

Sesión 5

Aprendizajes esperados

Al final de esta sesión verifica que puedas:



Reutilizar fragmentos de código por medio de funciones.

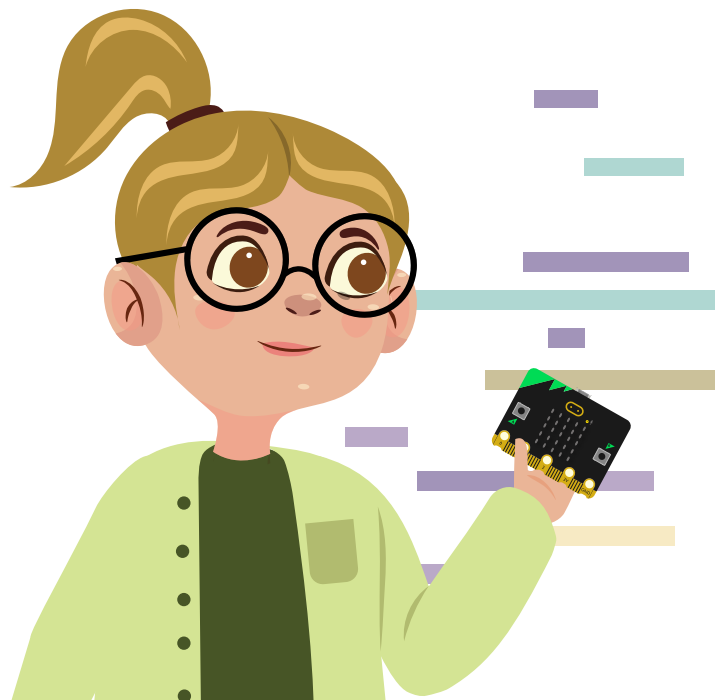
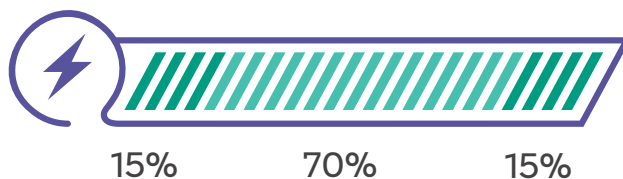


Representar una matriz de dos dimensiones como un arreglo.

Material para la clase

- ☐ Anexos 1.3, 4.1, 5.1.
- ☐ Acceso a MakeCode.

Duración sugerida



Anexo

Anexo 1.3

Un contexto típico en ciencias naturales es la dinámica conocida como depredación. En muchos ecosistemas esta dinámica permite mantener sanos a poblaciones compatibles con la disponibilidad de recursos. Cuando esta dinámica se rompe, a menudo se afectan especies nativas. Por ejemplo, cuando se introducen una especie no nativa que crece sin control por no tener depredadores naturales, puede acabar con especies locales. En este caso, la depredación es un fenómeno natural, también, en nuestro país. El siguiente ejercicio habla de una población de venados, pero algo parecido puede suceder con otras especies locales, como mamíferos y reptiles, aves y peces. No dudes en investigar qué animales en tu región tienen depredadores.

En este reto se propone hacer un programa en MakeCode para simular la evolución de una población de venados a través de los años. De ser posible, descargarlo e ejecutarlo en una micro:bit. Aquí están las especificaciones del programa:

- 1 Se empieza la simulación con 200 venados al oprimir el botón A.
- 2 Cada año, se debe definir la suerte de cada venado vivo según las siguientes probabilidades:
 - 10%: nace un venado, se debe incrementar en 1 la cantidad de venados.
 - 40%: un venado es comido por un depredador, se reduce en 1 la cantidad de venados.
 - 10%: un venado muere de causas naturales, se reduce en 1 la cantidad de venados.
 - 35%: el venado sigue vivo, la cantidad de venados sigue igual.



- 3 La pantalla LED de la micro:bit debe mostrar una representación gráfica de la cantidad de venados vivos al principio de cada año durante la simulación.
- 4 La simulación termina cuando los venados se extinguen.
- 5 Al final de la simulación, la micro:bit debe indicar cuántos años duró la población de venados.



Lo que sabemos, lo que debemos saber



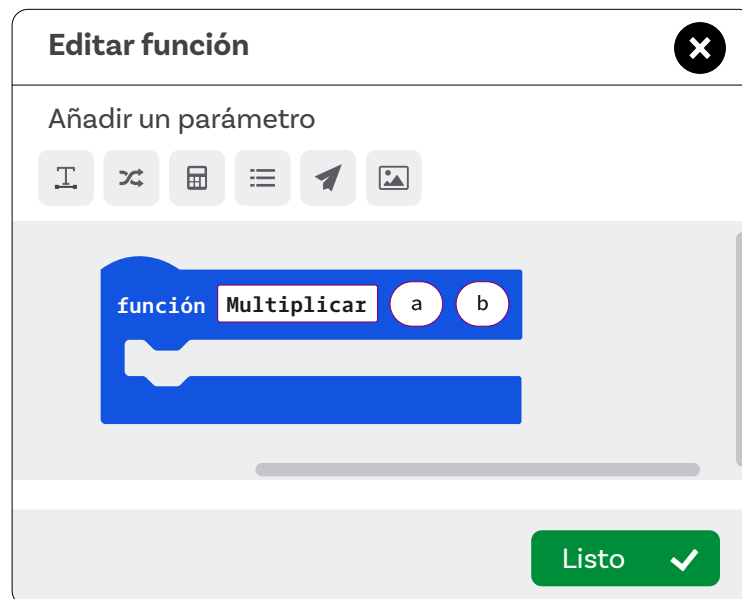
Esta sección corresponde al 15% de avance de la sesión

Es el momento de resolver el reto que se encuentra en el Anexo 1.3, te proponemos leerlo de nuevo antes de seguir.

Las funciones se encuentran en la sección Avanzado. Son fragmentos de código a los que se les asigna un nombre para ser ejecutados en cualquier momento llamándolos por dicho nombre. Además, estos fragmentos de código pueden tener uno o más parámetros, que se identifican en la parte superior y que determinan el resultado de la ejecución.

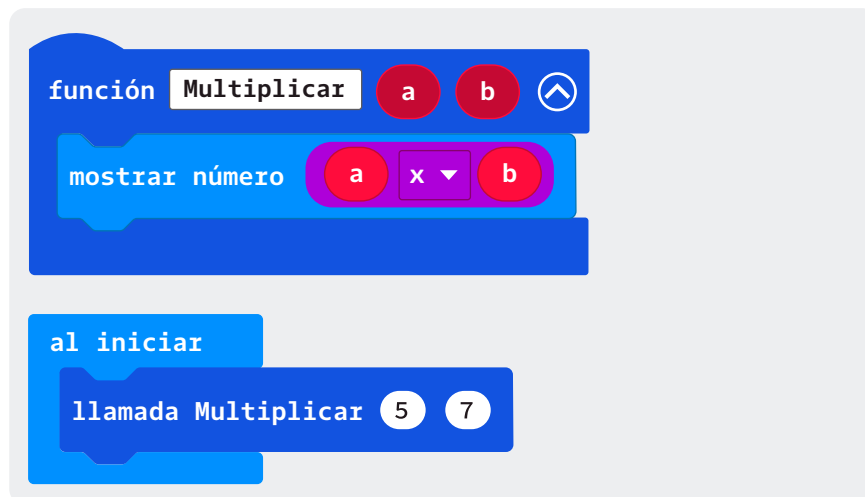
La Figura 1 presenta una función para multiplicar dos números, el resultado dependerá de los parámetros con los que se llame la función.

Figura 1. Bloque de función Multiplicar



Si se requiere:

- En la Guía 2 de grado 6 se introduce el concepto de funciones.

Figura 2. Programación función de multiplicación de dos números

Si quieres agregar parámetros a una función lo puedes hacer al momento de crearla, como se muestra en la *Figura 1*.

Luego, cuando estés definiendo lo que hace la función arrastra los parámetros, que están al lado del nombre de la función, para usarlos en los bloques al interior de esta. Mira la *Figura 2*.

Los parámetros facilitan la reutilización de las funciones al permitir definir las operaciones dentro de estas de forma genérica, es decir, sin necesidad de fijar los valores que se van a utilizar internamente. Por esto los parámetros solo existen dentro de su función, a diferencia de las variables que son visibles globales para todos los bloques. ¡Compruébalo en la lista de variables!

Manos a la obra

Conectadas



Esta sección corresponde al 85% de avance de la sesión

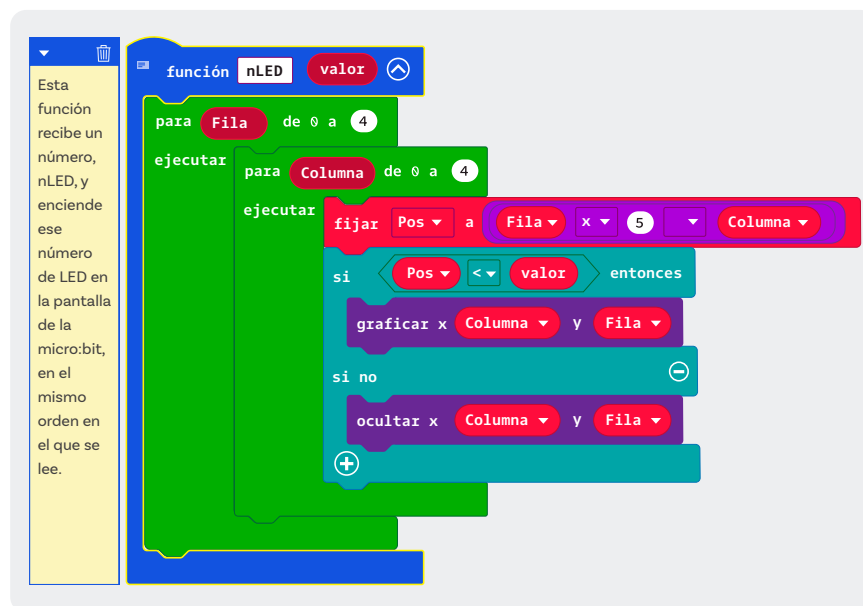
Organízate en grupos de dos a tres personas de acuerdo con las indicaciones de tu docente y observen la función para encender los primeros n LED de la *micro:bit* que se encuentra en la *Figura 3*. Para hacerlo, la función representa la matriz de LED de la *micro:bit* en un arreglo. Un arreglo es un conjunto de datos que recibe el mismo nombre, pero tiene varios valores.

Anexo

Anexo 5.1

	Columnas				
Fila	0	1	2	3	4
0					
1					
2					
3					
4					

Figura 3. Función de control de LED en Matriz



Veamos cómo lo hace. Observar la representación en el Anexo 5.1 de la matriz de LED de la *micro:bit* como una tabla de 5 filas y 5 columnas (se enumeran de 0 a 4, de izquierda a derecha y de arriba a abajo).

Comenzando en la esquina superior izquierda de la tabla, escribir sobre cada celda números consecutivos desde el 0 hasta el 24. Estos números se usan para identificar cada celda, se llaman posición. Así, en vez de usar dos números, el de la fila y la columna, solo se necesita uno, el de la posición.

Ahora, ver la celda (2,3), fila 2, columna 3, por ejemplo.



¿Qué número escribieron en esa celda?
¿Cómo calcular ese número a partir de las “coordenadas” de la celda?

La fórmula para este procedimiento está en la función compartida.



¿Cómo cambiaría esta fórmula si la matriz fuera de 10x10?

Anexo

Anexo 4.1



Ahora, a complementar el programa de simulación de la sesión 3 con una visualización gráfica del número de venados al comenzar cada año a partir del código del Anexo 4.1.

Usar la función para encender los primeros nLED de la micro:bit. Como la micro:bit solo tiene 25 LED, cada uno representará 10 venados.

Finalmente, para alcanzar a ver la gráfica de cada año, agregar una pausa de 500 ms.

Antes de irnos



Esta sección corresponde al 100% de avance de la sesión

De forma individual, regresa a revisar los aprendizajes esperados. Elige la opción de respuesta que mejor describa lo que alcanzaste.

- 1 ¿Puedes utilizar los aprendizajes logrados para resolver el reto planteado?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no
- 2 ¿Puedes reutilizar fragmentos de código por medio de funciones?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no
- 3 ¿Puedes representar una matriz de dos dimensiones como un arreglo?
 - ☐ Sí
 - ☐ Parcialmente
 - ☐ Aún no



Si tus respuestas fueron “Parcialmente” o “Aún no”, vuelve a las actividades propuestas en *MakeCode*. Luego discute con tus compañeras y compañeros de grupo lo que se hizo en cada momento de la actividad y el rol al que correspondía. Si todavía te quedan dudas, consúltale a tu docente.



Computación y sociedad

Las simulaciones se utilizan en contextos de ciencias naturales y de ciencias sociales. Permiten hacer “experimentos” que no se podrían realizar en la realidad.

Por ejemplo, no se podría colocar una población de lobos y venados para experimentar con ellos y saber cuándo tendría problema una de las poblaciones. Solo se puede observar lo que sucede en la naturaleza.

En una simulación, muy rápidamente se pueden probar predicciones e ideas.

Volvamos al reto que tenías.



¿Cómo te fue en la solución del reto? ¿Qué fue difícil?, ¿por qué?

Haz una lluvia de ideas sobre los conceptos que hayas retenido con respecto a los dos aprendizajes. Incluye lo que se debe hacer para crear y utilizar una función con parámetros. También compara las dos formas de acceder a una celda de una tabla.

