

Arreglos

Ahora vamos a entender cómo **almacenar y manipular datos**.

Hasta ahora, hemos explorado cómo las variables nos permiten guardar información. Pero, **¿qué sucede cuando queremos manejar múltiples piezas de datos relacionados?**

Aquí es donde entran en juego los **arreglos**.

¿Qué es un arreglo?

Un arreglo es una estructura de datos que nos permite almacenar múltiples valores bajo un mismo nombre y se accede a su valor mediante un índice.

Puedes pensar en un arreglo como una fila de cajas de almacenamiento, donde cada caja puede contener un dato.



Cuando accedes al menú de MakeCode, encontrarás que ya existen dos tipos de listas o arreglos predefinidos:

Uno para almacenar números y otro para almacenar texto.

Editar nombres de listas:
Puedes editar los nombres de estas listas para que reflejen más claramente lo que almacenan.



Por ejemplo, si tienes una lista de números que representan puntuaciones en un juego, podrías renombrar la lista como Puntuaciones.

Esto hace que tu código sea más legible y fácil de entender.

Importante

Es importante recordar que todos los elementos dentro de un arreglo deben ser del mismo tipo de dato. Esto significa que si tienes un arreglo de números, solo deberías almacenar números en él.

Lo mismo aplica para los arreglos de texto; solo deberían contener cadenas de caracteres.

Índice de un arreglo

Los datos se almacenan en **posiciones contiguas en la memoria**.



Cada posición en un arreglo se identifica mediante un número llamado índice. Los índices permiten acceder a los elementos del arreglo de manera eficiente.



Inicio en cero

Los índices de los arreglos en MakeCode comienzan en cero.

Esto significa que el primer elemento tiene el índice 0, el segundo elemento el índice 1, y así sucesivamente.

Acceso a elementos

Para acceder a un elemento se utiliza su índice.

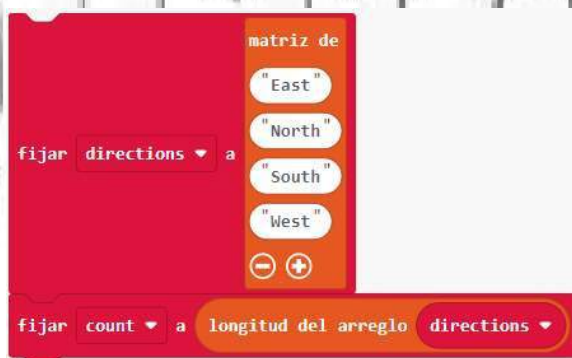


arreglo.

Pongamos a prueba tus conocimientos
¿Qué hace el siguiente código?

___X___ Crea un arreglo llamado direcciones con números del 1 al 4 y guarda el número 4 en la variable count.

___ Crea un arreglo llamado direcciones con las cuatro direcciones cardinales y guarda el número de direcciones en la variable count. La variable count almacena el valor de 3.



Largo de un arreglo

Las matrices siempre tienen una longitud, se trata de la cantidad de elementos almacenados. Los índices comienzan con 0 por lo que el índice del último elemento es uno menos que la longitud del

Funciones de Recorrida

Las funciones de recorrida son aquellas que **permiten pasar por cada elemento de un arreglo y realizar operaciones con ellos.**

En MakeCode estas funciones se representan con **bloques** que automatizan el proceso de iteración sobre los arreglos.

Bloque "para cada elemento"

Este bloque permite **ejecutar un conjunto de instrucciones** para cada elemento del arreglo.

Dentro del bloque, puedes usar la variable que representa el elemento actual para realizar operaciones como mostrarlo en la pantalla o modificarlo.



Video ejemplo:

<https://www.youtube.com/watch?v=zTP8lh9Wlr0>

En la siguiente imagen vemos un programa que utiliza este bloque. 

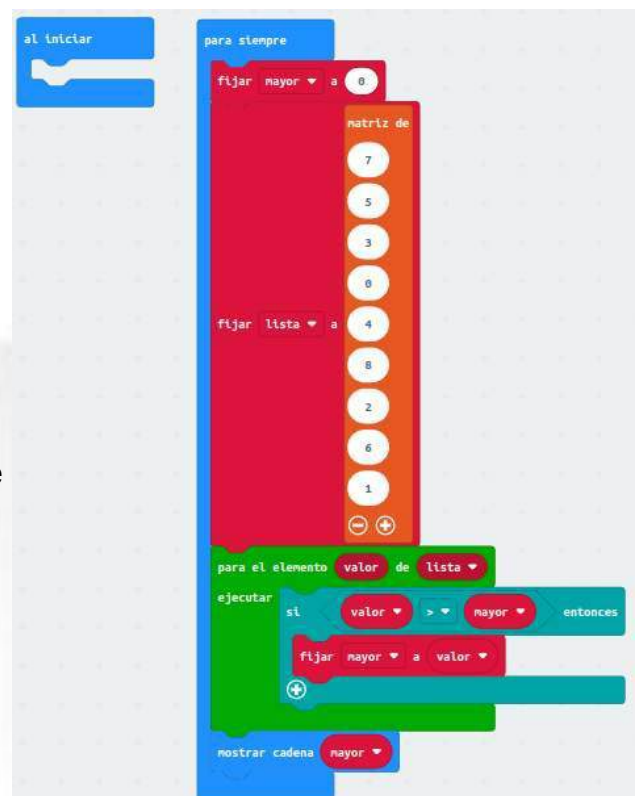
Hemos creado una lista aleatoria de valores y el programa lo que hace es encontrar y mostrar el mayor de ellos.

El funcionamiento del programa lo leemos así:

Se crea una variable mayor para guardar el valor del elemento de la lista de mayor valor, al arreglo le hemos dejado el nombre lista y para recorrerla se utiliza la variable valor.

El condicional va comprobando si $\text{valor} > \text{mayor}$ y mientras sea cierto se guarda en mayor el valor leído y si no es cierto se continúa con el siguiente elemento de la lista.

Una vez finalizado el condicional se muestra la cadena con la variable mayor.



Funciones o Procedimientos

Para repasar:

Las funciones son acciones que **devuelven un valor después de su ejecución**.

Por ejemplo, obtener la longitud de un arreglo o el valor de un elemento en una posición específica son funciones que siempre devuelven un resultado.

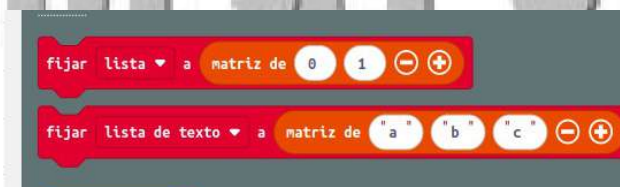
- **Los procedimientos** son acciones que **realizan una tarea pero no devuelven un valor directamente**.

Por ejemplo, establecer un valor o añadir un elemento a un arreglo son procedimientos que cambian el estado del arreglo sin devolver un valor.

Síntesis del menú:

Arreglos - Matrices

- **Fijar**



Descripción: Asigna el valor de la entrada (ya sea numérica o de texto) a una variable en el programa. Es útil para inicializar variables con valores específicos.

- **matriz vacía**



Descripción: Crea una lista o arreglo vacío. Puedes agregar elementos a esta lista utilizando el bloque “añadir valor al final (push)”.

- **longitud del arreglo**



Descripción: Devuelve el número de elementos en un arreglo. Es útil para saber cuántos elementos hay en una lista.

- **obtener el valor en 'x'**



Descripción: Obtiene el valor del elemento en la posición 'x' del arreglo. Permite acceder a un elemento específico mediante su índice.

- **eliminar el valor en 'x'**



Descripción: Elimina el elemento en la posición 'x' del arreglo. El tamaño del arreglo se reduce en uno y los elementos posteriores se desplazan hacia arriba.

- **obtener y eliminar el último valor de (pop)**

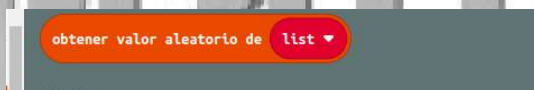


Descripción: Elimina y devuelve el último elemento de un arreglo. Es útil cuando necesitas extraer el último valor de una lista.

- **obtener y eliminar el primer valor de (shift)**

Descripción: Elimina y devuelve el primer elemento de un arreglo. Es útil cuando necesitas extraer el primer valor de una lista.

- **obtener valor aleatorio**



Descripción: Devuelve un valor aleatorio de un arreglo. Es útil para seleccionar elementos al azar.

- **establecer el valor en 'x' a 'nn'**



Descripción: Cambia el valor del elemento en la posición 'x' del arreglo al valor especificado en 'nn'.

- **añadir valor 'nn' al final (push)**



Descripción: Añade un nuevo elemento al final de un arreglo.

- **eliminar el último valor de**



Descripción: Elimina el último elemento de un arreglo y lo devuelve.

- **eliminar el primer valor de**



Descripción: Elimina el primer elemento de un arreglo y lo devuelve.

- **insertar 'nn' al principio (unshift)**



Descripción: Añade un elemento al principio de un arreglo y devuelve la nueva longitud del arreglo.

- **insertar en 'x' valor 'nn' (insertAt)**



Descripción: Inserta el valor 'nn' en la posición específica indicada por 'x', aumentando la longitud del arreglo en uno.

- **eliminar valor en 'x' (removeAt)**



arreglo se reduce en uno.

Descripción: Elimina el elemento en la posición 'x' del arreglo. El tamaño del

- **bloque encontrar índice de (IndexOf)**



coincidencia, devuelve -1.

Descripción: Devuelve la posición o índice de la primera aparición de un valor en un arreglo. Si no se encuentra ninguna

- **invertir (reverse)**



Descripción: Invierte el orden de los elementos en un arreglo.

En MakeCode para micro:bit, los arreglos son especialmente útiles porque **nos permiten:**

- Agrupar datos relacionados, como una serie de lecturas de sensores.
- Acceder a los datos mediante un índice, lo que significa que podemos seleccionar un elemento específico del arreglo utilizando su posición.
- Realizar operaciones en todos los elementos del arreglo con facilidad, como sumar las puntuaciones de un juego.

Los arreglos son una herramienta en programación que nos ayuda a organizar y procesar colecciones de datos de manera eficiente.

Video ejemplo:

<https://www.youtube.com/watch?v=VK6POqoSpwo>