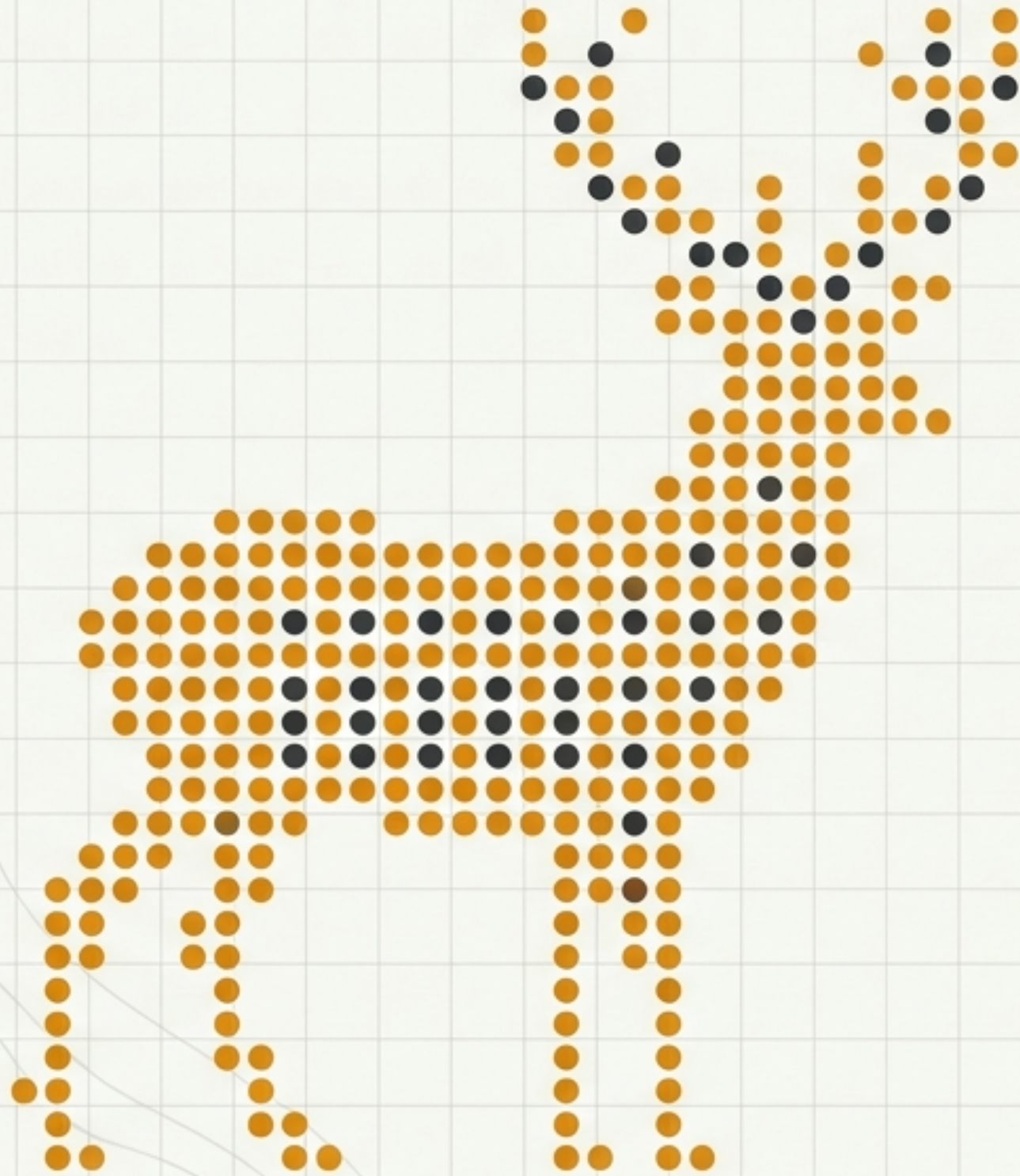




Naturaleza en Código

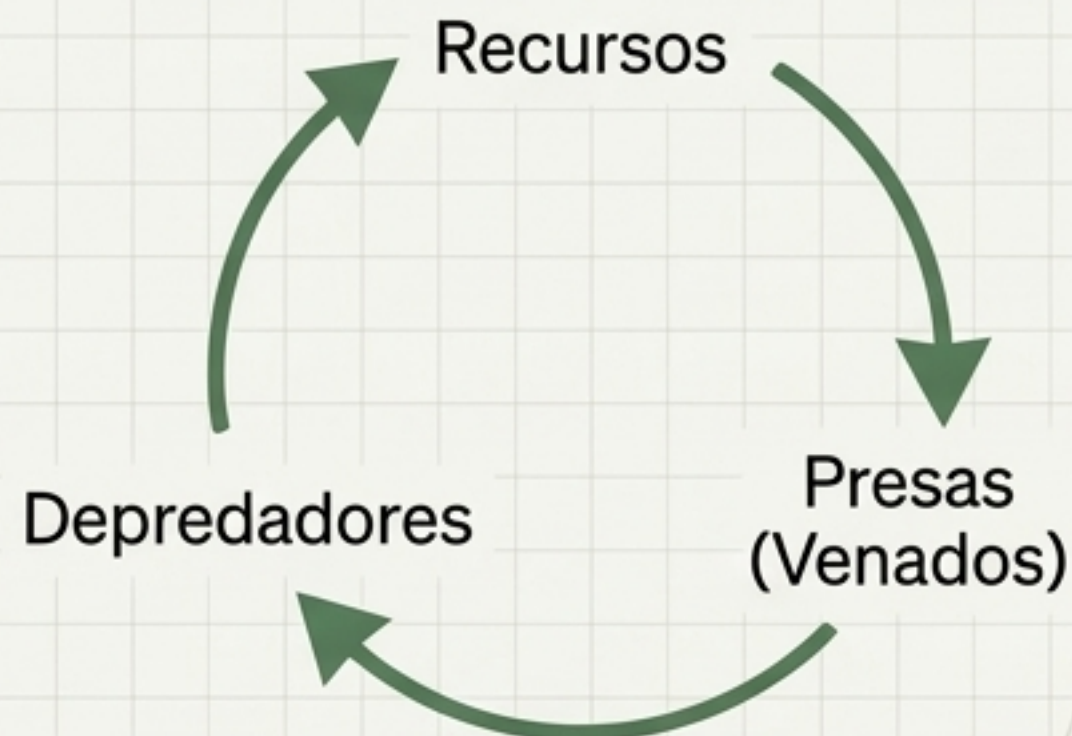
Simulando dinámicas de depredador-presa a través de matrices y funciones en micro:bit.

Ciencias Computacionales + Ciencias Naturales | Sesión 5



En la naturaleza, el balance lo es todo

Sin depredadores naturales, una especie consume todos los recursos disponibles y su ecosistema colapsa. Cuando esta dinámica biológica se rompe, las especies nativas sufren.



¿Podemos predecir el futuro de una población usando tecnología sin alterar el ecosistema real?

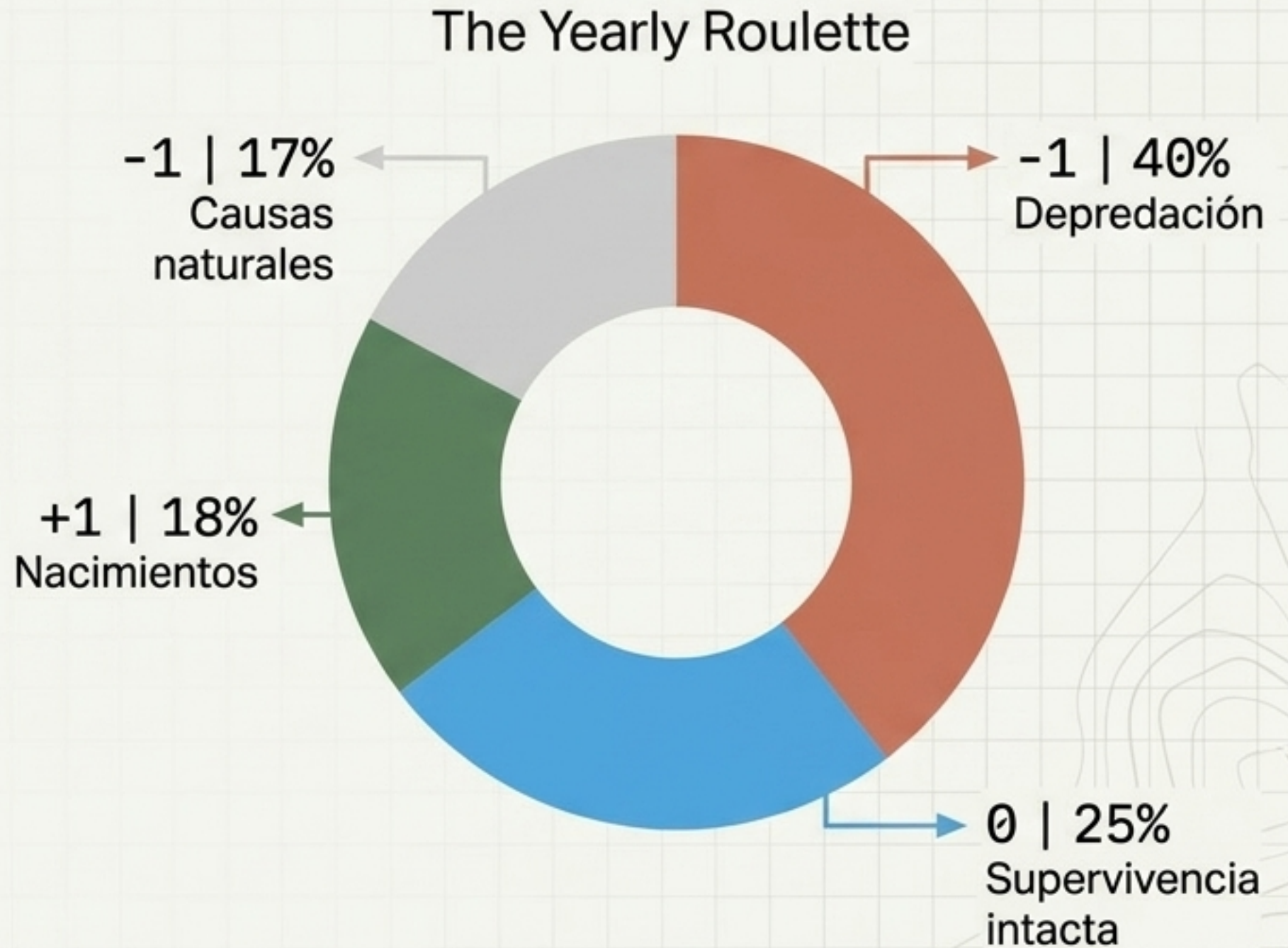
Las Reglas de la Simulación



A

200

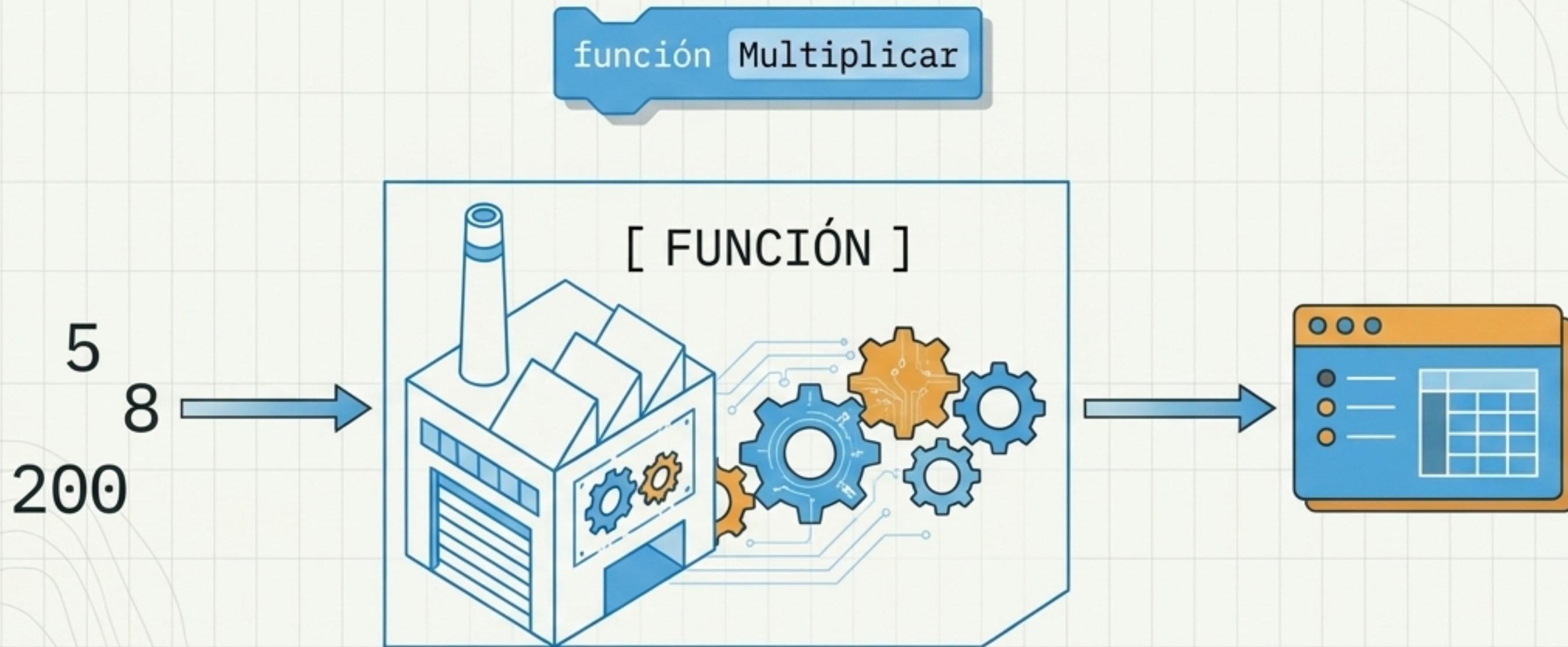
Población inicial
al presionar [A]



La simulación se ejecuta hasta alcanzar la extinción (Población = 0).

La herramienta lógica: Funciones reutilizables

Una función es un fragmento de código encapsulado bajo un nombre propio (disponible en la sección Avanzado). En lugar de reescribir la misma lógica 100 veces para nuestra simulación, la escribimos una sola vez y la llamamos cuando la necesitamos.



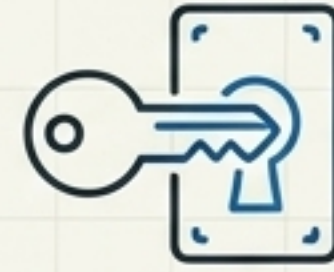
El ingrediente secreto del código



VARIABLES

Globales. Son visibles y modificables para todo el ecosistema digital.

Ejemplo: El contador total de venados vivos.



PARÁMETROS

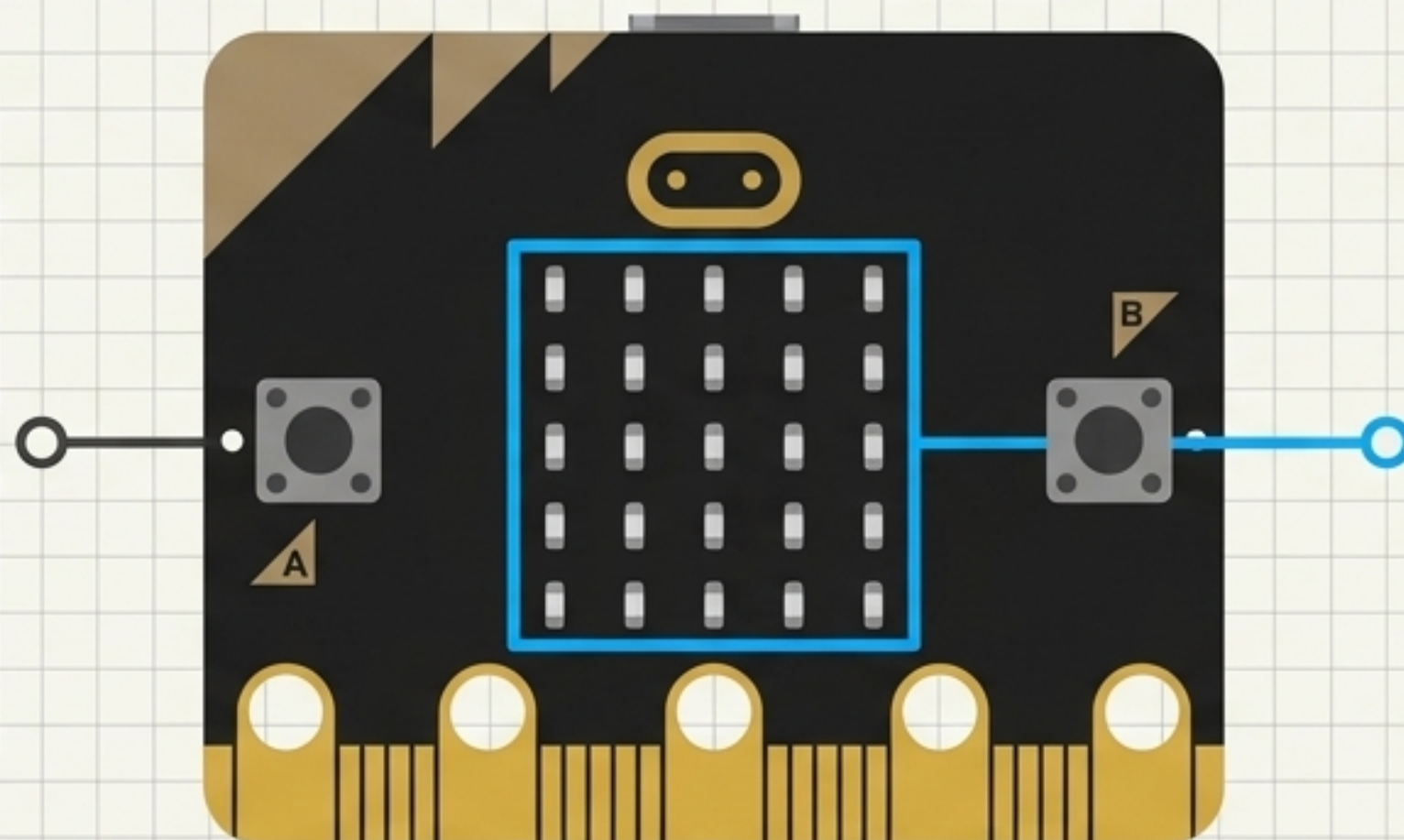
Locales. Existen exclusivamente dentro de su función específica. Permiten definir operaciones complejas antes de saber qué valores exactos se van a usar en la simulación.

Los parámetros hacen que nuestras funciones sean genéricas, adaptables e independientes.

El reto de visualización física

El Problema

Nuestra simulación maneja cientos de venados, pero la placa micro:bit cuenta con únicamente 25 luces LED.



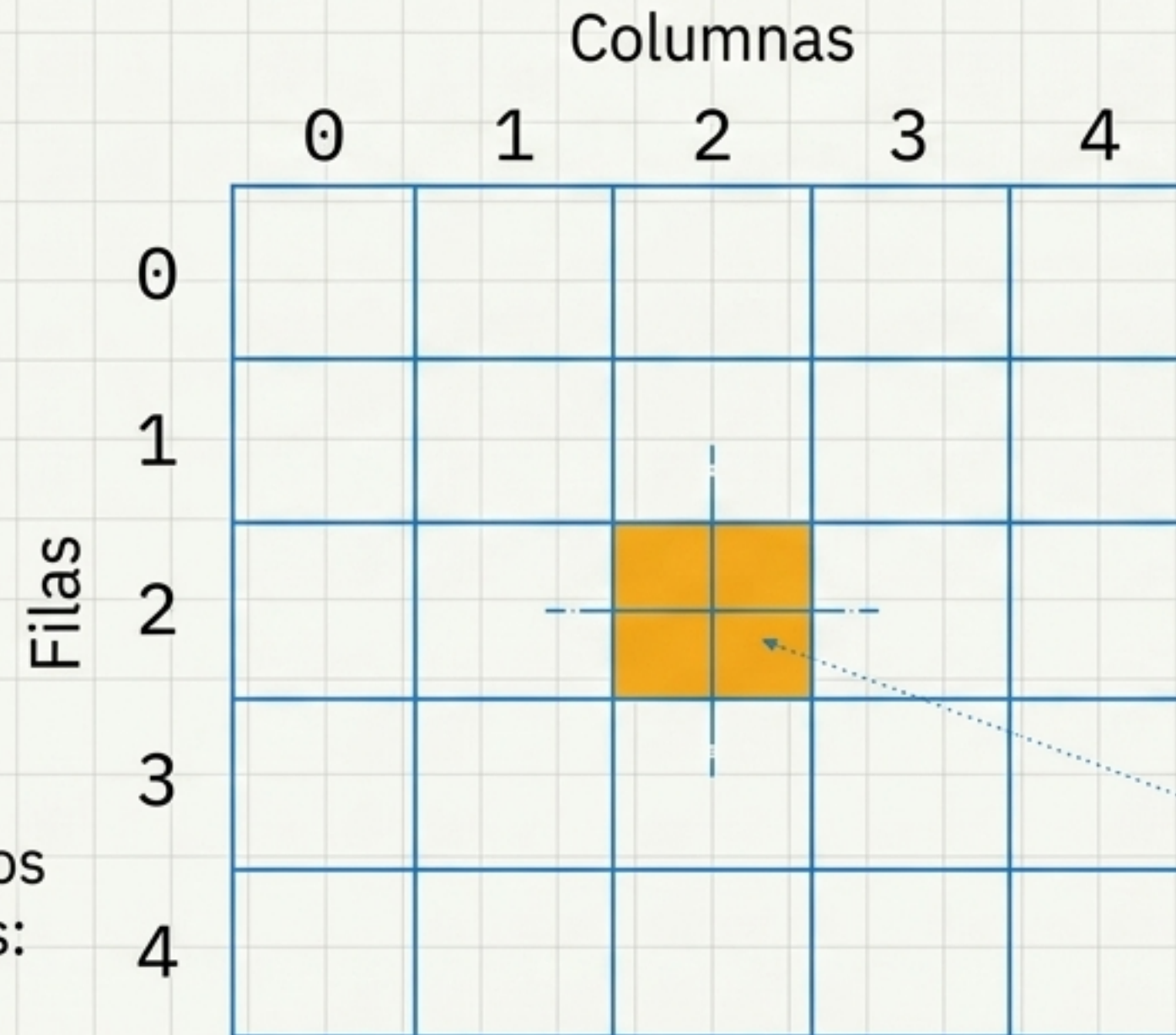
La Solución

Escalar los datos. Cada luz LED encendida representará gráficamente a 10 venados.

Meta: Mostrar el volumen exacto de la población de venados al inicio de cada año directamente en la pantalla de la placa.

Anatomía de la matriz (Plano 2D)

La pantalla opera como un mapa de coordenadas bidimensional. Se lee de izquierda a derecha, y de arriba hacia abajo.

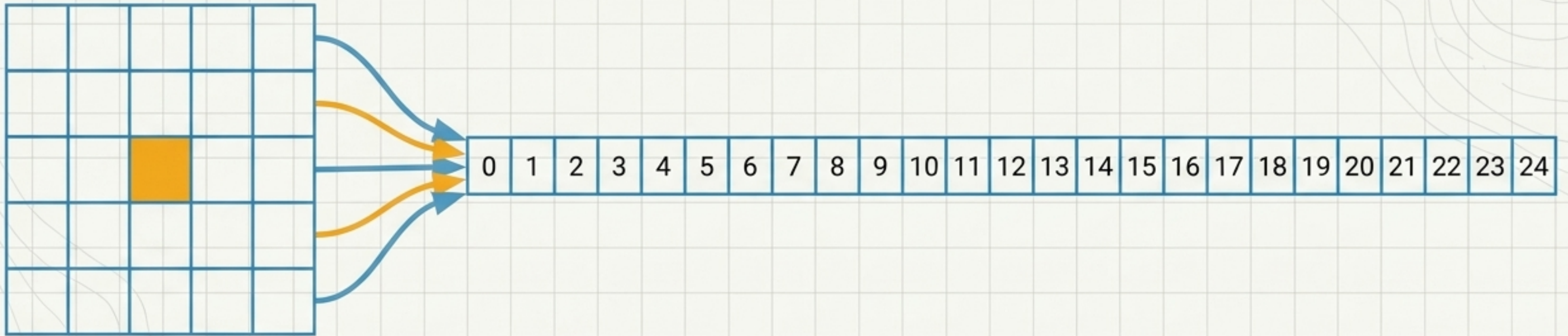


- Nombrar un LED específico requiere dos coordenadas precisas: (Fila, Columna).

El punto central exacto de la pantalla es la coordenada (2, 2).

De dos dimensiones a una línea de tiempo

Para controlar los LEDs eficientemente, la micro:bit permite representar la cuadrícula 2D como un Arreglo (Array) de una dimensión.



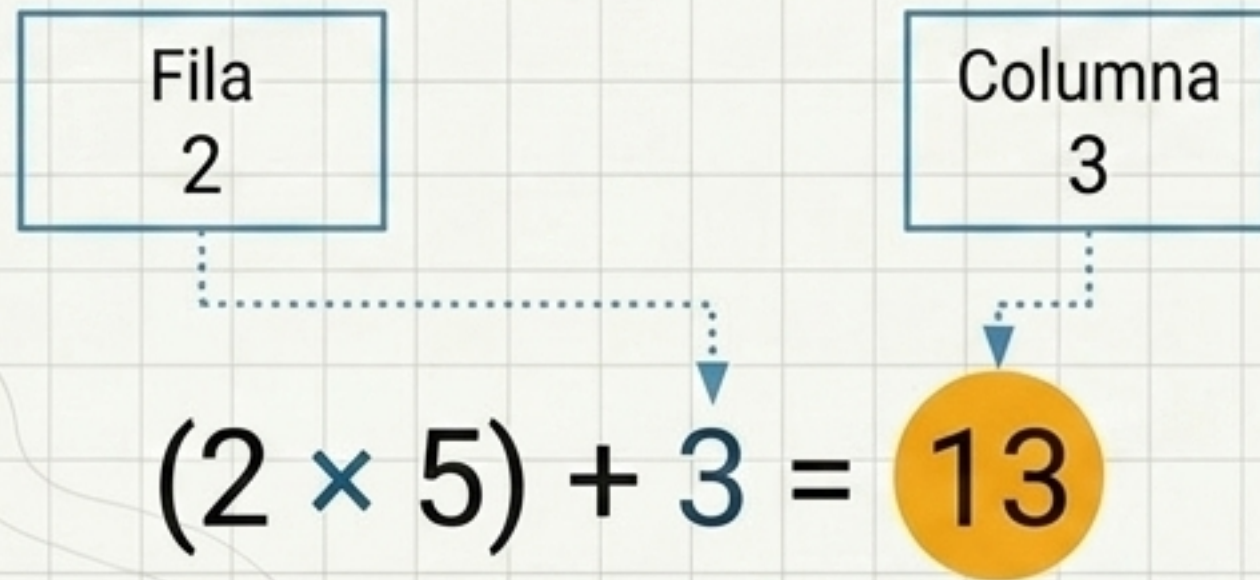
Key Insight

En lugar de pedir al sistema la 'Fila X, Columna Y', simplemente pedimos encender la 'Posición Z'. **El mapa de coordenadas se aplasta en un mapa lineal.**

La lógica matemática detrás de la matriz

Esta sencilla ecuación traduce el mundo 2D al mundo 1D instantáneamente.

$$\text{Posición} = (\text{Fila} \times 5) + \text{Columna}$$

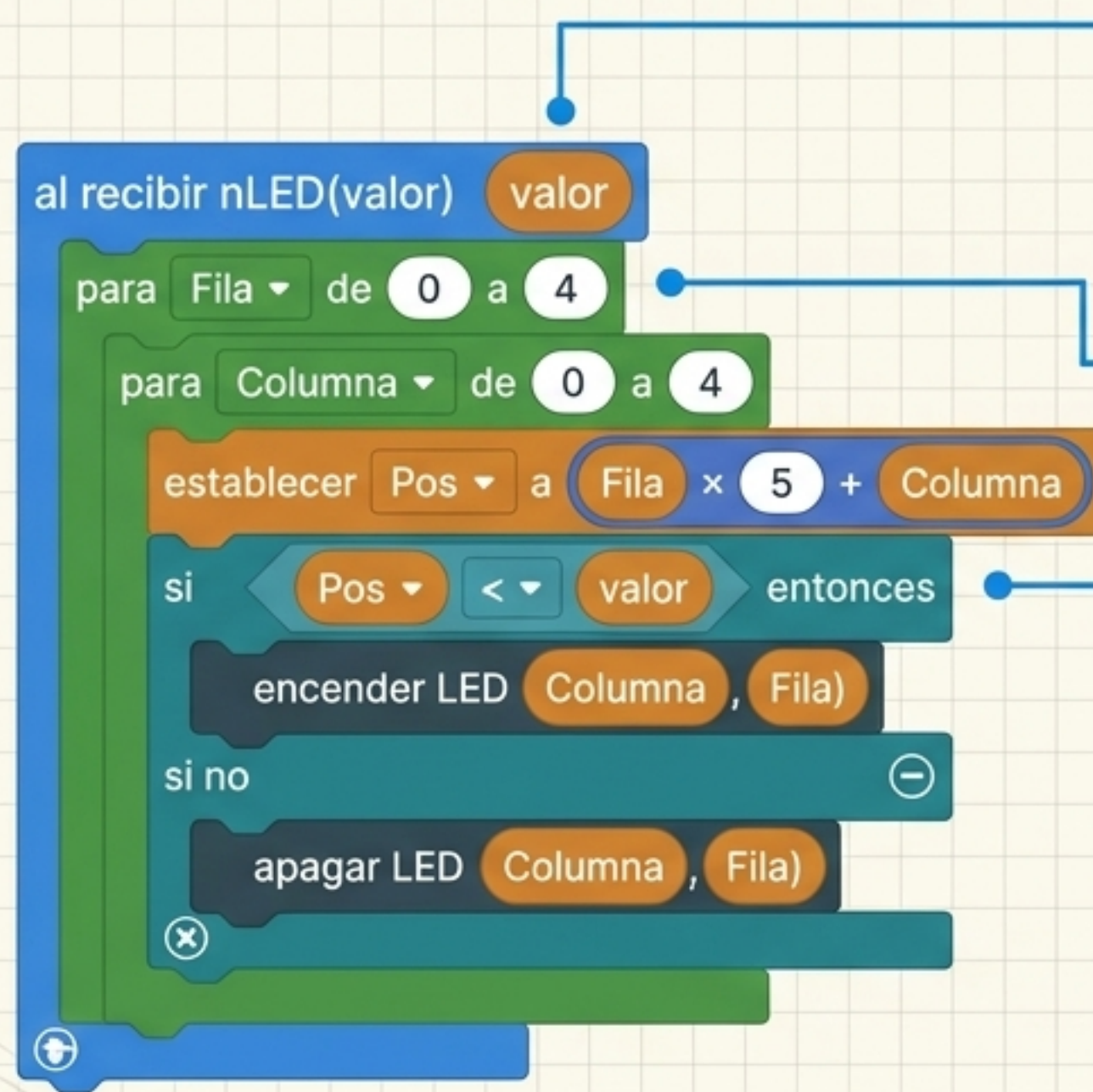


Es el 14º LED!

Pregunta: ¿Cómo cambiaría esta fórmula matemática si nuestra matriz fuera de 10x10 en lugar de 5x5?

(Respuesta: Se multiplicaría por 10)

Pintando la pantalla con código

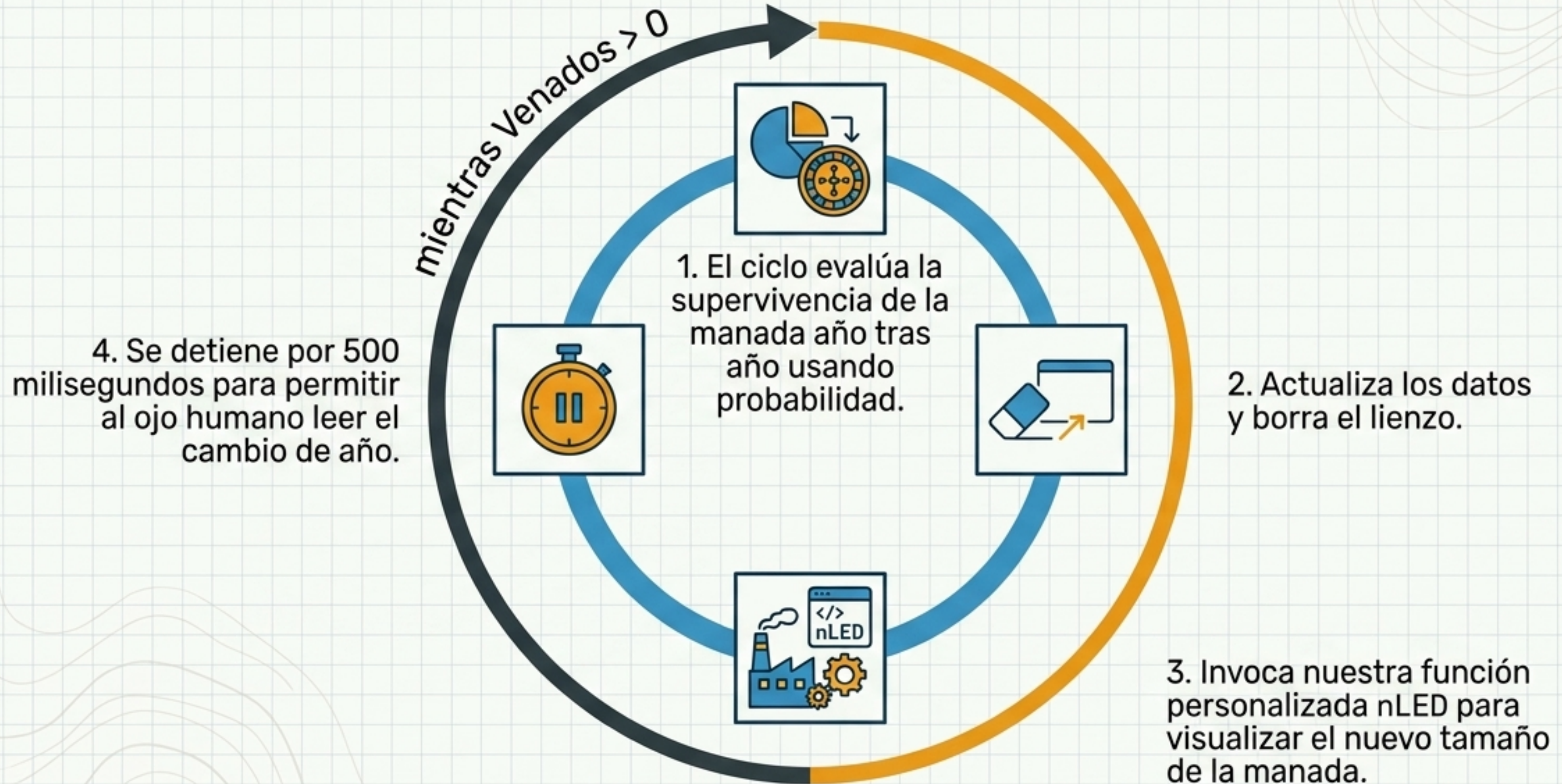


El Parámetro: Dicta la cantidad exacta de LEDs a encender basados en la población actual.

Bucles anidados: El radar que recorre sistemáticamente cada espacio del ecosistema digital.

La Condicional: Verifica la posición calculada usando nuestra fórmula matemática y decide encender (graficar) o apagar (ocultar) la luz.

El ensamblaje de la simulación





El poder de jugar a ser dioses digitales

No podemos alterar ecosistemas frágiles poniendo poblaciones reales de lobos y venados en riesgo solo para experimentar qué sucede.

Las simulaciones computacionales nos permiten probar predicciones, cometer errores drásticos y aprender a salvar especies, todo a la velocidad de la luz y con un impacto ambiental cero.



Verificación de Sistemas (Fin de Misión)



¿Logré predecir el comportamiento biológico de la simulación?



¿Comprendo cómo reutilizar lógica de programación creando Funciones con Parámetros?



¿Puedo transformar visualmente una cuadrícula 2D en un arreglo continuo lineal de 1D?

La programación es prueba y error continuos.
Experimenta en MakeCode y ajusta tus ecosistemas.