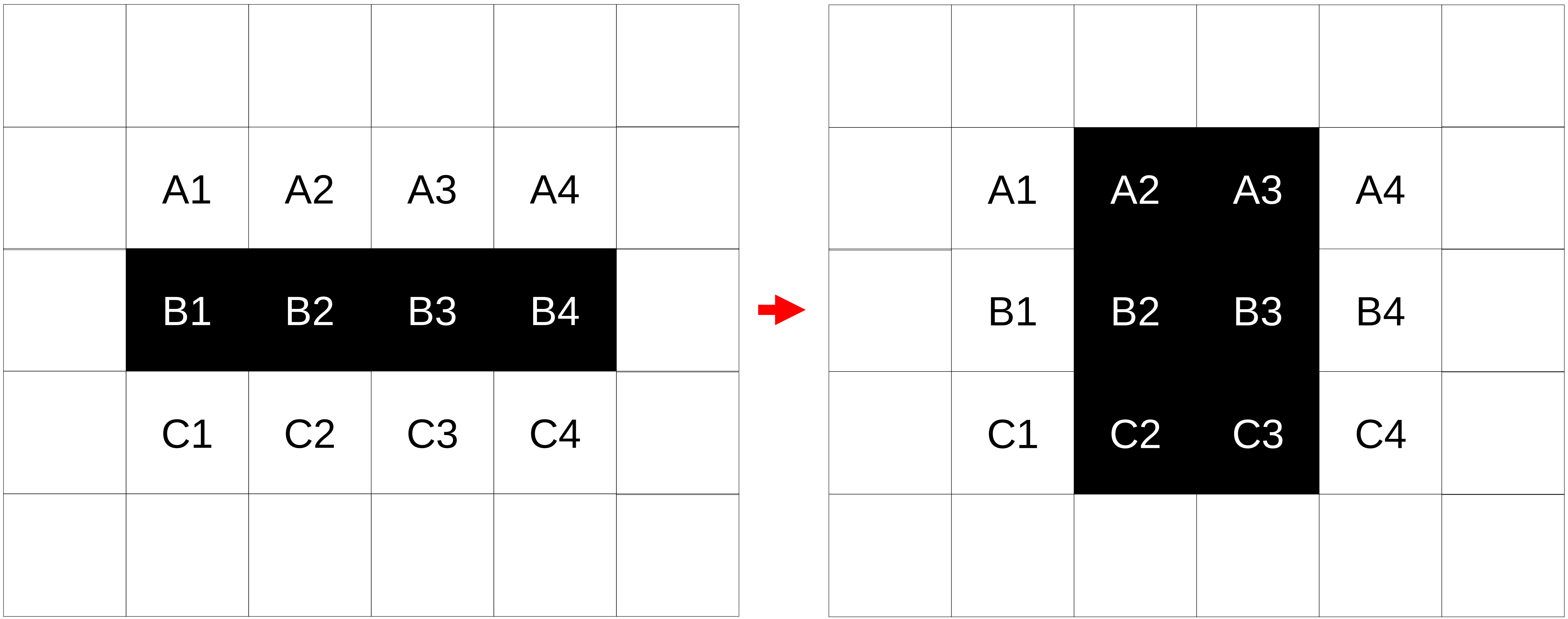


Tetrómino 1B

Las celdas A2, A3, C2 y C3 están rodeadas cada una por 3 celdas vivas, y por lo tanto, en esos espacio nacerá una nueva celda.

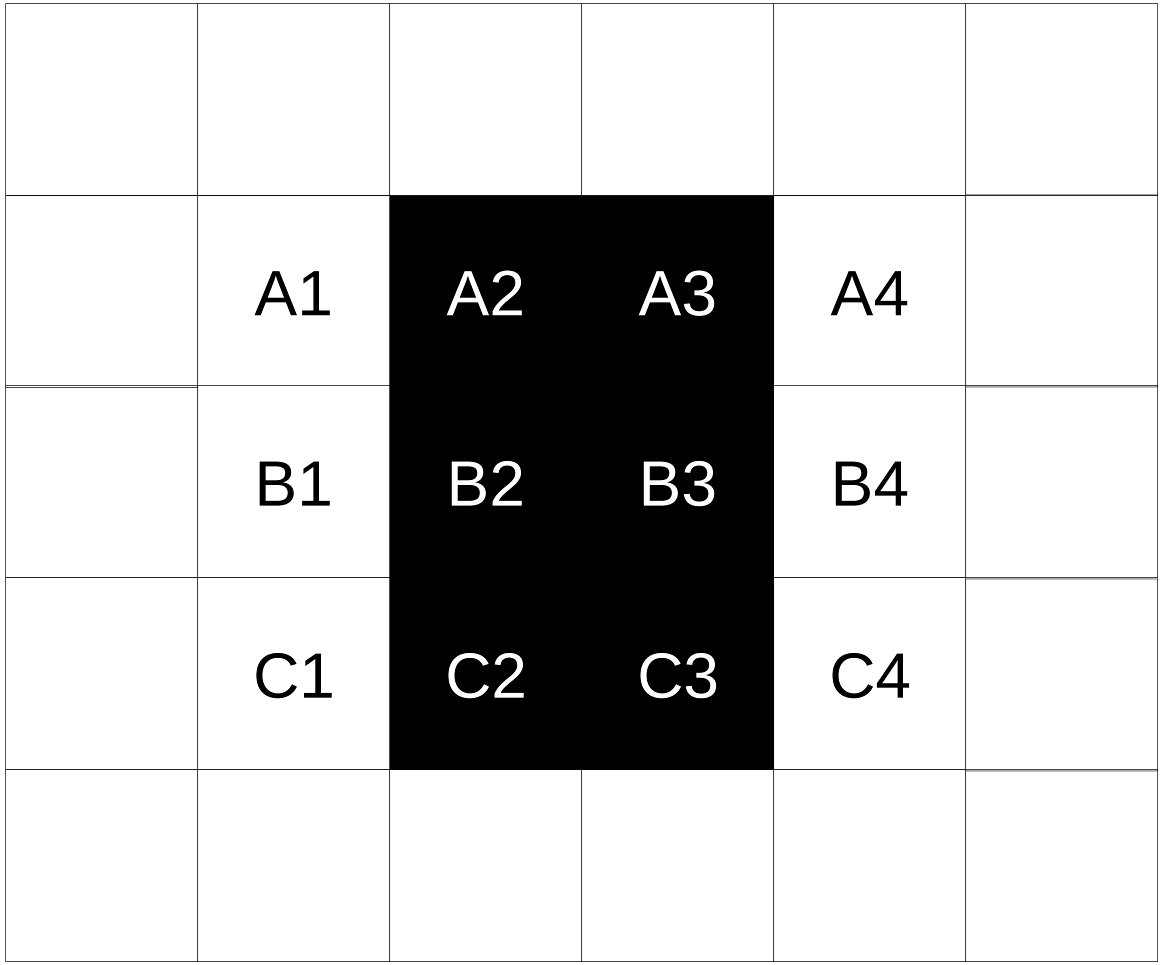
Las celdas A1, A4, C1 y C4 están rodeadas por 2 celdas vivas y permanecerán muertas.




A1 A2 A3 A4
B1 B2 B3 B4
C1 C2 C3 C4
A1 A2 A3 A4
B1 B2 B3 B4
C1 C2 C3 C4

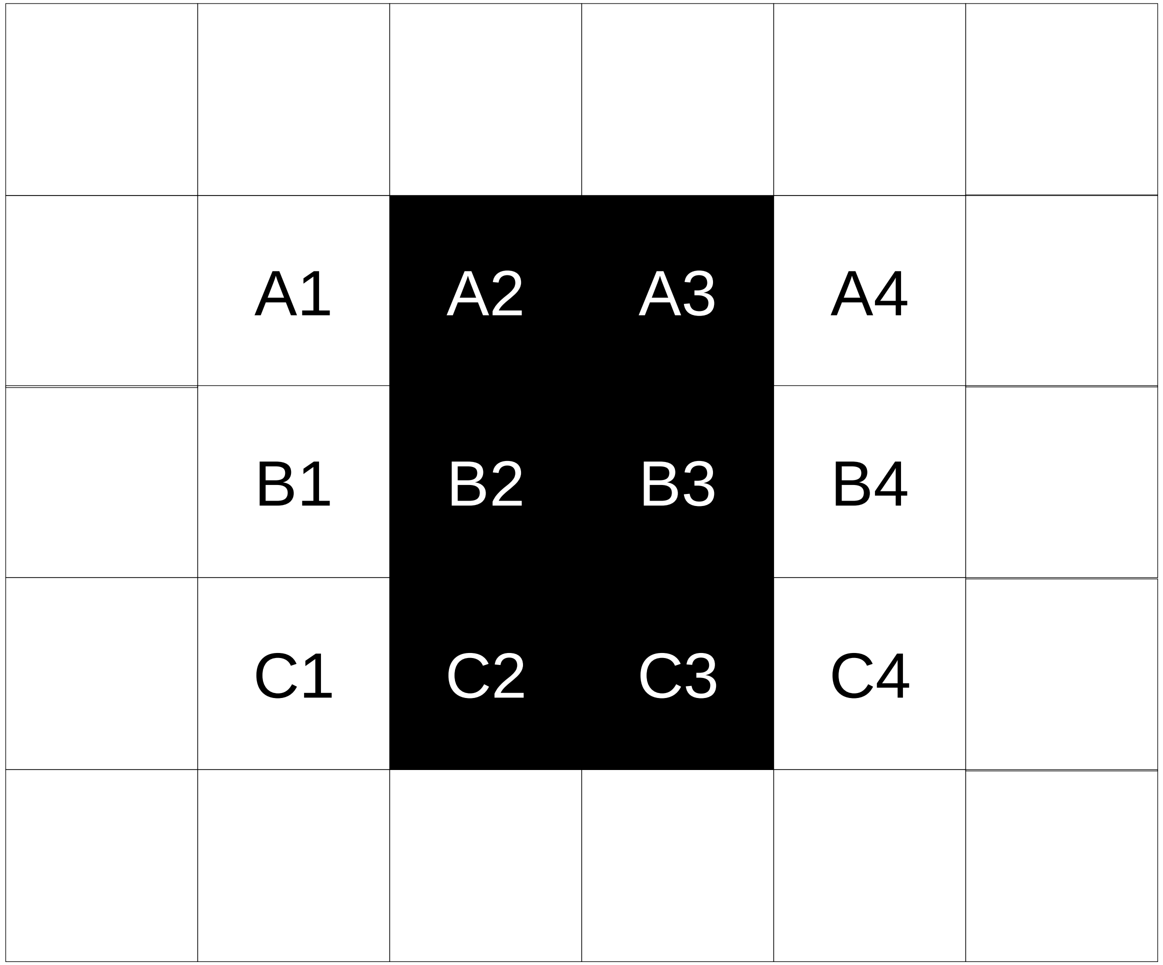
En esta nueva configuración, las celdas B1 y B4 están rodeadas por 3 celdas vivas por lo allí surgirá una nueva celda negra.

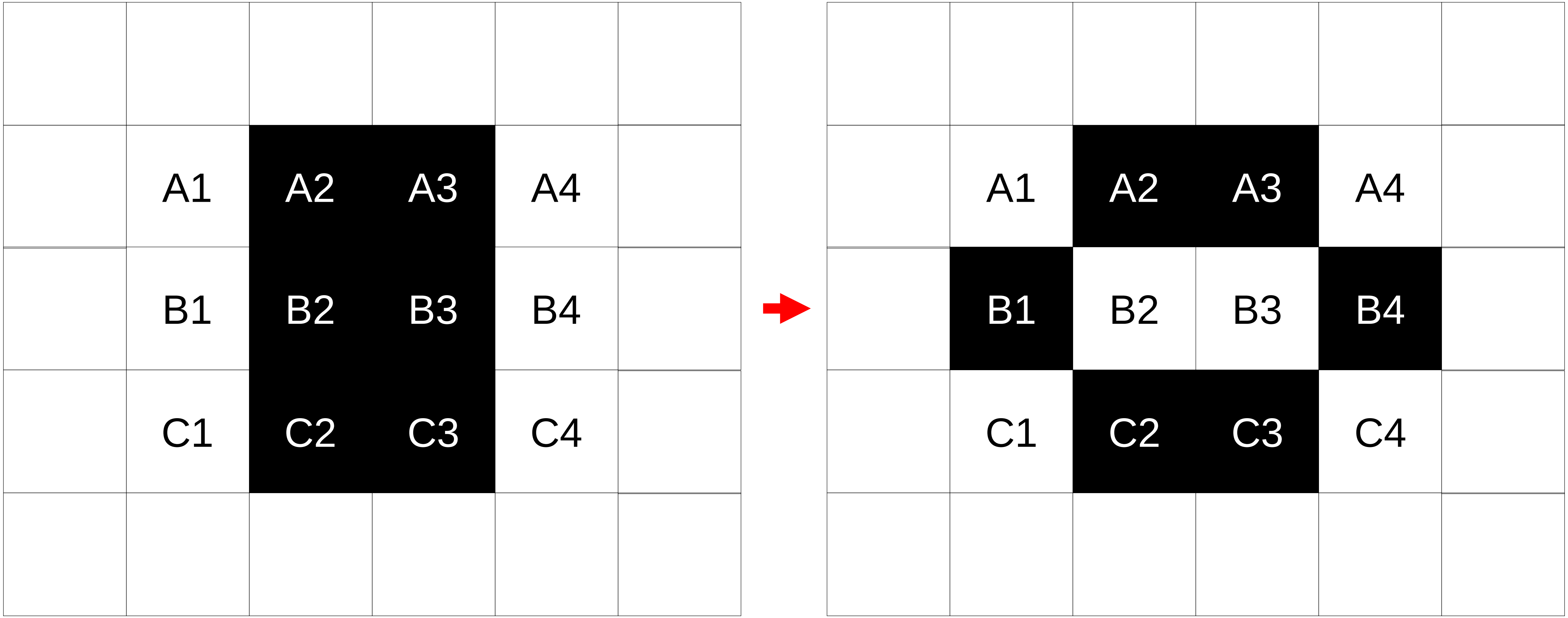
Las celdas de las esquinas (A1, A4, C1 y C4) están rodeadas por 2 celdas vivas por lo que no surgirán celdas negras.



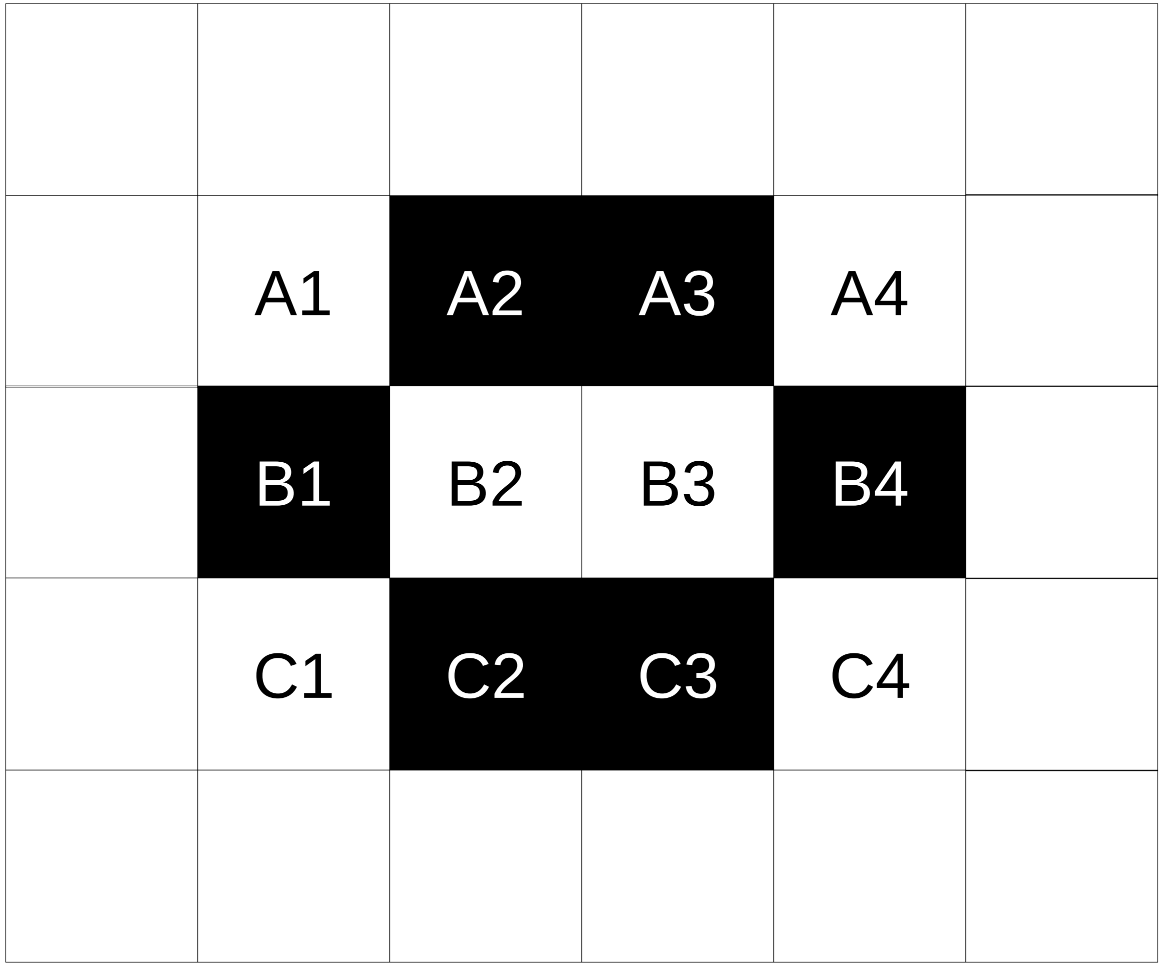
Las celdas A2, A3, C2 y C3 están rodeadas por 3 celdas vivas cada una, por lo tanto, sobrevivirán en la siguiente iteración.

Por otro lado, las celdas B2 y B3 están rodeadas por 5 celdas vivas cada una, por lo que morirán por superpoblación.



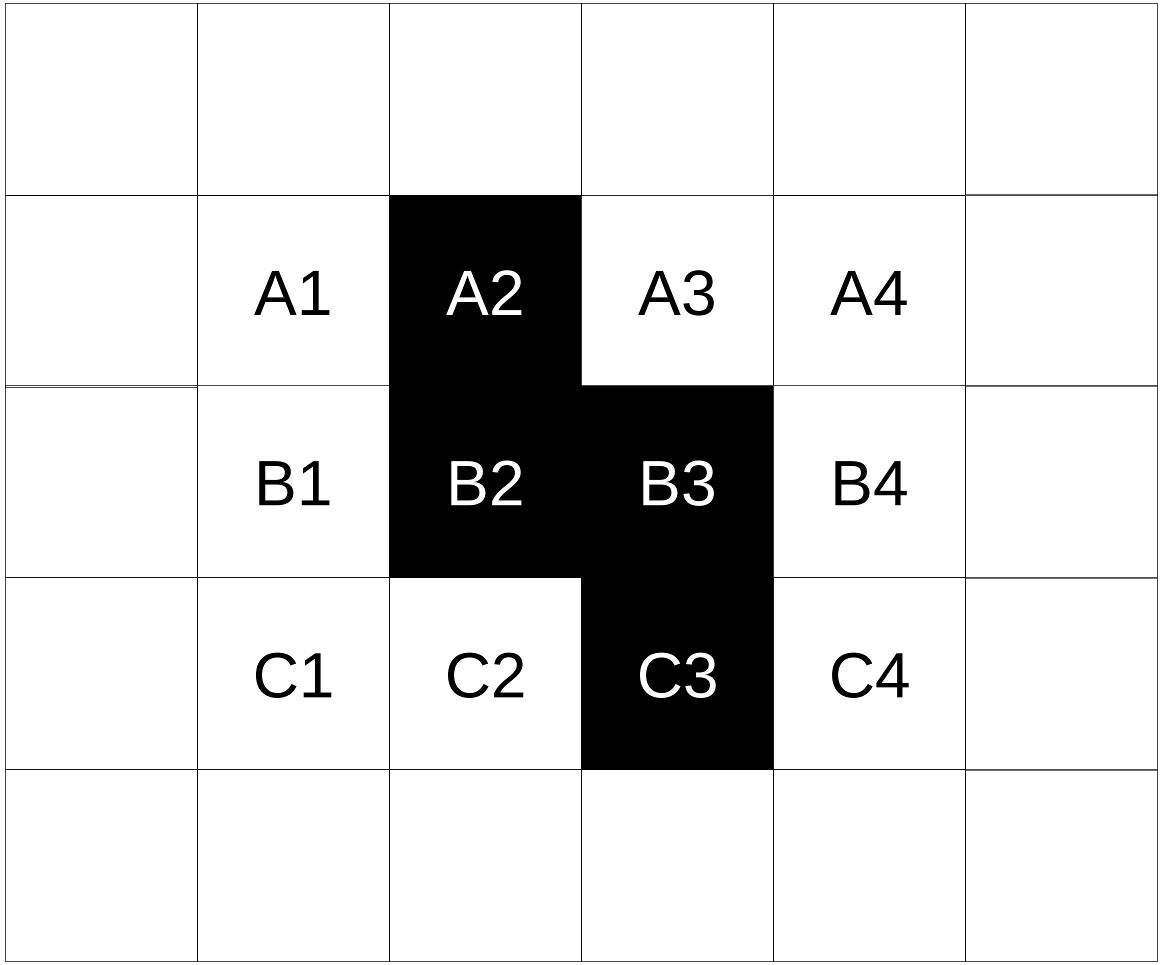

A1
A2
A3
A4
B1
B2
B3
B4
C1
C2
C3
C4
A1
A2
A3
A4
B1
B2
B3
B4
C1
C2
C3
C4

Notar que esta configuración es estable. Las celdas vivas están rodeadas por 2 celdas vivas cada una por lo que sobrevivirán. Las celdas muertas B2 y B3 están rodeadas por 5 celdas vivas por lo que no podrán nacer celdas negras. De las celdas esquina tampoco surgirán celdas negras.



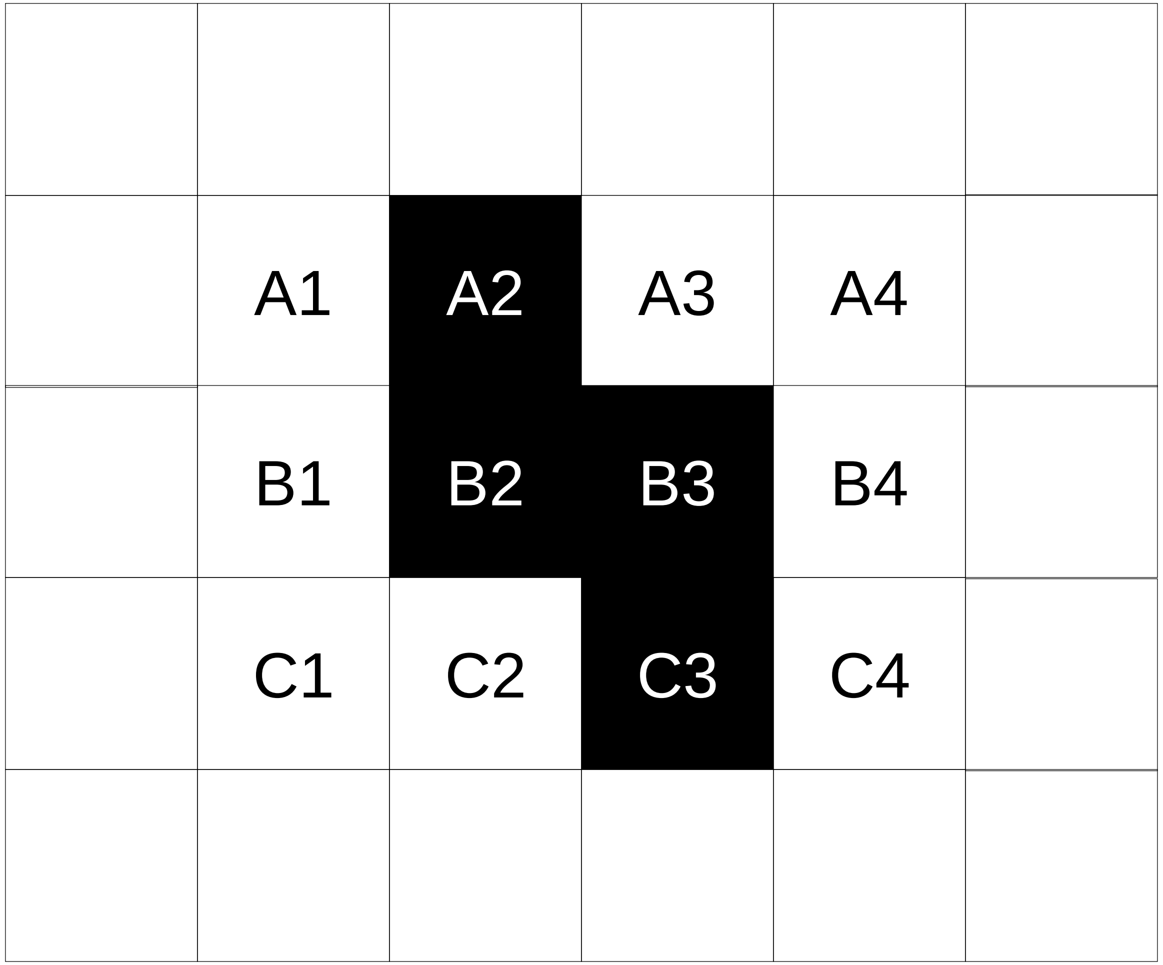
Tetrómino 1D

Para esta configuración se observa que las celdas A3 y C2 están rodeadas de 3 celdas vivas y por lo tanto allí nacerán celdas negras en la siguiente iteración.

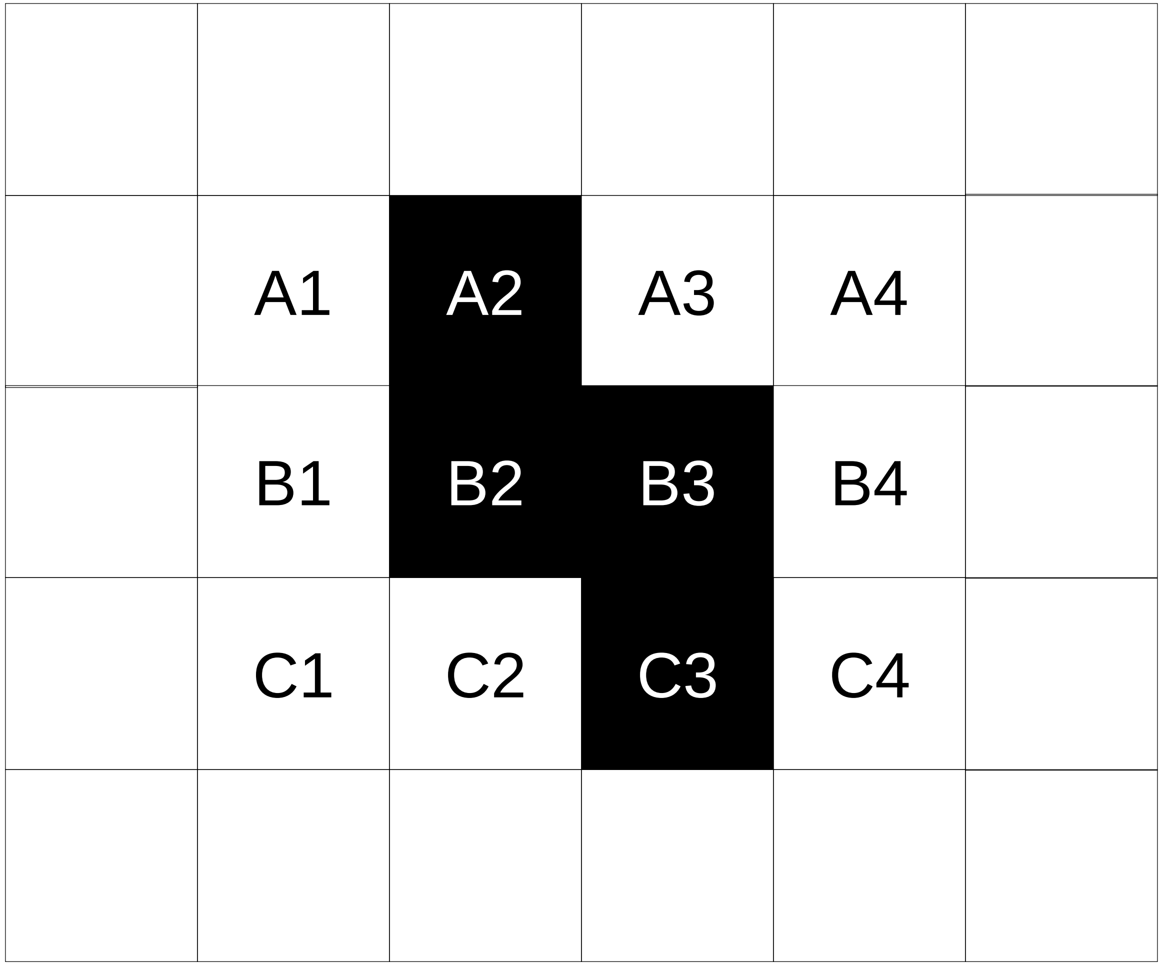


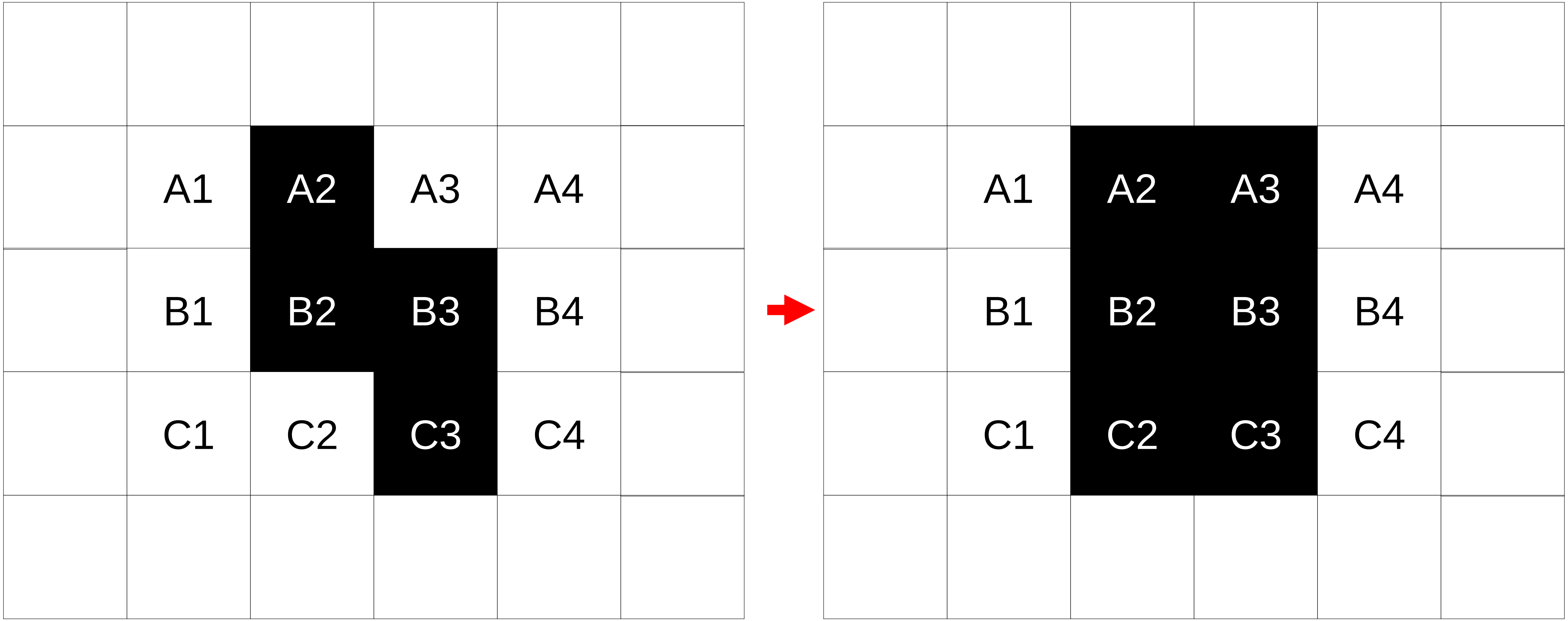
Las celdas A2 y C3 están rodeadas por 2 celdas vivas cada una por lo que sobrevivirán en la siguiente generación.

Las celdas B2 y B3 están rodeadas de 3 celdas cada una por lo que también vivirán en la siguiente iteración.



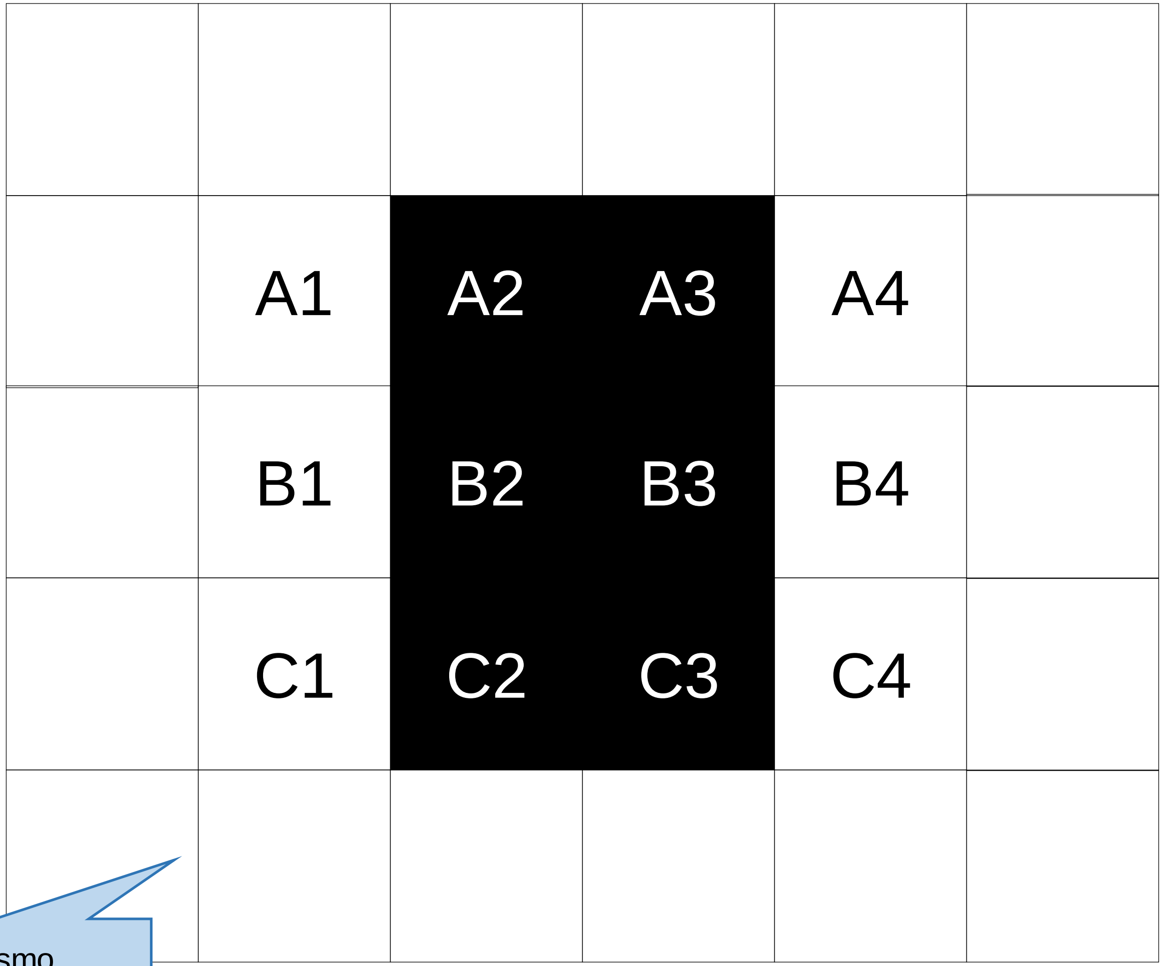
El resto de las celdas muertas (A1, A4, B1, B4, C1 y C4) están rodeadas por 2 o menos celdas vivas por lo que no formarán nuevas celdas negras en la siguiente iteración.




A1
A2
A3
A4
B1
B2
B3
B4
C1
C2
C3
C4
A1
A2
A3
A4
B1
B2
B3
B4
C1
C2
C3
C4

Notar que esta configuración es idéntica a la obtenida con el Tetrómino 1B y por lo tanto, *llegará al mismo estado de equilibrio estable*.

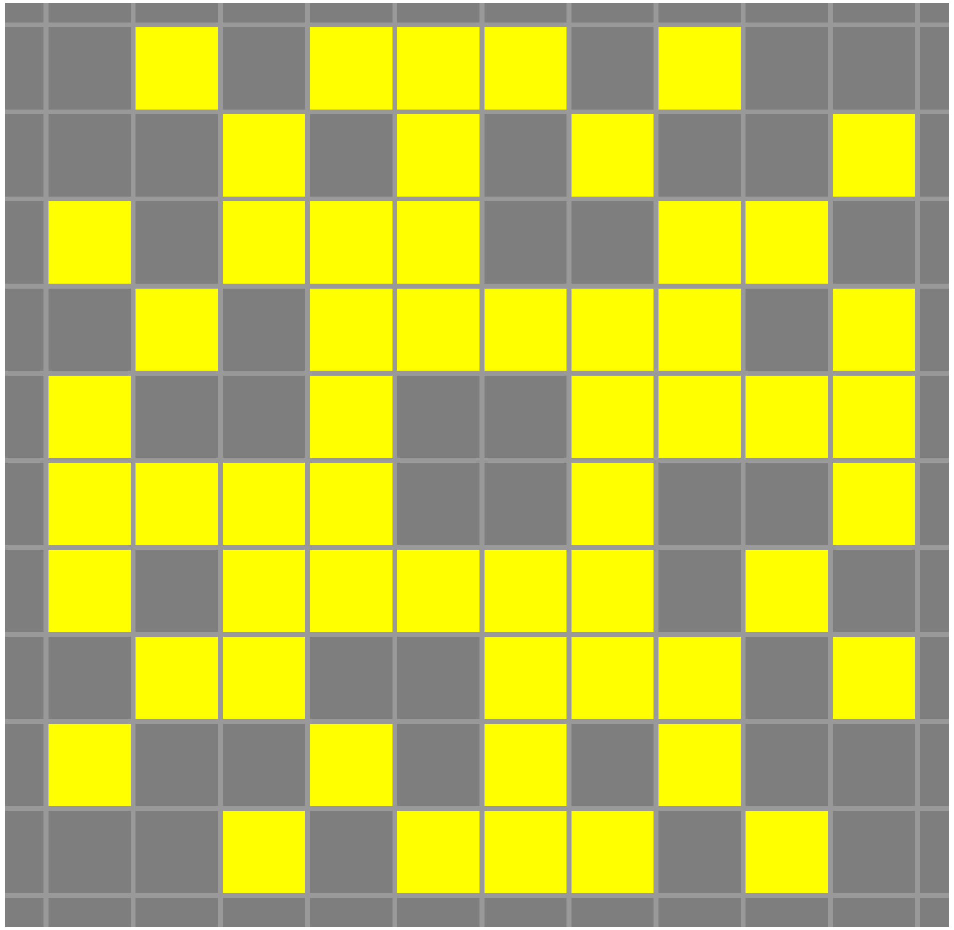
Esto implica que **diferentes configuraciones iniciales pueden tener el mismo destino**.



Si hacen el mismo análisis para el Tetrómino 1C llegarán al mismo resultado.

Así como existen configuraciones iniciales que alcanzan el mismo estado de equilibrio final, también existen configuraciones a las cuales *no es posible llegar* a partir de alguna configuración inicial.

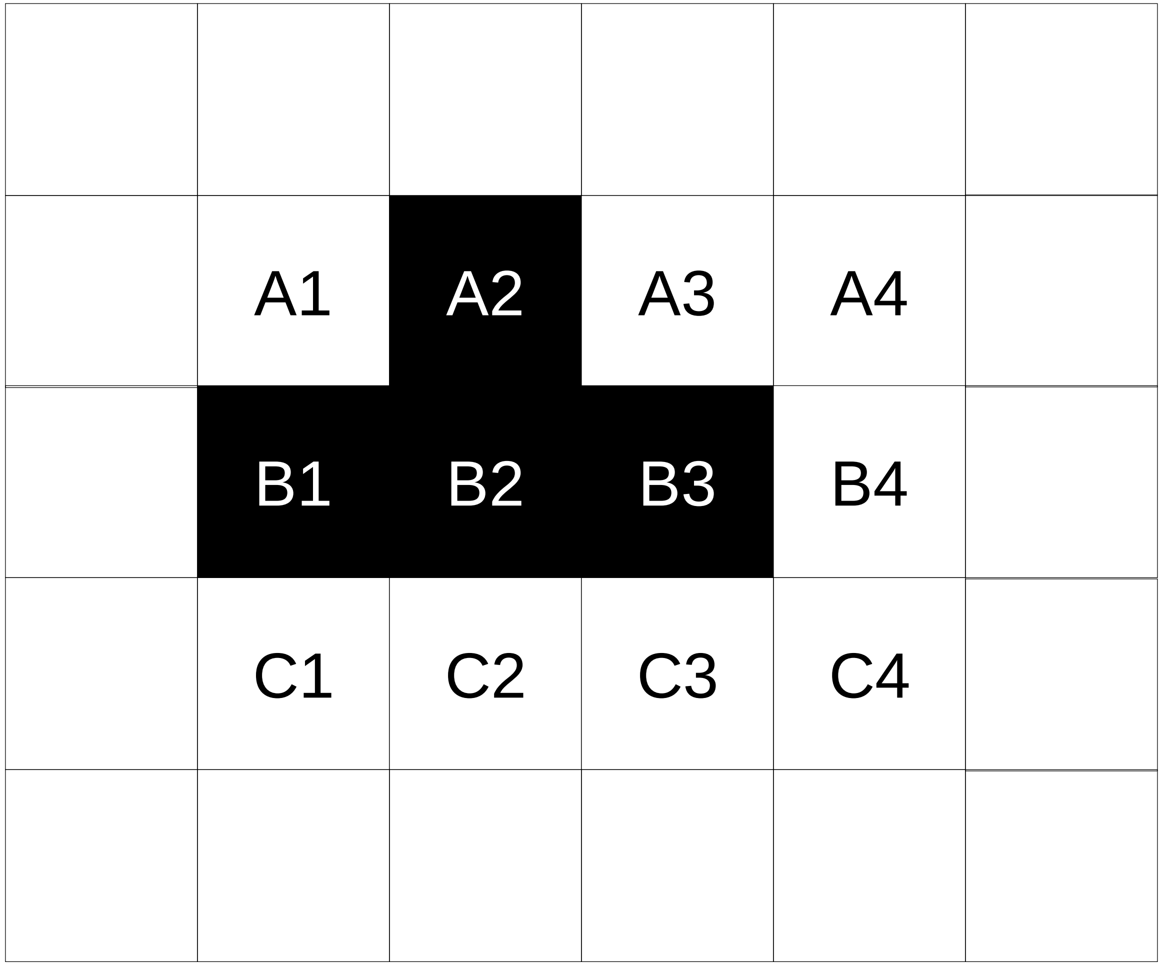
A estas configuraciones se les denomina **Jardines del Edén**.



Tetrómino 1E

Esta configuración es un ejemplo de cómo a partir de un único cambio en la configuración de una celda, se puede generar un patrón de evolución más complejo.

Compare esta estructura y su evolución con la de los Tetróminos 1C y 1D.



Pentómino

Para estudiar la evolución del pentómino ampliaremos la enumeración de la grilla.

Ahora utilizaremos una grilla de 5 filas y 6 columnas.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

Notar que la celda A2 está rodeada inicialmente de 1 sola celda viva (B3) por lo que no sobrevivirá para la siguiente iteración.

La celda B3 está rodeada de 3 celdas vivas (A2, C2 y C3) por lo que sobrevivirá para la siguiente iteración.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

La celda C1 está rodeada de 1 sola celda viva (C2) por lo que no sobrevivirá para la siguiente iteración.

Las celdas C2 y C3 están rodeadas de 2 celdas vivas cada una por lo que sobrevivirán en la siguiente iteración.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

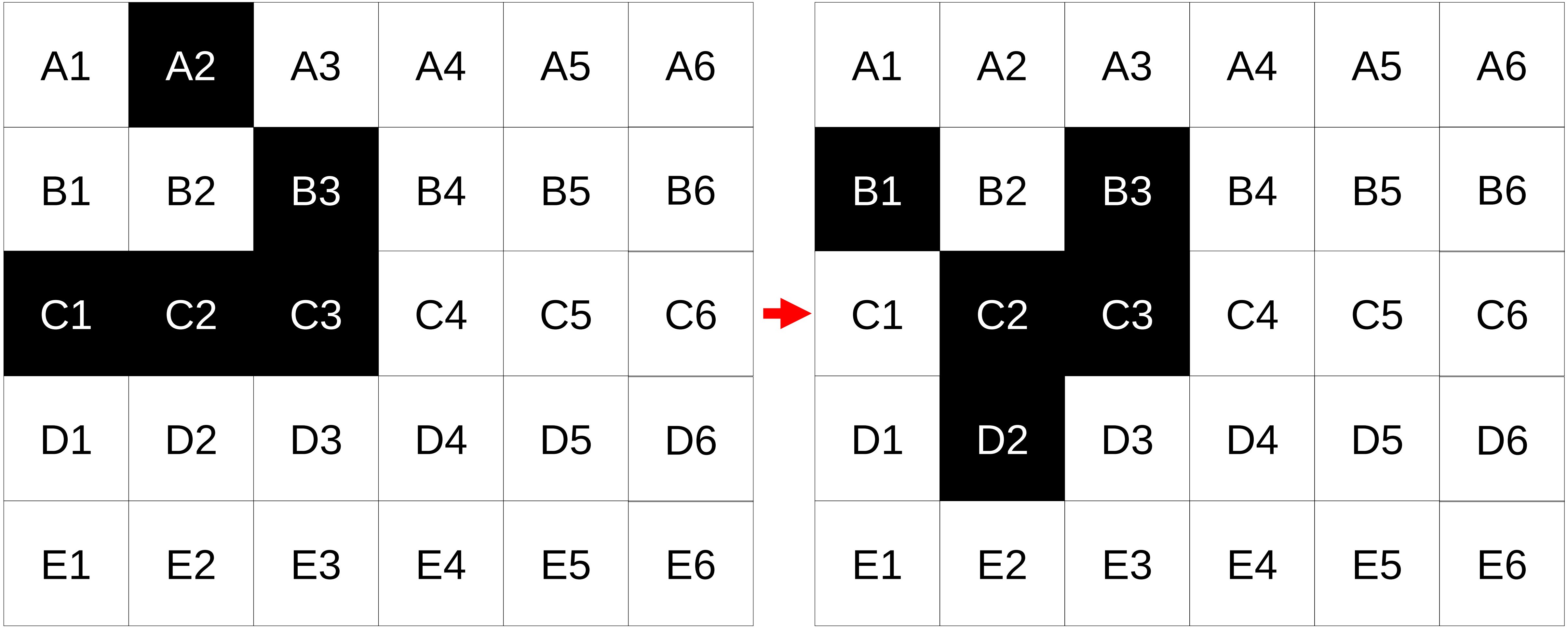
Notar que dada esta configuración inicial, las celdas B1 y D2 que están desprovistas de “vida” están rodeadas de 3 celdas vivas cada una por lo que darán lugar a una celda negra en la siguiente generación.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

En resumen:

- Las celdas A2 y C1 mueren.
- Las celdas B3, C2 y C3 sobreviven.
- Las celdas B1 y D2 nacen.
- Las demás celdas no se modifican.

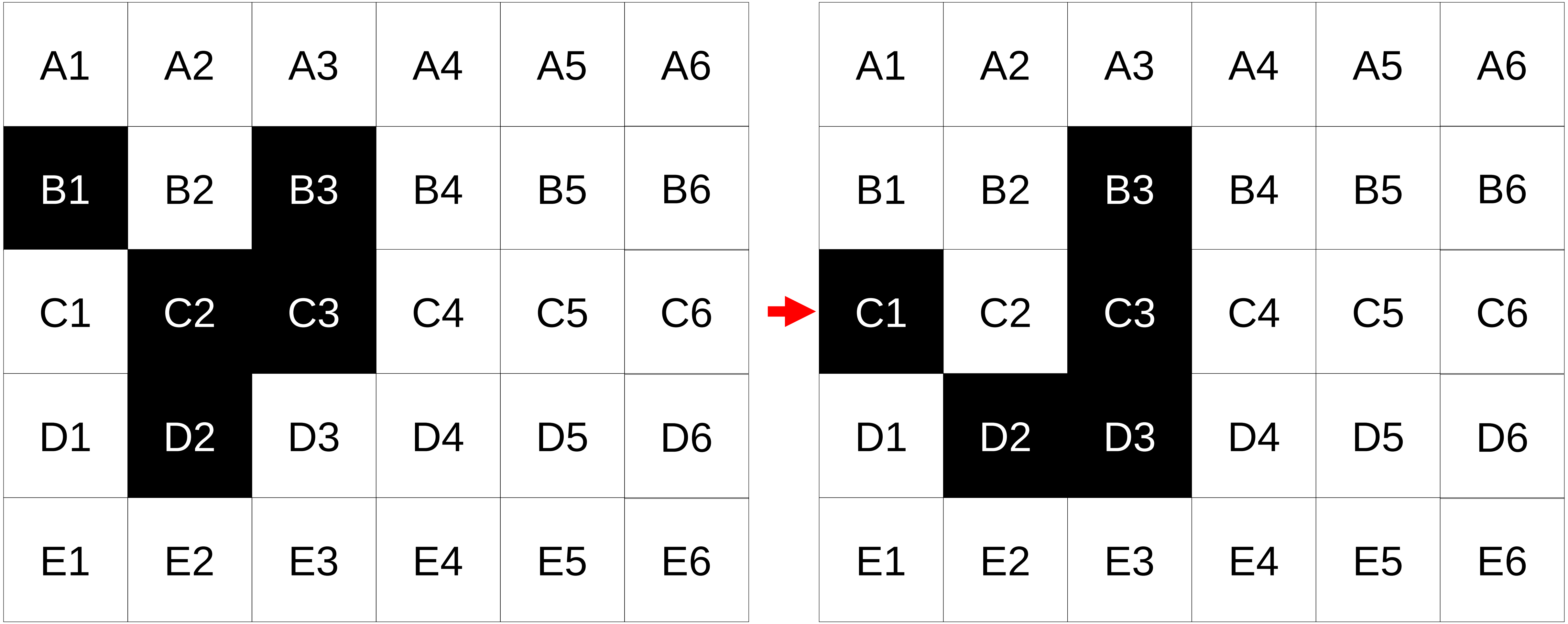
A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6


A1 A2 A3 A4 A5 A6
B1 B2 B3 B4 B5 B6
C1 C2 C3 C4 C5 C6
D1 D2 D3 D4 D5 D6
E1 E2 E3 E4 E5 E6
A1 A2 A3 A4 A5 A6
B1 B2 B3 B4 B5 B6
C1 C2 C3 C4 C5 C6
D1 D2 D3 D4 D5 D6
E1 E2 E3 E4 E5 E6

Haciendo un análisis similar para la siguiente iteración se obtiene que:

- Las celdas B1 y C2 mueren.
- Las celdas B3, C3 y D2 sobreviven.
- Las celdas C1 y D3 nacen.
- La celda B2 no nace por superpoblación.

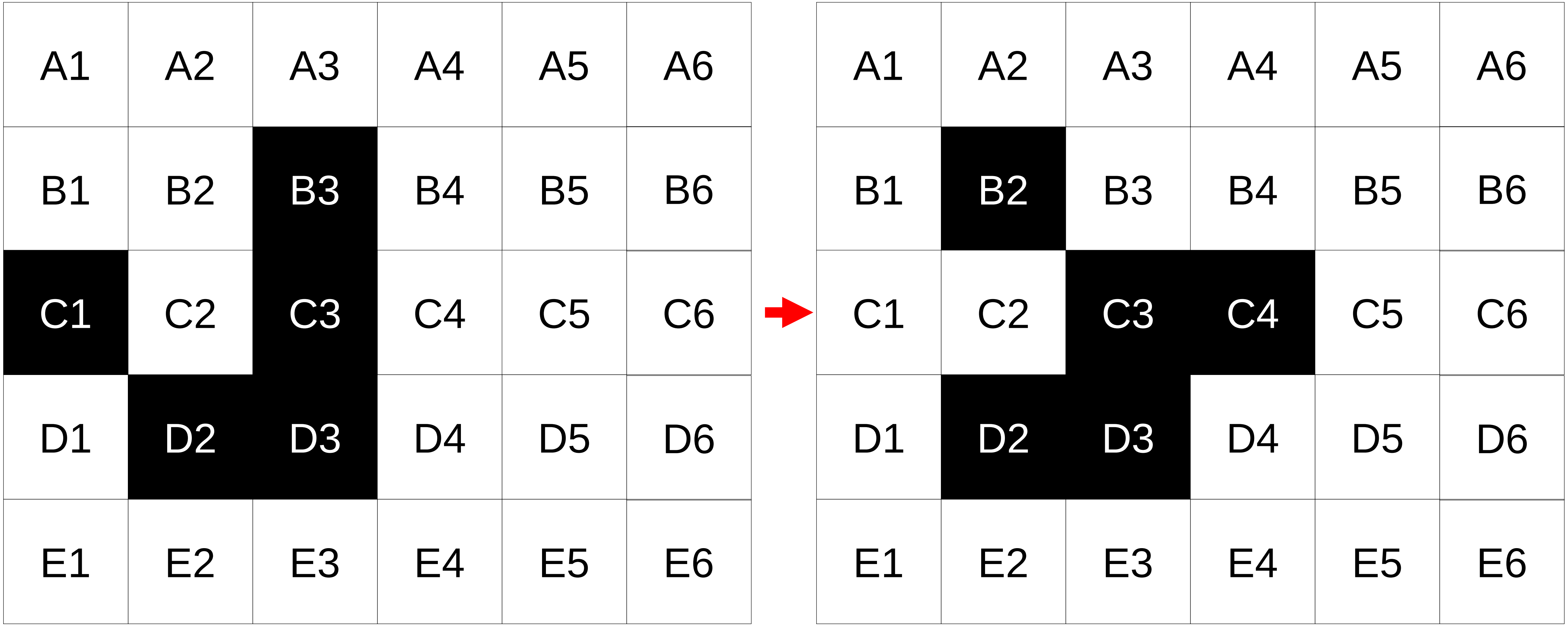
A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6


A1 A2 A3 A4 A5 A6
B1 B2 B3 B4 B5 B6
C1 C2 C3 C4 C5 C6
D1 D2 D3 D4 D5 D6
E1 E2 E3 E4 E5 E6
A1 A2 A3 A4 A5 A6
B1 B2 B3 B4 B5 B6
C1 C2 C3 C4 C5 C6
D1 D2 D3 D4 D5 D6
E1 E2 E3 E4 E5 E6

En la siguiente iteración:

- Las celdas B3 y C1 mueren por aislamiento.
- Las celdas C3, D2 y D3 sobreviven.
- Las celdas B2 y C4 nacen.
- La celda C2 no puede nacer por superpoblación.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6


A1 A2 A3 A4 A5 A6
B1 B2 B3 B4 B5 B6
C1 C2 C3 C4 C5 C6
D1 D2 D3 D4 D5 D6
E1 E2 E3 E4 E5 E6
A1 A2 A3 A4 A5 A6
B1 B2 B3 B4 B5 B6
C1 C2 C3 C4 C5 C6
D1 D2 D3 D4 D5 D6
E1 E2 E3 E4 E5 E6

En la próxima generación:

- La celda B2 muere por aislamiento y la celda C3 muere por superpoblación.
- Las celdas C4, D2 y D3 sobreviven.
- Las celdas B3 y D4 nacen.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

→

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

Notar que la configuración obtenida es exactamente la misma que la configuración inicial trasladada una fila hacia abajo y una columna hacia la derecha.

A esta estructura transportadora se le conoce como **Deslizador**.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

Uno de las estructuras deslizantes más conocidas es el **Cañón de Deslizadores de Gosper** (del inglés ‘*Gosper’s Glider Gun*’), la cual permite generar infinitos deslizadores.

A partir de los deslizadores de Gosper es posible construir y computar funciones lógicas[3].

[3] [Let’s BUILD a COMPUTER in CONWAY's GAME of LIFE ⠠⠵ - YouTube]

La evolución temporal del Pentómino R es bastante compleja. Luego de más de 1700 iteraciones se observan estructuras residuales estables y algunos deslizadores, cuadrados de 4x4, estructuras como las que forman los Tetróminos 1C o 1D, osciladores como el del Ejercicio 1E.

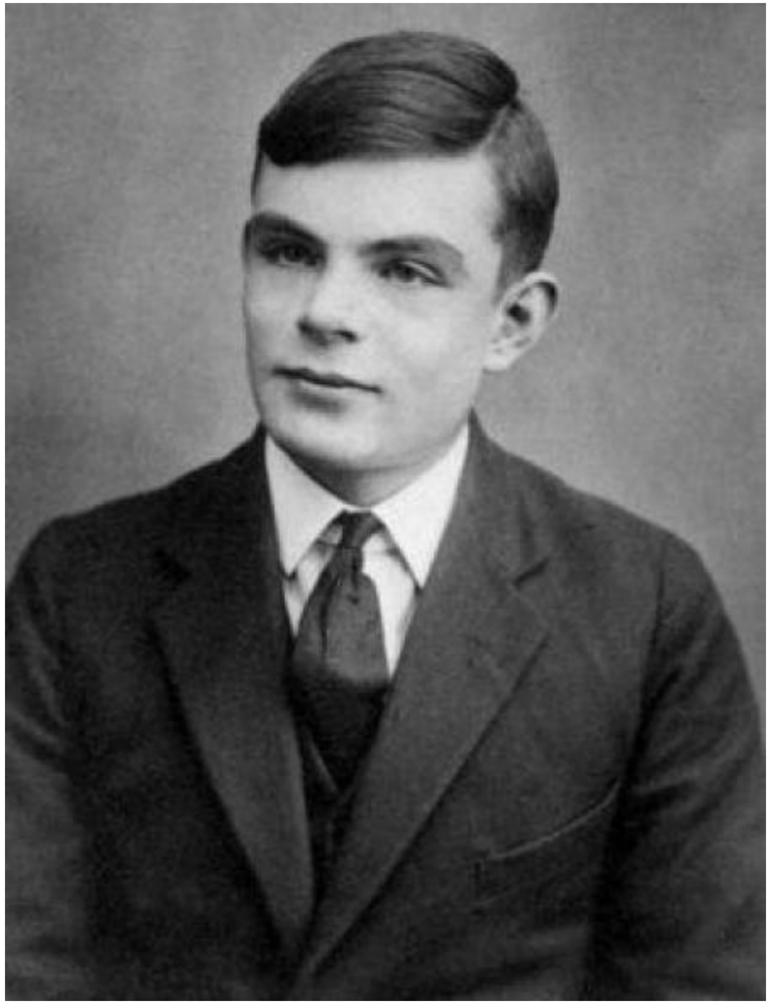
A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

Este comportamiento no es posible de predecir a partir de la configuración inicial.

Compare esta estructura con otros pentóminos y tetróminos trabajados. La sola adición o cambio inicial de una celda puede generar una evolución mucho más compleja que estructuras similares.

A1	A2	A3	A4	A5	A6
B1	B2	B3	B4	B5	B6
C1	C2	C3	C4	C5	C6
D1	D2	D3	D4	D5	D6
E1	E2	E3	E4	E5	E6

Una de las características más notables del Juego de la Vida de Conway es que con la adecuada configuración inicial es posible realizar cómputos y cálculos complejos. Esto la transforma en una **Máquina de Turing**, es decir, una máquina de cómputo universal, capaz, entre otras cosas, de emularse a sí mismas (tal y como van Neumann propuso en sus *autómatas autoreplicantes*).



El Juego de la Vida de Conway es un contraejemplo notable del Determinismo Científico, el cual plantea que, de saberse todas las reglas que rigen en el Universo sería posible predecir su evolución tanto hacia pasado como hacia el futuro. El rol del error y el azar en la búsqueda de las leyes universales es, por tanto, aparente, irrelevante y ajeno a la naturaleza del Universo.

¿Qué ejemplos de teorías sobre el Universo sabemos que no son deterministas?

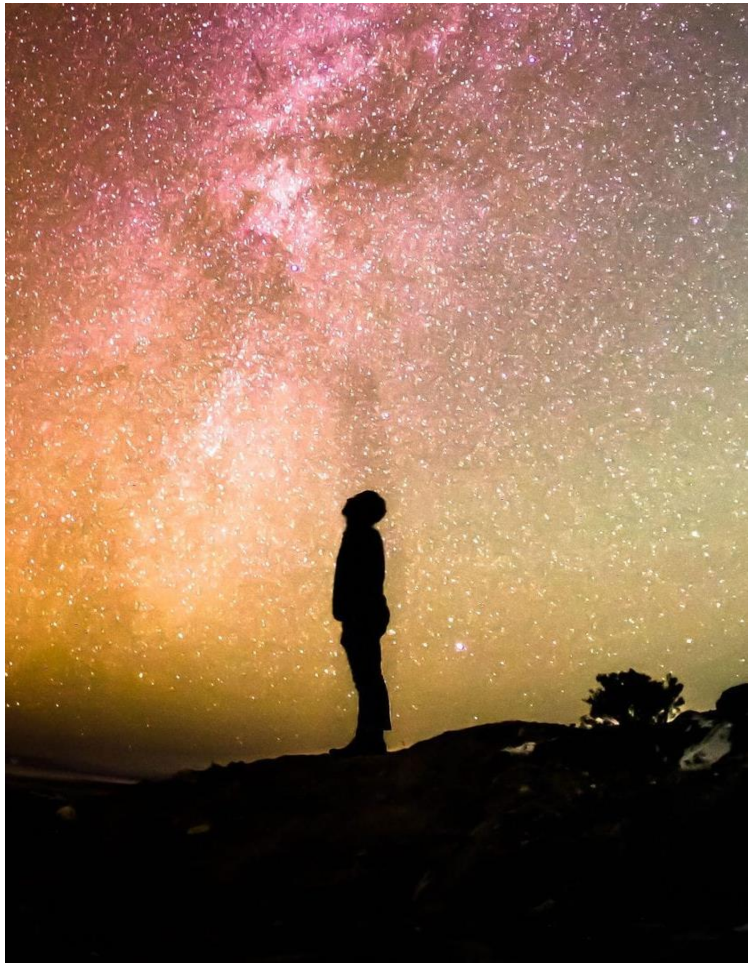
“*Hemos de considerar el estado actual del Universo como el efecto de su estado anterior y como la causa del que ha de seguirle. Una inteligencia que en un momento determinado conociera todas las fuerzas que animan a la naturaleza, así como la situación respectiva de los seres que la componen, si además fuera lo suficientemente amplia como para someter a análisis tales datos, podría abarcar en una sola fórmula los movimientos de los cuerpos más grandes del Universo y los del átomo más ligero; nada le resultaría incierto y tanto el futuro como el pasado estarían presentes ante sus ojos*.” – Pierre-Simon, marqués de Laplace (1814)[4]

[4] de Laplace, P. S. (1814), Ensayo filosófico sobre las posibilidades, p. 25, Ediciones Altaya. Traducción, introducción y notas: Pilar Castillo. Barcelona. 1995. Colección Grandes obras del pensamiento.

¿Cómo es posible que un autómata celular sencillo pueda tener un comportamiento impredecible?

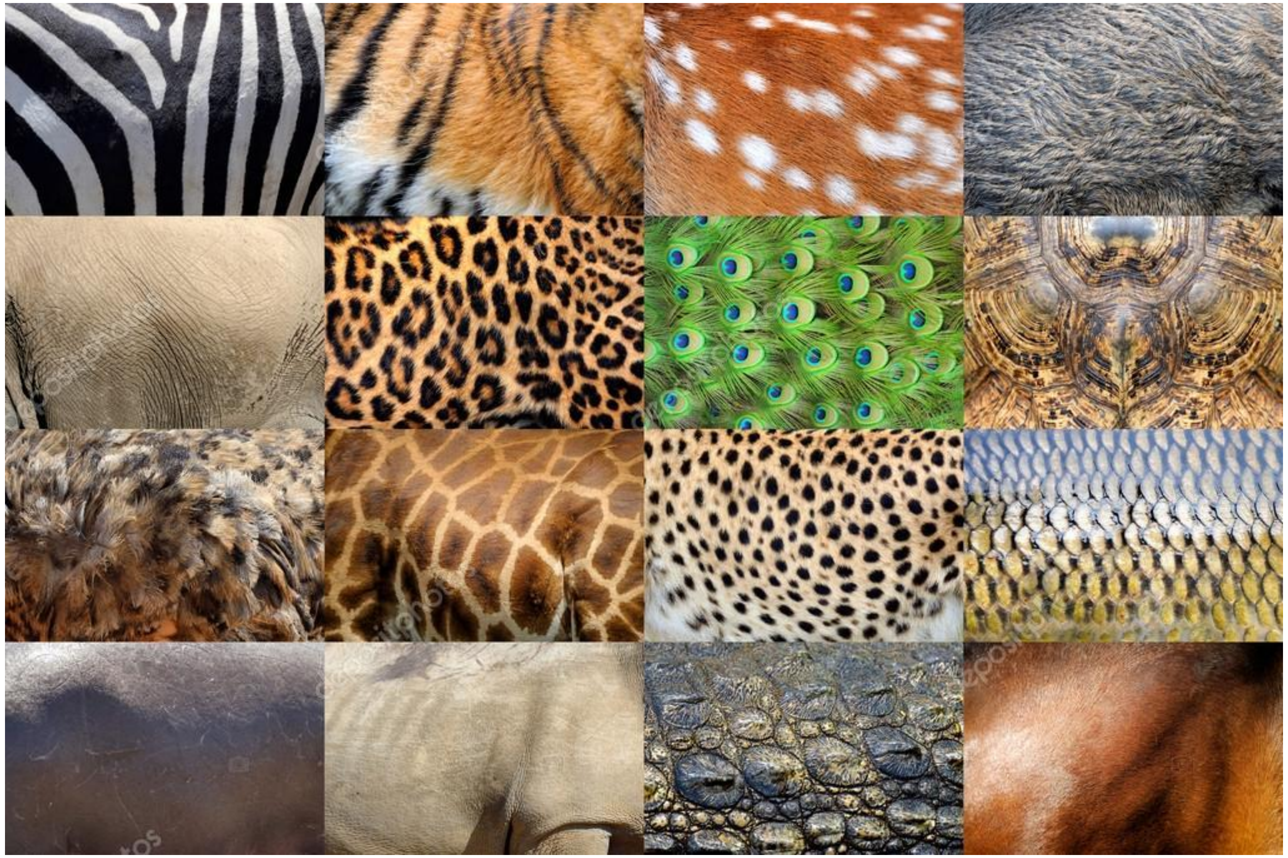
¿Podría el Universo en el que nos encontramos ser emulado por un autómata tipo-*Life*?

¿Podría un organismo entero, o la misma mente, ser emulado por un autómata tipo-*Life*?



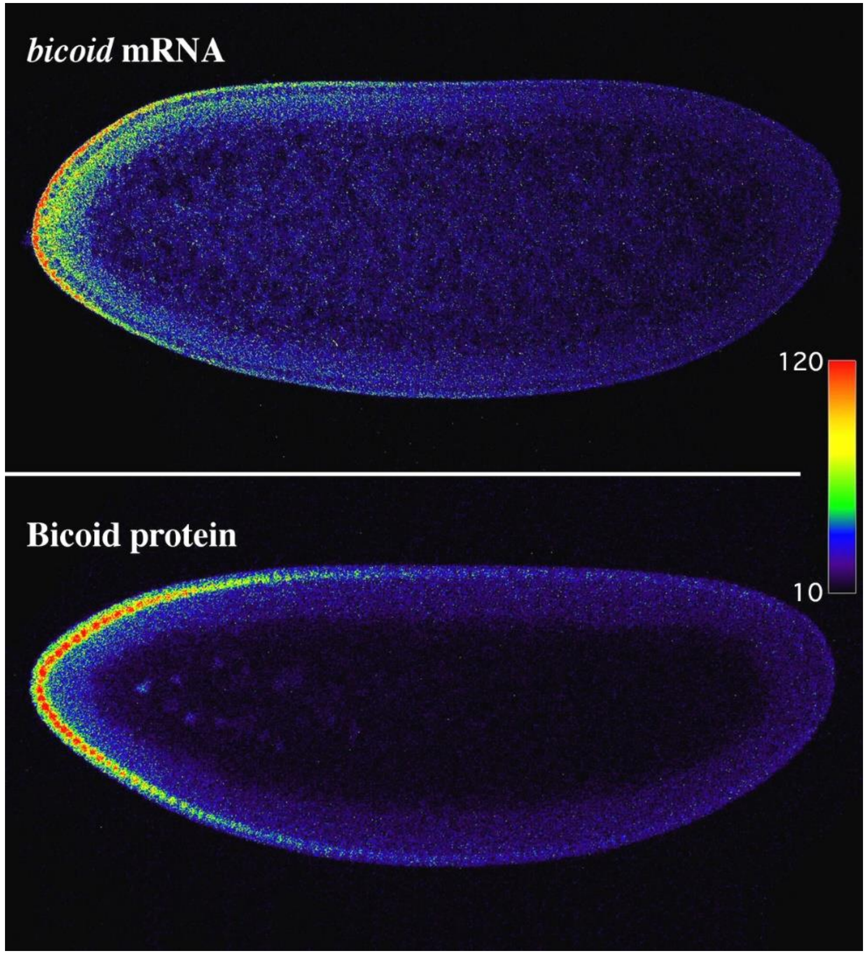
Game of Life y la Morfogénesis

¿Cómo puede relacionarse la aparición de complejidad y patrones impredecibles a partir de configuraciones y reglas sencillas en '*Life*' con la Morfogénesis Biológica?

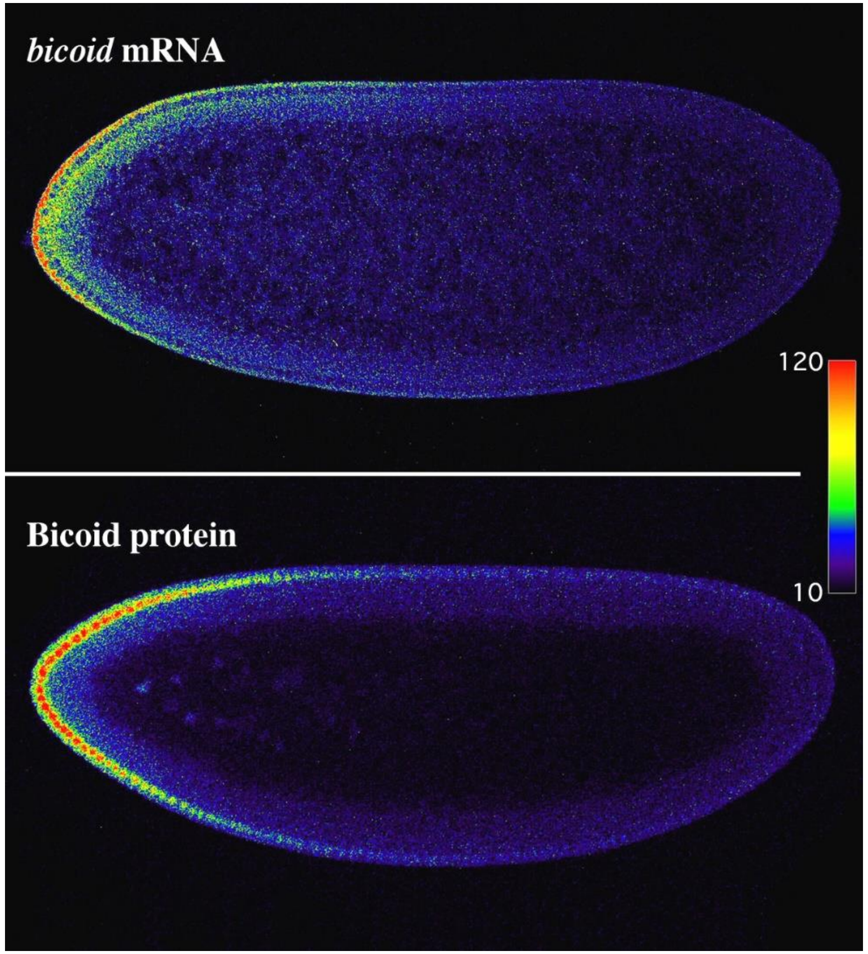


La **Morfogénesis** es el conjunto de procesos biológicos que llevan a que un organismo desarrolle su forma.

La aparición de *patrones espacio-temporales* y la alta diversidad y complejidad observada en la naturaleza es producto de *interacciones locales* entre células a partir de reglas establecidas por el *genoma* de los organismos.



Podemos imaginarnos cada célula de un organismo como una celda de ‘*Life*’, cada una con un estado asociado a una etapa del desarrollo del organismo (Ej.: el gradiente de concentración de un **morfógeno**[5]). Las conexiones locales entre las células y las reglas establecidas por el genoma darán lugar a una evolución temporal (no discreta) de dicho organismo.



[5] Spirov, A. et al. (2009). Formation of the bicoid morphogen gradient: an mRNA gradient dictates the protein gradient. *Development* **136**, 605-614.

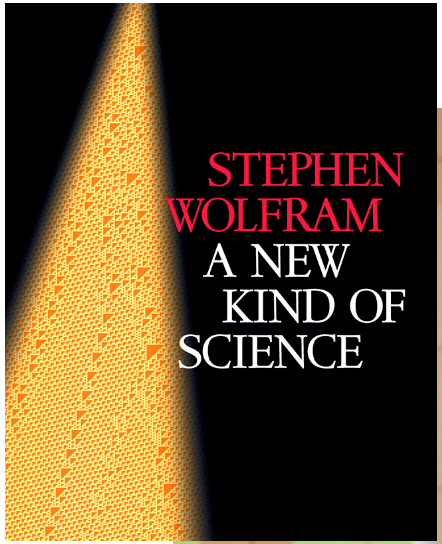
Contenido de la clase

- Introducción
- Autómata de Conway: Game of Life
- Autómata de Wolfram

Contenido de la clase

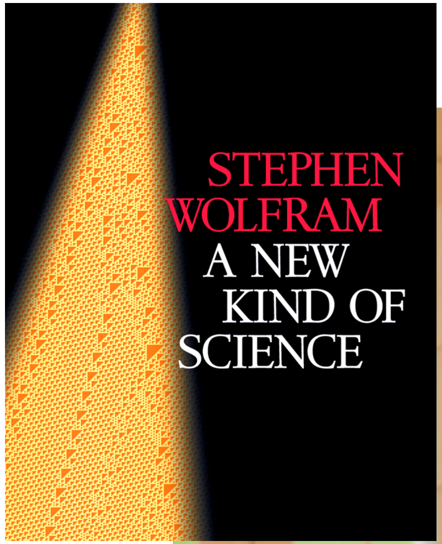
- Introducción
- Autómata de Conway: Game of Life
- **Autómata de Wolfram**

En 1986, Stephen Wolfram publica su autómata celular *unidimensional* compuesto de una única hilera horizontal de celdas, las cuales pueden tener solamente 2 estados. La evolución de las celdas en la siguiente iteración depende únicamente de su estado propio y los estados de su vecindad inmediata (adyacente).



Los estados de cada celda pueden ser simbolizados mediante 0s y 1s (“vivo” o “muerto”, “verdadero” o “falso”). Esta denominación permite computar sencillamente funciones lógicas mediante **Funciones Booleanas**.

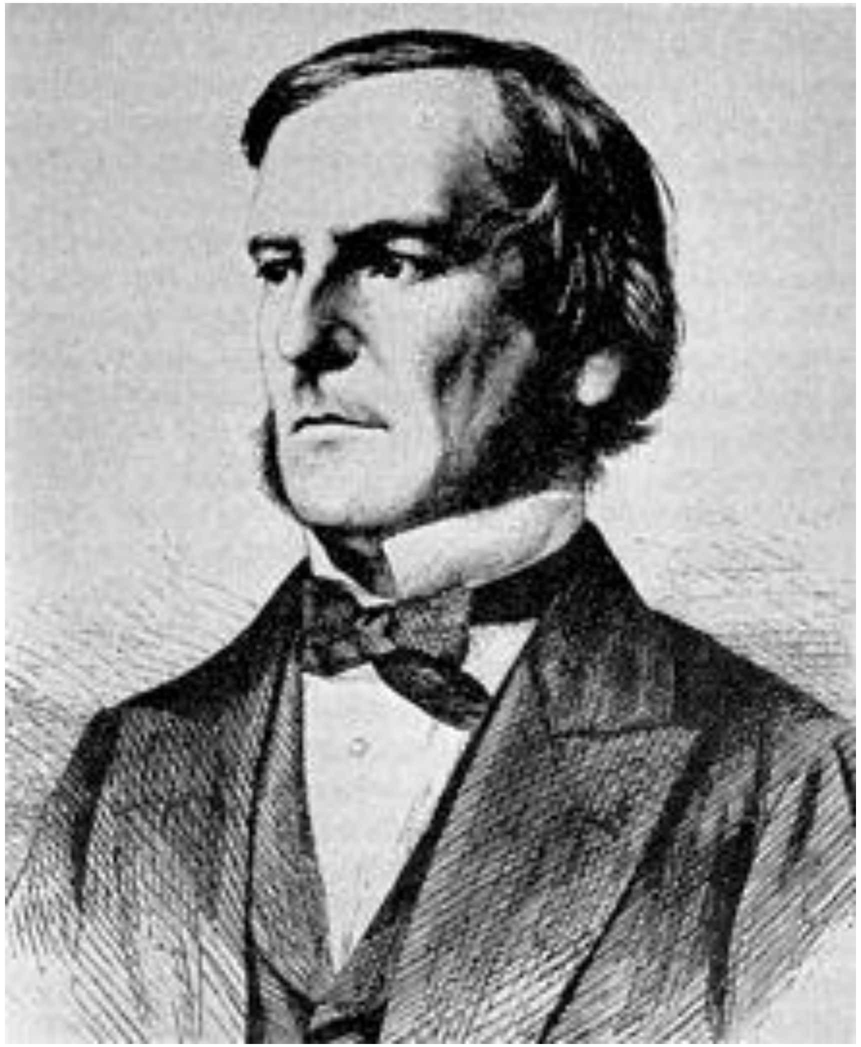
Los patrones obtenidos en este autómata son evoluciones temporales de la hilera inicial.



¿Qué ejemplos conocen de patrones temporales simbolizados como patrones espaciales?

Una función es Booleana si a partir de un conjunto preimagen en el dominio $\{0,1\}$ existe un conjunto imagen en el codominio $\{0,1\}$.

Estas funciones llevan el nombre en honor a **George Boole**, quien desarrolló toda una rama del álgebra y la lógica basado en estas funciones.



A partir de una entrada de dos elementos de valores 0 y 1, todas las combinaciones posibles que se pueden dar en esta entrada son: $\{(0,0),(0,1),(1,0),(1,1)\}$.

Las cuatro funciones posibles asociadas a dos entradas son:

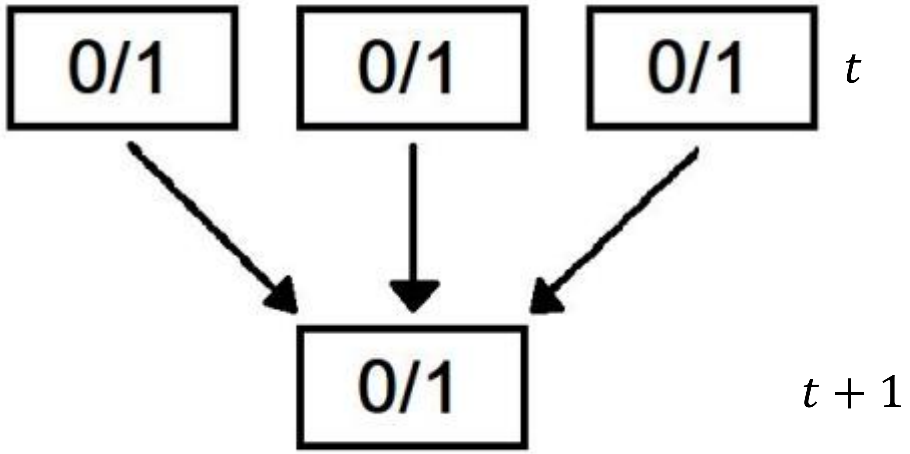
1) Función 1: Tomar el valor 0 independientemente del valor de entrada.
2) Función 2: Tomar el valor de la entrada.
3) Función 3: Tomar el valor opuesto al de entrada.
4) Función 4: Tomar el valor 1 independientemente del valor de entrada.

Volveremos sobre esto en el próximo práctico de “*Redes Neuronales*”.

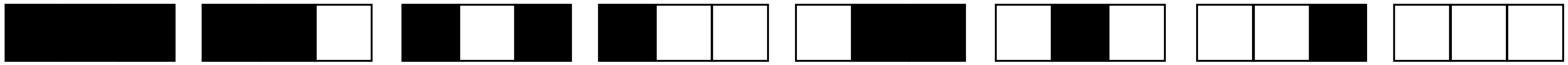
De esta forma podemos construir una tabla con todas las posibles respuestas a partir de todos los valores posibles de dos entradas.

Entrada	**Función 1 (Todo 0)**	**Función 2 (Igual)**	**Función 3 (Opuesto)**	**Función 4 (Todo 1)**
{0,0}	{0,0}	{0,0}	{1,1}	{1,1}
{0,1}	{0,0}	{0,1}	{1,0}	{1,1}
{1,0}	{0,0}	{1,0}	{0,1}	{1,1}
{1,1}	{0,0}	{1,1}	{0,0}	{1,1}

El autómata de Wolfram está construido de forma tal que la vecindad de una celda son las dos celdas adyacentes (izquierda y derecha), por lo que la celda en el tiempo $t + 1$ depende de los estados de las tres celdas superiores (tiempo t). **Podemos considerar a dicha celda como el resultado de una función booleana de tres entradas**.

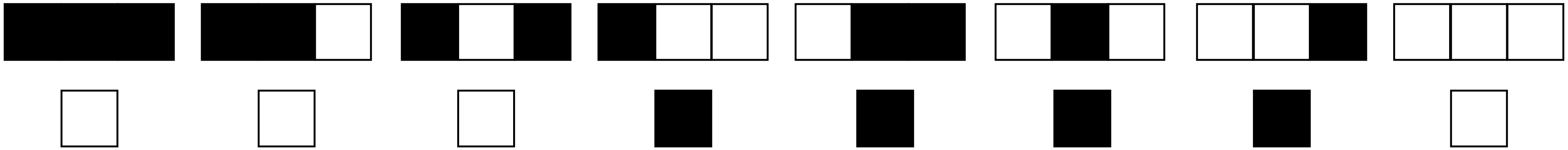


Si representamos con negro a las celdas con valor ‘1’ y en blanco a las celdas con valor ‘0’, podemos escribir las 8 combinaciones posibles de celdas.



Puesto que existen 8 combinaciones posibles de entradas, y cada celda puede tomar solo 2 valores (0 o 1), existen $2^8 = 256$ funciones booleanas capaces de ser computables. Cada una de estas funciones son las **Reglas del Autómata de Wolfram**.

Por ejemplo, la Regla 30 implica que a partir del conjunto de celdas entrada en el tiempo t se obtiene la siguiente hilera en el tiempo $t + 1$:



De acuerdo con la nomenclatura de negro (1) y blanco (0), la respuesta de la función (Regla 30) aplicada es:

$$\mathbf{0\ 0\ 0\ 1\ 1\ 1\ 1\ 0}$$

Notar que la codificación de la respuesta es un **número binario**. Dicho número binario representa el número de la regla (30). Realizando la conversión a **base diez**:

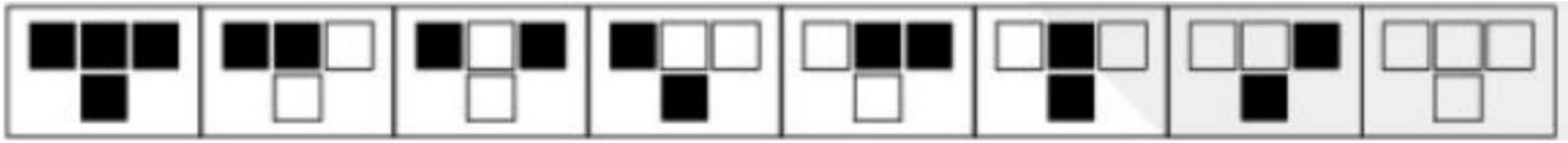
$$0\ 0\ 0\ 1\ 1\ 1\ 1\ 0$$

$$(0 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$$

$$0 + 0 + 0 + 16 + 8 + 4 + 2 + 0 = 30$$

En el nombre de la Regla está contenida la función que ejecuta.

Ejercicio 3: Parte a)



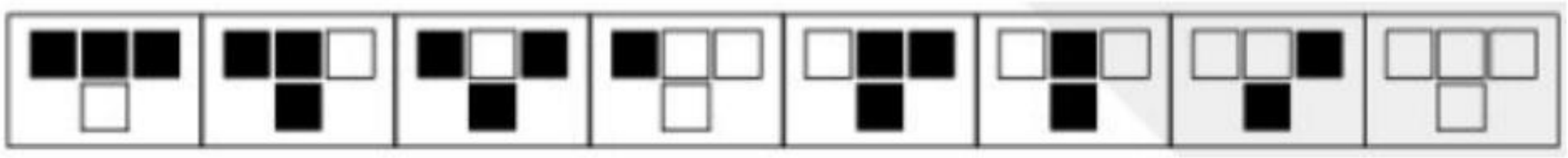
La regla que buscamos es:

$$1\ 0\ 0\ 1\ 0\ 1\ 1\ 0$$

$$(1 \times 2^7) + (0 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$$

$$128 + 0 + 0 + 16 + 0 + 4 + 2 + 0 = 150$$

Ejercicio 3: Parte b)



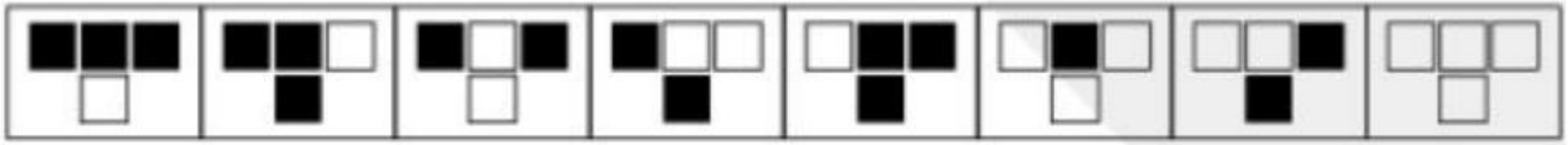
La regla que buscamos es:

$$0\ 1\ 1\ 0\ 1\ 1\ 1\ 0$$

$$(0 \times 2^7) + (1 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (1 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$$

$$0 + 64 + 32 + 0 + 8 + 4 + 2 + 0 = 110$$

Ejercicio 3: Parte c)



La regla que buscamos es:

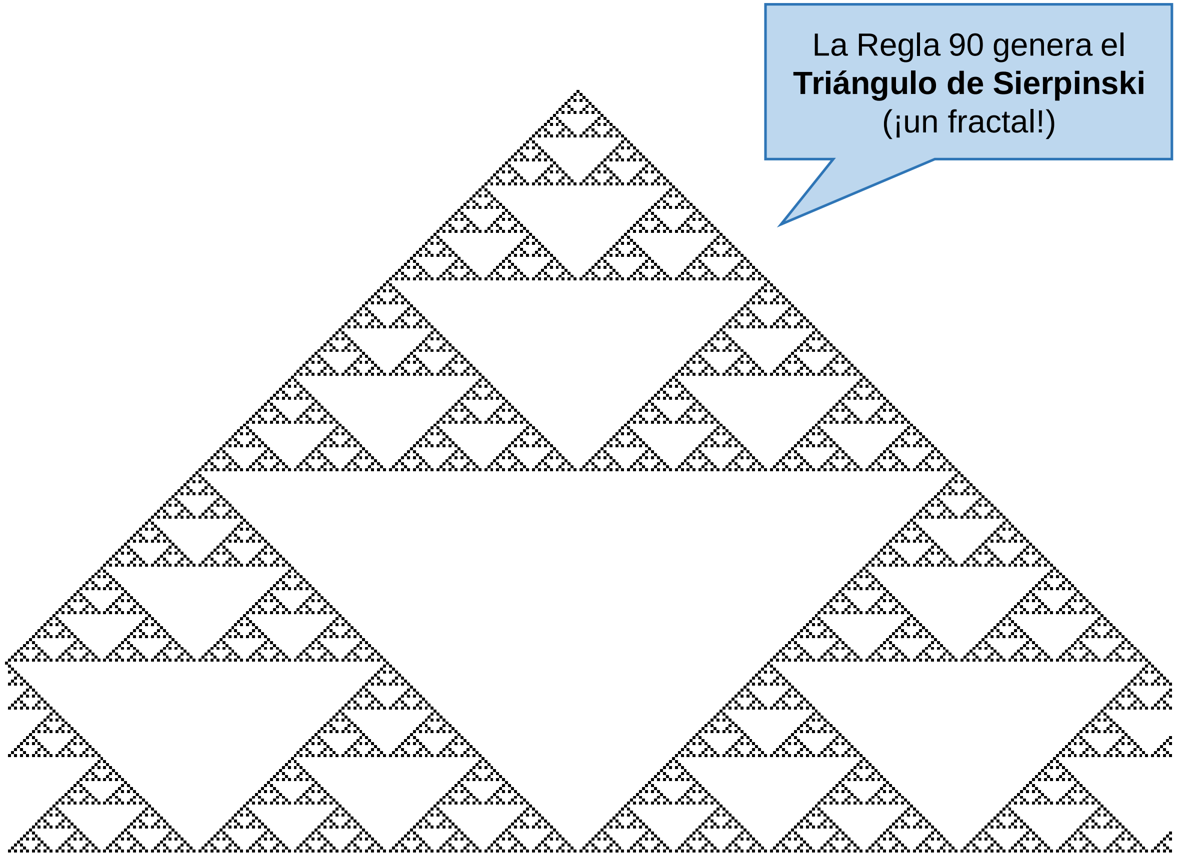
$$0\ 1\ 0\ 1\ 1\ 0\ 1\ 0$$

$$(0 \times 2^7) + (1 \times 2^6) + (0 \times 2^5) + (1 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (0 \times 2^0)$$

$$0 + 64 + 0 + 16 + 8 + 0 + 2 + 0 = 90$$

A partir de una regla establecida podemos obtener la evolución temporal de la hilera inicial[6].

Dicha evolución temporal se observa como hileras apiladas hacia abajo, cada una representando una nueva iteración.

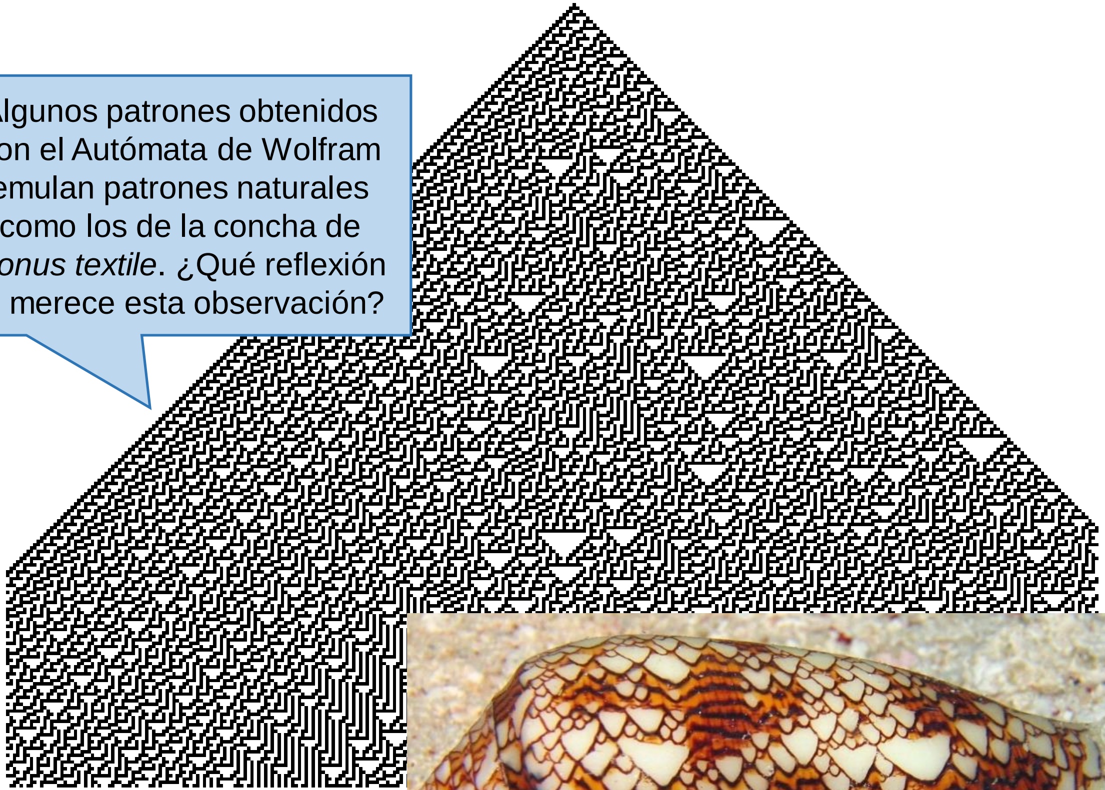


[6] Utilicen este link: [Elementary cellular automaton simulator (devinacker.github.io)]

Algunas reglas tienen comportamientos complejos, mientras que otras alcanzan rápidamente un estado de equilibrio.

Wolfram clasificó las 256 respuestas en **cuatro clases** según su complejidad.

Algunos patrones obtenidos con el Autómata de Wolfram emulan patrones naturales como los de la concha de *Conus textile*. ¿Qué reflexión le merece esta observación?



Las estructuras de **Clase 1** son aquellas cuyas configuraciones iniciales dan lugar a un estado final uniforme, generalmente en pocas iteraciones.

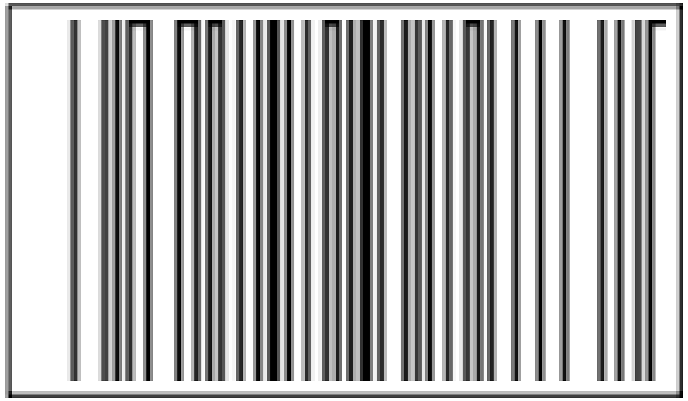
Algunas reglas de Clase 1 son: 0, 32, 128, 254.

rule 128

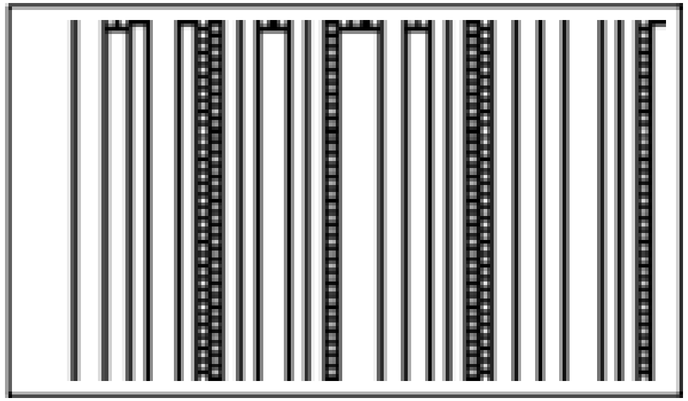
rule 254

Las estructuras de **Clase 2** son aquellas cuyas configuraciones iniciales dan lugar a un patrón **uniforme o cíclico**.

Algunas reglas de Clase 2 son: 76, 108, 204, 236.



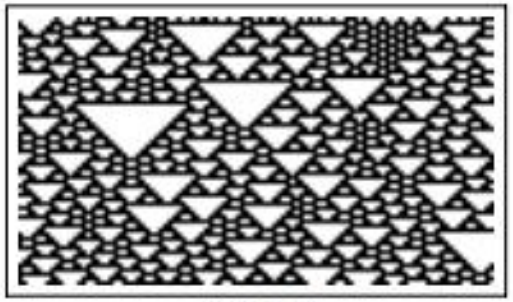
rule 76



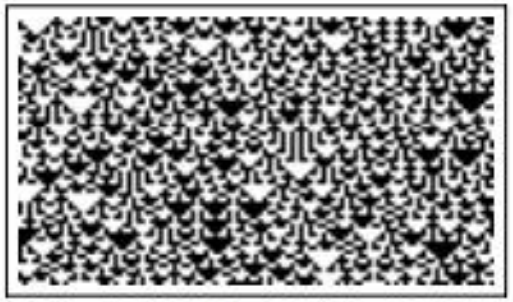
rule 108

Las estructuras de **Clase 3** son aquellas cuyas configuraciones iniciales dan lugar a un patrón **pseudoaleatorio** con algunas regularidades (Ej.: triángulos).

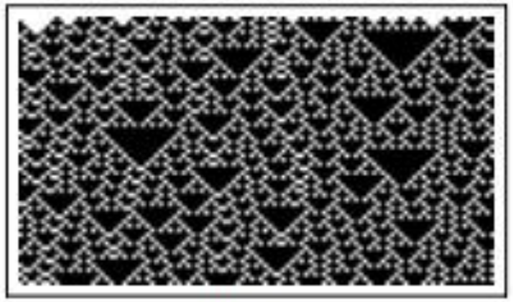
Algunas reglas de Clase 3 son: 22, 54,122,150.



rule 126



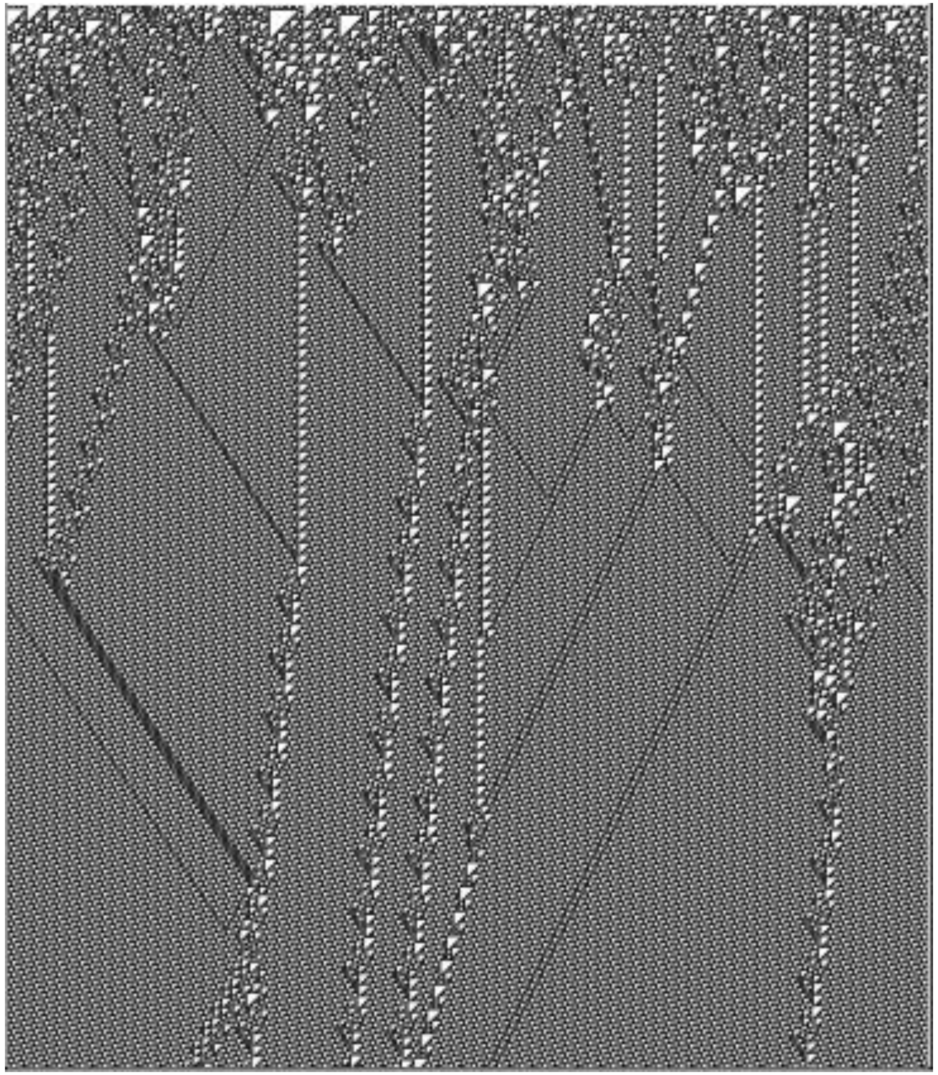
rule 150



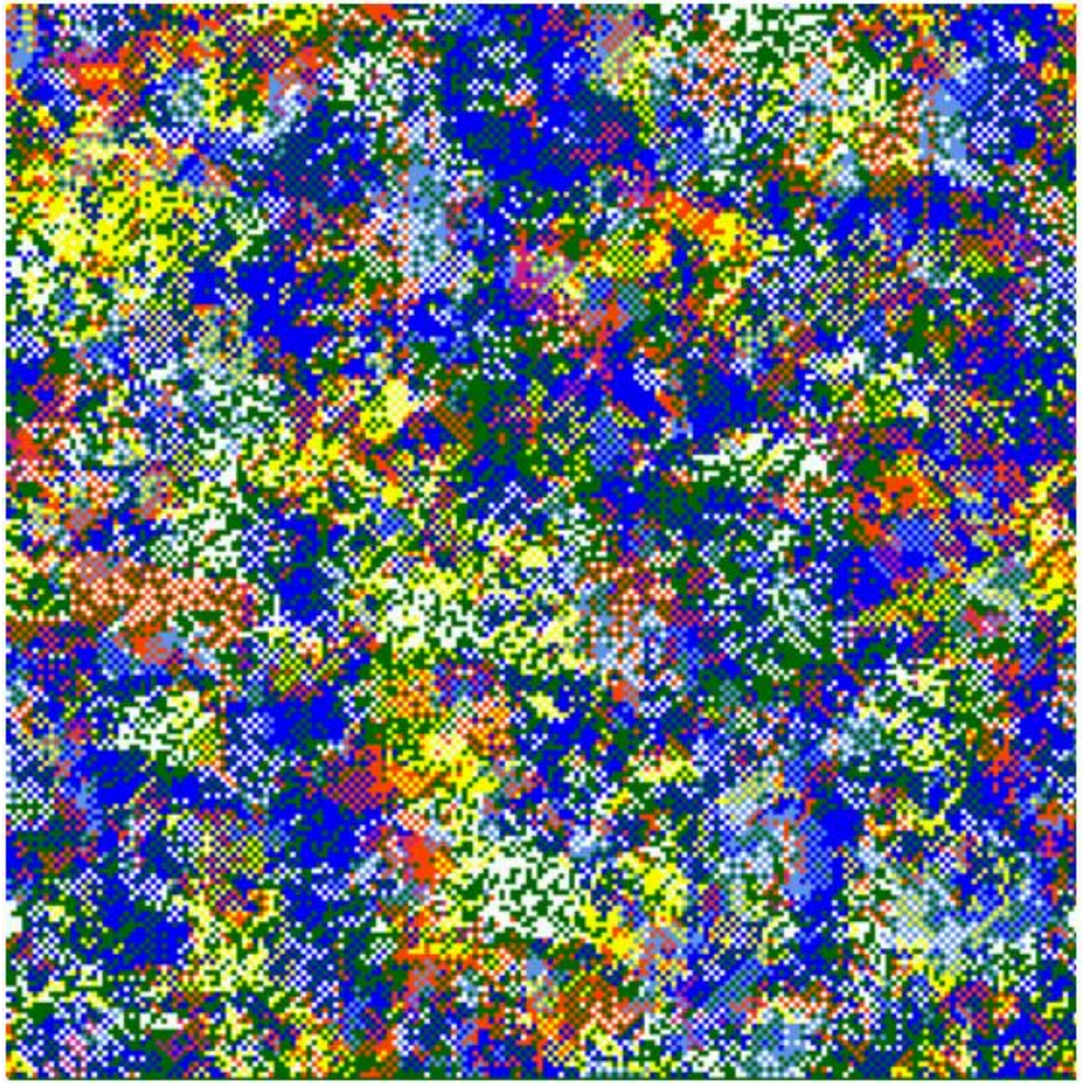
rule 182

Las estructuras de **Clase 4** son aquellas cuyas configuraciones iniciales dan lugar a un **patrón ordenado** inmerso en un fondo de aleatoriedad.

Algunas reglas de Clase 4 son: 110, 124, 135, 141.

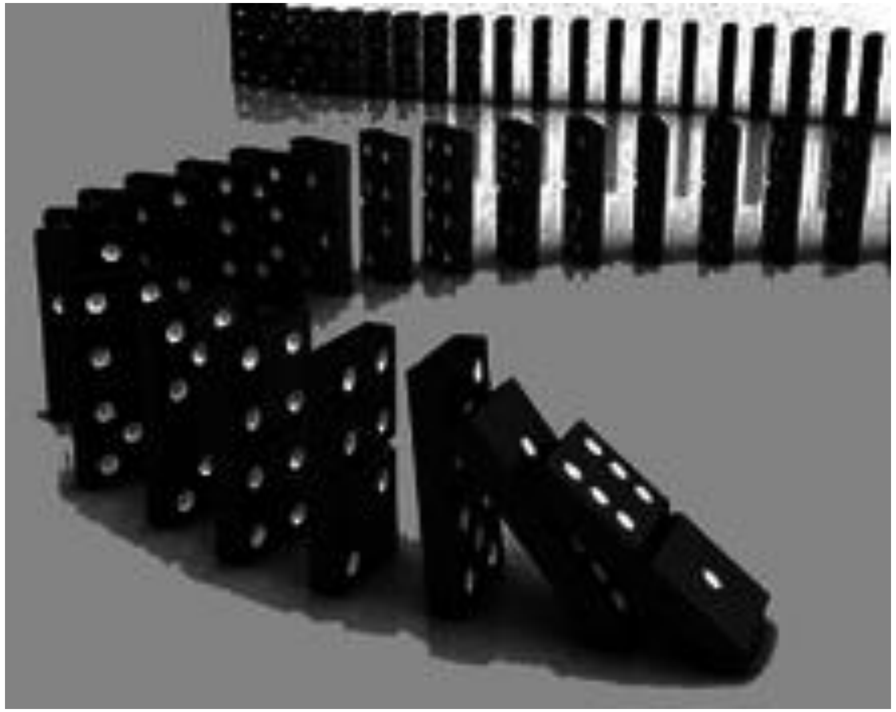


Discuta la diferencia entre los conceptos de comportamiento determinista, aleatorio, pseudoaleatorio y caos.



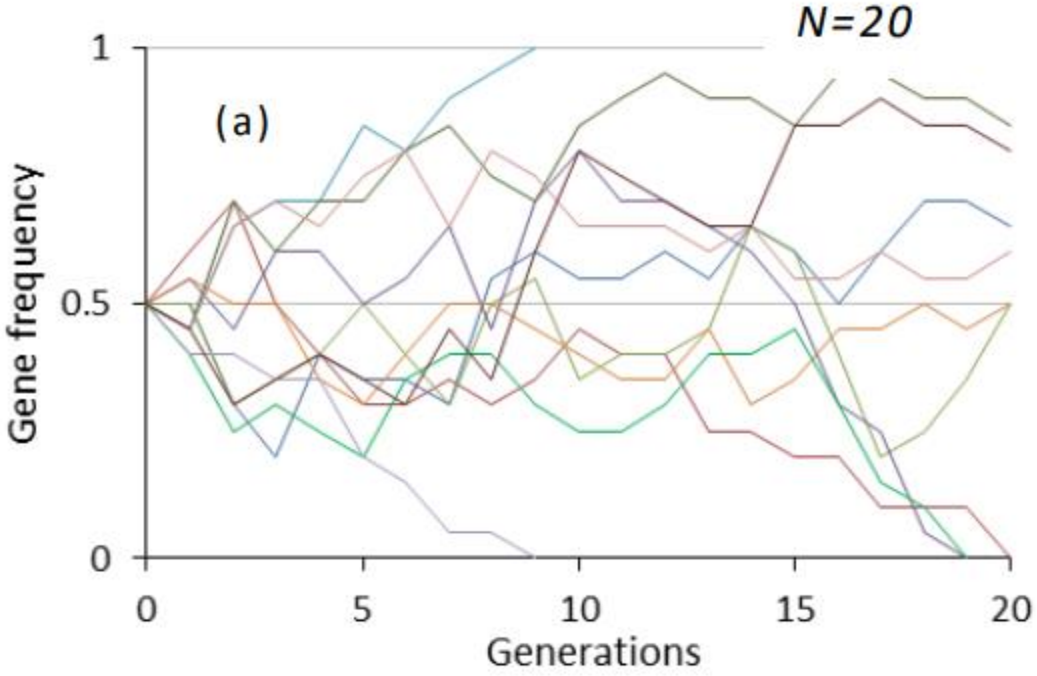
Un **comportamiento determinista** es aquel que para las mismas condiciones iniciales genera siempre el mismo resultado de acuerdo a un conjunto de reglas (funciones, ecuaciones, reglas lógicas, etc.).

Ej.: Las leyes de la mecánica newtoniana son deterministas, las máquinas de Turing son deterministas.



Un **comportamiento aleatorio** es aquel que para las mismas condiciones iniciales el resultado no siempre es el mismo, sino que está dado por una distribución de probabilidades.

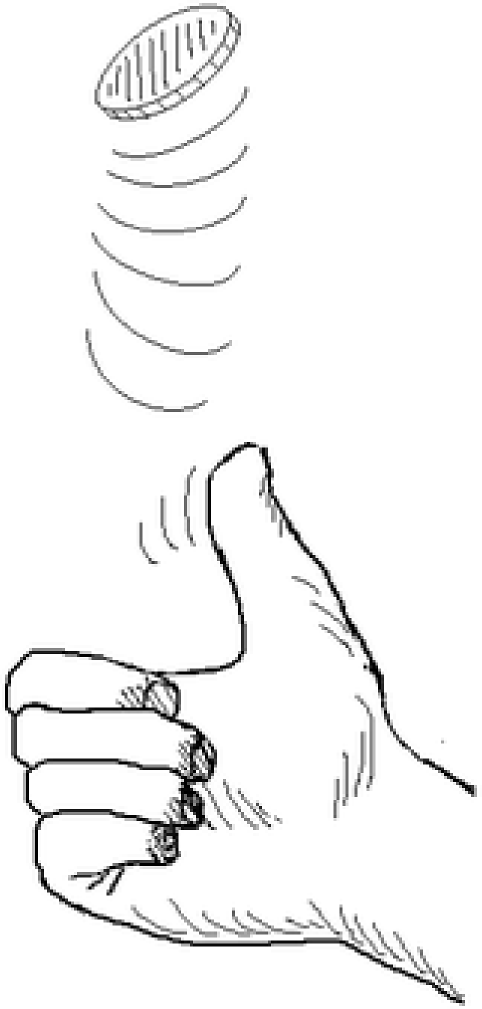
Ej.: Mutaciones aleatorias, Deriva genética[7], Movimiento Browniano.



[7] Shaukat, S., et al. (2020). Random numbers and monte carlo simulation: Applications in genetic drift and random walk models of predator-prey interaction. *Int. J. Biol. Biotechnol*, *17*, 195-218.

Es importante recalcar que muchos procesos que consideramos aleatorios en realidad son deterministas.

Por ejemplo, si se conocieran todas las condiciones iniciales y las fuerzas que actúan al tirar una moneda, podríamos ser capaces de predecir con exactitud de qué lado caería la moneda.

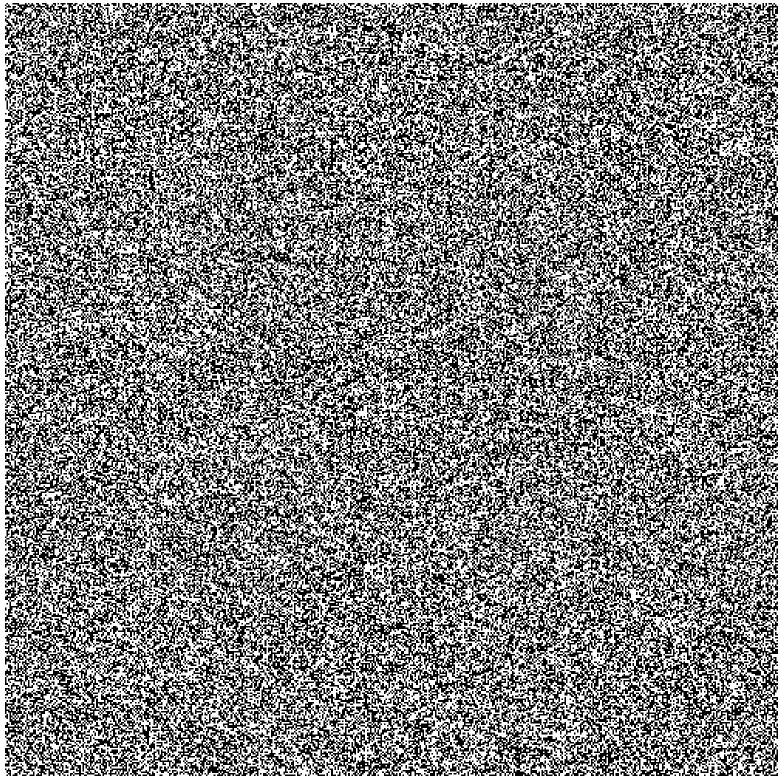


Al decir que el proceso “*tirar una moneda*” es aleatorio estamos ignorando la complejidad del cálculo determinista, suplantándola por el calculo probabilístico que es más sencillo *pero menos certero*.

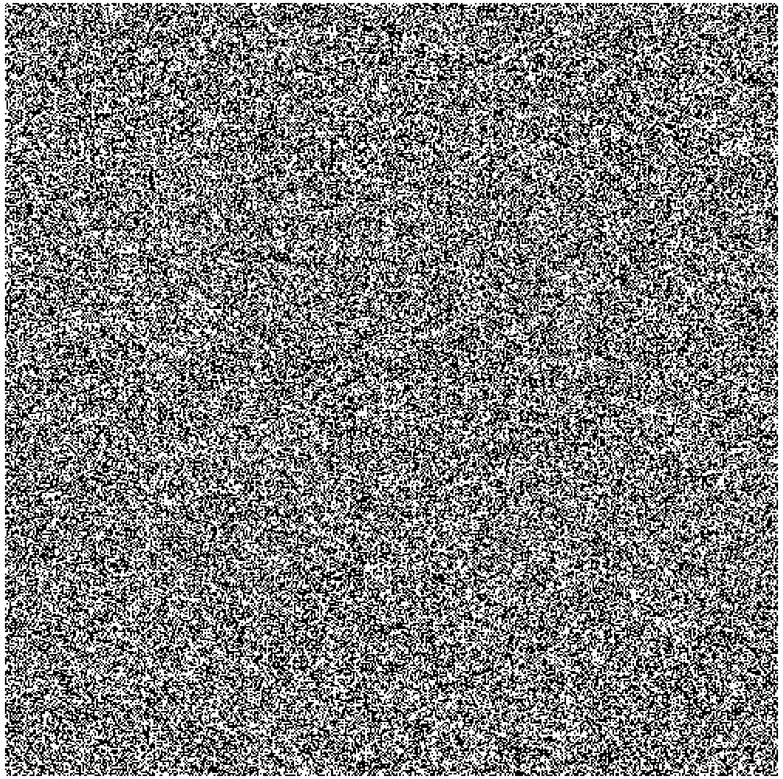
En esencia, todas las ramas de la Ciencia toman una postura dual entre la búsqueda de leyes deterministas y enfoques probabilísticos.

¿Cómo es posible generar “números aleatorios” con un ordenador determinista? ¿Son efectivamente aleatorios esos números?

Es posible simular la generación de números “aleatorios” a partir de máquinas deterministas (como una computadora) mediante **Algoritmos Pseudoaleatorios**.



Cuando decimos que el comportamiento de un autómata es “aleatorio” en realidad queremos decir que es “pseudoaleatorio” porque en el fondo, las reglas por las que se rige son deterministas. Cualquier planilla de cálculo o software matemático utiliza códigos deterministas e intenta obtener un resultado que simula una distribución de probabilidad.



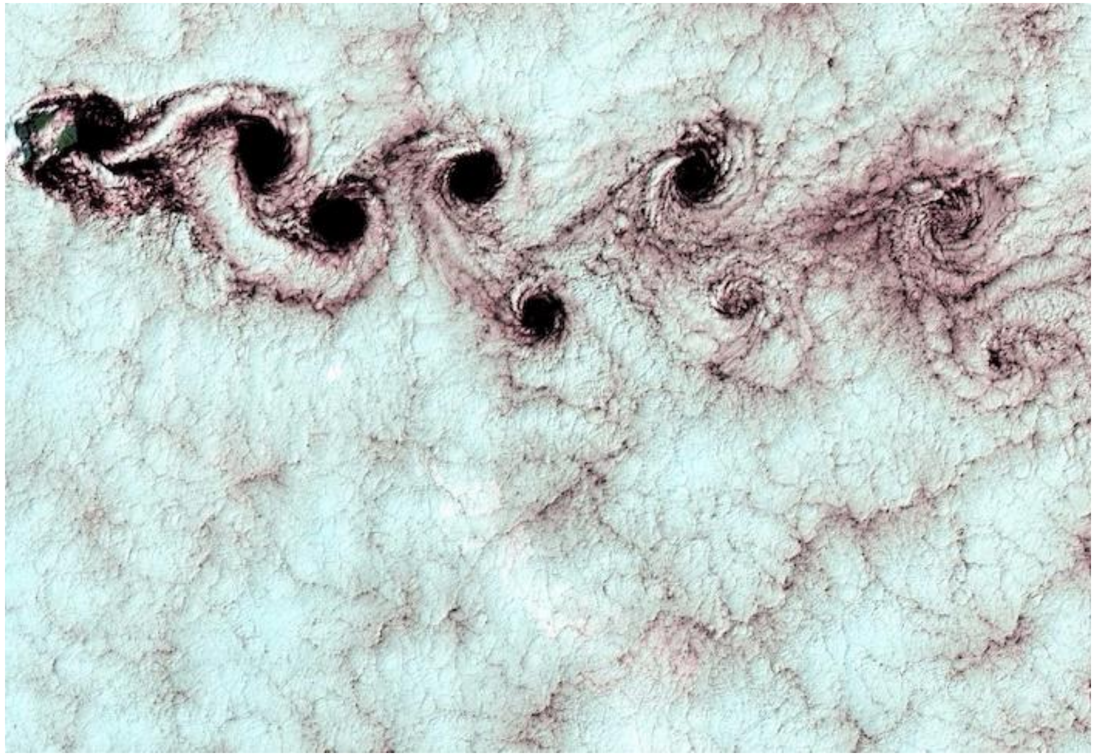
Un **comportamiento es caótico** cuando pequeños cambios en las condiciones iniciales de un mismo sistema genera respuestas completamente diferentes, "*aparentemente aleatorias*".

Ej.: El péndulo doble, el clima, las órbitas de los planetas, el cerebro humano, la evolución biológica.

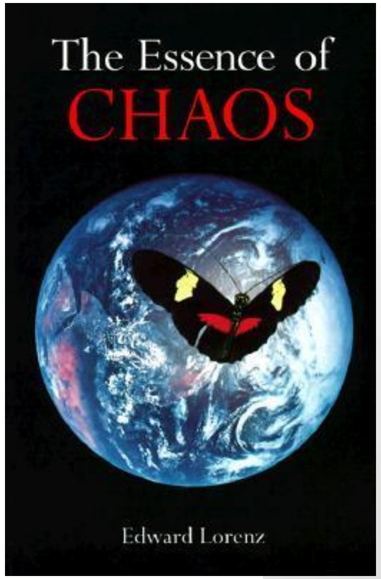
Pequeñas diferencias en las condiciones iniciales pueden llevar a resultados y comportamientos completamente diferentes a pesar de que el sistema se rige por reglas deterministas y que por tanto deben llevar a un único resultado final. Esto implica que los sistemas caóticos son **impredecibles** a largo plazo.

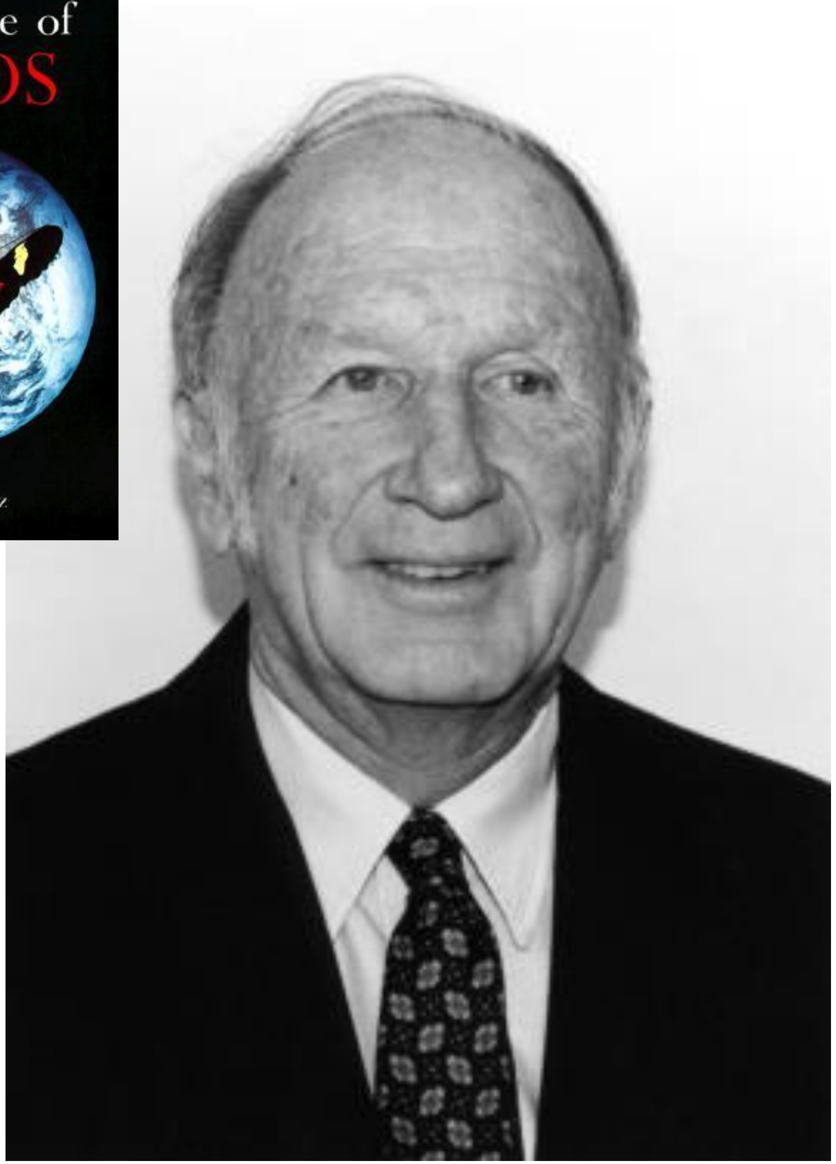
Muchos sistemas en la naturaleza son en parte caóticos debido a la enorme (e incomputable) variedad de factores que determinan su evolución temporal.

Piense, por ejemplo, en todos los factores que influyen en la predicción del tiempo meteorológico.



“*Caos: cuando el presente determina el futuro, pero el presente aproximado no determina aproximadamente el futuro*”. – Edward Lorenz





Resumen

Un autómata celular es un sistema dinámico que evoluciona a pasos discretos de acuerdo con un conjunto de reglas deterministas.

El autómata de Conway (‘*Game of Life*’) es bidimensional, y se rige por leyes de supervivencia, nacimiento y muerte de acuerdo con las interacciones de las celdas vecinas.

El autómata de Wolfram es unidimensional y se rige por reglas codificadas a través de números binarios y funciones booleanas.

Ambos autómatas son capaces de generar complejidad a partir de un número sencillo de reglas y configuraciones iniciales.

Resumen

Un sistema es determinista si a partir de las mismas condiciones iniciales se obtienen los mismos resultados. Un sistema es aleatorio si a partir de las mismas condiciones iniciales se obtienen resultados diferentes con cierta probabilidad.

En general, muchos sistemas naturales son deterministas, pero la complejidad para abordarlos es tan grande que utilizamos un enfoque probabilístico que es más sencillo pero menos certero.

Los computadores logran generar números pseudoaleatorios a partir de algoritmos que simulan una distribución aleatoria de probabilidad.

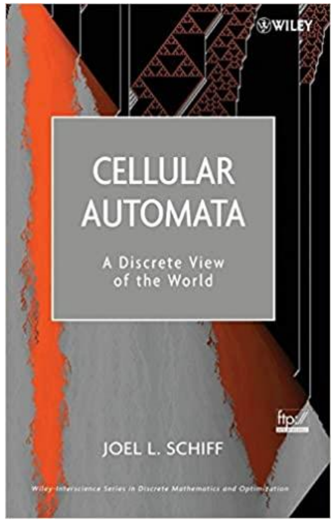
Resumen

Un sistema es caótico si cambiando las condiciones inicial en pequeñas cantidades se obtienen comportamientos muy diferentes a largo plazo. Los sistemas caóticos son deterministas, pero impredecibles.

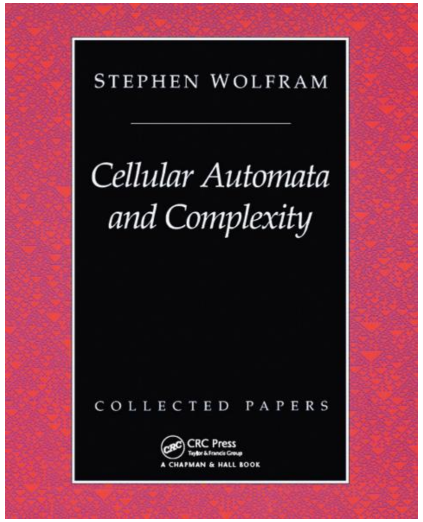
La emergencia de patrones complejos así como la incapacidad de predecirlas, es un fuerte contraejemplo del Determinismo Científico.

Los autómatas celulares son útiles en Biología, por ejemplo en Morfogénesis, Sociobiología, Ecología, así como otras ramas de la ciencia (Física, Química, Astronomía) y del conocimiento (Arquitectura, Música).

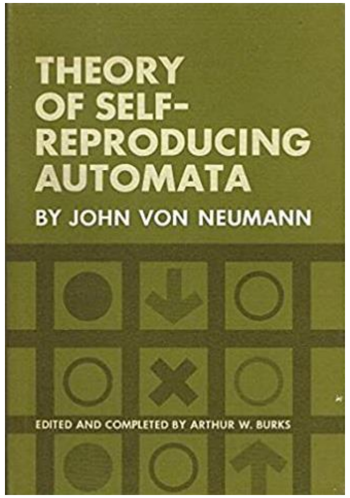
Bibliografía



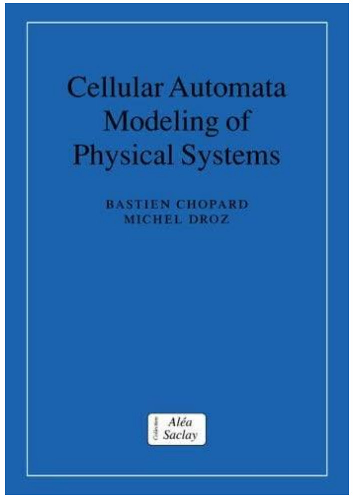
Schiff, J. L. (2011). *Cellular automata: a discrete view of the world*. John Wiley & Sons. **[Disponible en Biblioteca]**



Wolfram, S. (2018). *Cellular automata and complexity: collected papers*. CRC Press. **[Disponible en Biblioteca]**

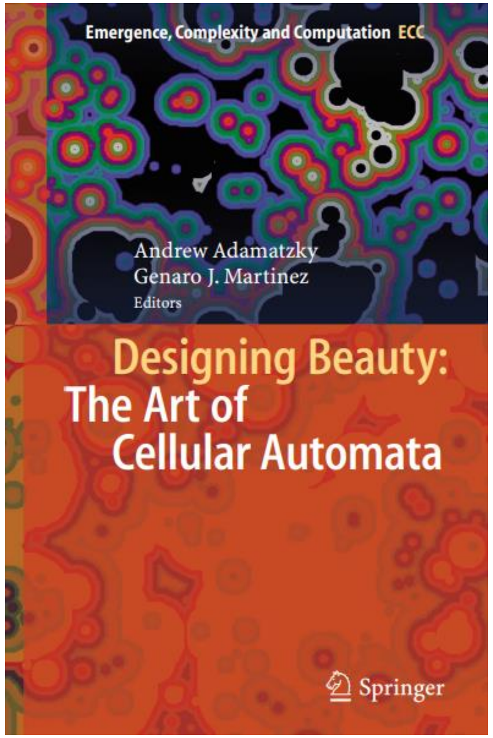


Neumann, J. V. (1966). *Theory of self-reproducing automata*. Edited by Arthur W. Burks. **[Disponible en Web]**

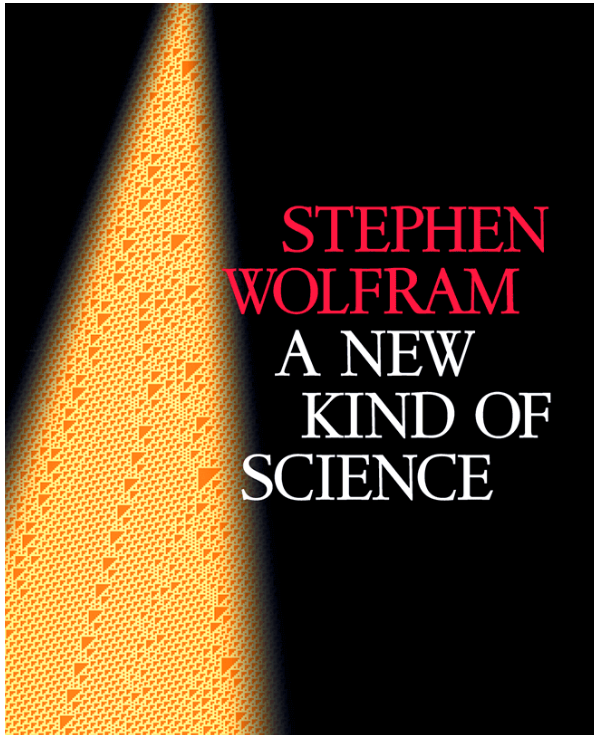


Chopard, B., & Droz, M. (1998). *Cellular automata. Modelling of Physical Systems.* Cambridge University Press. **[Disponible en Biblioteca]**

Bibliografía



Adamatzky, A., & Martínez, G. J. (Eds.). (2016). *Designing beauty: the art of cellular automata* (Vol. 20). Springer. [Disponible en Web]



Wolfram, S. (2021) *A new kind of science*. Cap. 6: *Starting from Randomness*. Págs.: 223-965. [Disponible en Web (por capítulos)]

Material complementario

Es posible obtener música (en diferentes géneros e instrumentos) a partir del autómata de Wolfram.
En este link podrán acceder a la página: [WolframTones](#).
En este otro link acceden a un video explicativo (en inglés):
How Computers Write Music: Cellular Automata - YouTube