



FICHA 1

LUCES Y CÓDIGOS



Aprendizajes

Al final de esta actividad se espera que puedas:

- Identificar un conjunto de pasos e instrucciones para realizar una tarea.
- Simular la ejecución de ese conjunto de instrucciones y pasos para saber si funcionan bien.
- Manejar el editor **MakeCode** de la **micro:bit** para escribir un programa y simular su funcionamiento.
- Utilizar entradas y salidas de la micro:bit
- Utilizar variables booleanas.
- Reconocer que muchos artefactos tienen dentro un procesador.
- Describir qué es un **programa**, una **persona que programa**, un **procesador**, una **entrada** y una **salida**.

Sesión 1



Qué sabemos, qué tenemos que saber



¿En qué piensas cuando escuchas la palabra computador? Es posible que lo primero que se te venga a la mente es una tableta, un computador portátil o un computador de escritorio. Pero ¿puedes pensar en algo diferente? ¿Quizás un celular? Sigue buscando ejemplos, ¿quizás un reloj? ¿O una lavadora? ¿Qué es un computador entonces?

¿Cómo “sabe” la lavadora cuándo prender el motor para empezar a lavar? O ¿cuándo dejar caer el detergente? Cuando oprimas el ciclo de lavado **suave** por ejemplo, la lavadora “determina” la cantidad de agua, su temperatura, cuántas revoluciones del tambor, cuánto tiempo, entre otras. ¿Cómo logra esto?

Si estás pensando que debe haber un “programa” que cuando eliges el ciclo suave, da las instrucciones para que la lavadora empiece a lavar, escurrir y centrifugar, estás en lo correcto.

Las lavadoras modernas, al igual que los computadores y muchos otros artefactos incluyen **procesadores** que ejecutan instrucciones de un programa desarrollado por **una persona que programa**. Este **programa** incluye instrucciones sobre el tiempo de lavado, la temperatura del agua, el momento de agregar el jabón, entre muchas otras. Los artefactos y electrodomésticos actuales son cada vez más “inteligentes”, pero para ello necesitan que un(a) **programador(a)** haga un programa que debe ejecutar un procesador electrónico.

Algoritmo: secuencia lógica de pasos.

Programa: es una secuencia de instrucciones, escritas para realizar una tarea específica en un procesador.

Programador(a): persona que escribe el programa para un procesador.

Procesador: dispositivo electrónico que entiende esas instrucciones y las ejecuta automáticamente.



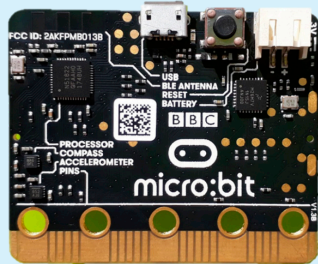
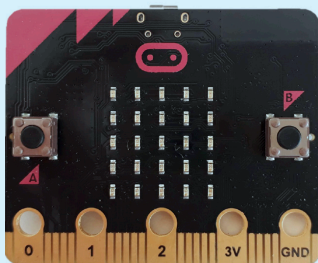


SÍMBOLO	INTRODUCCIÓN
	Tomar y levantar una ficha de la pila de inicio.
	Bajar y soltar la ficha en la casilla actual.
	Mover la ficha una casilla a la derecha.
	Mover la ficha una casilla a la izquierda.
	Mover la ficha una casilla hacia el frente.
	Mover la ficha una casilla hacia atrás.

TABLA 1

				Inicio

Ejemplos de ubicación de fichas



<https://microbit.org/>

Desconectados



En esta sección, jugaremos a ser un **procesador**. Para ello deberemos escribir un **programa** (serie de instrucciones) utilizando los **símbolos-instrucciones** que se muestran en la tabla 1 de la izquierda para llevar objeto por objeto desde la casilla “Inicio” hasta un lugar indicado con el símbolo sin pasar por encima de las serpientes ni de objetos ya colocados, ni colocar un nuevo objeto donde ya se haya puesto otro.

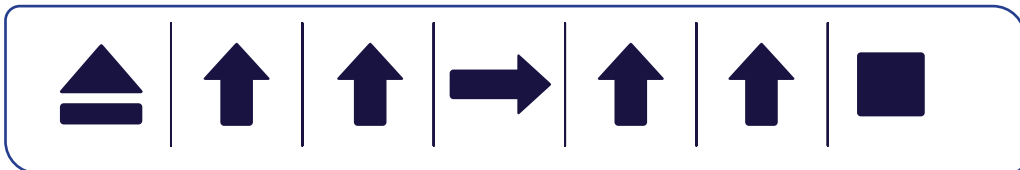
Como objetos pueden usar fichas, monedas u otro objeto similar que se pueda poner en pila. Los objetos comienzan todos en la casilla “Inicio”.

Para jugar, cada uno de los integrantes debe seleccionar uno de los siguientes roles:

- **Cliente:** decidirá donde deben quedar las fichas sobre un tablero (ver anexos) y se la muestra a quien programa (ver ejemplo). El procesador no puede ver la ubicación de las fichas.
- **Programador(a):** deberá escribir sobre una hoja un **programa** utilizando las **instrucciones-símbolos** de la tabla. El programa consiste en una secuencia de estos símbolos que le dirá al **procesador** lo que debe hacer.
- **Procesador** deberá leer el programa y ejecutar las instrucciones para mover y colocar las fichas.
- **Verificador(a):** revisar que las fichas hayan quedado en el lugar indicado por la tarjeta del cliente. Si el grupo es de tres estudiantes la persona llamada **Cliente** hará esta tarea.
- Mientras otra persona está trabajando, observa lo que hace, detecta errores del programa o del procesador y los anótalos para discutirlos luego.

Una vez terminado el ejercicio con una tarjeta, cambiamos los roles y utilizamos otra tarjeta.

Ejemplo de programa:



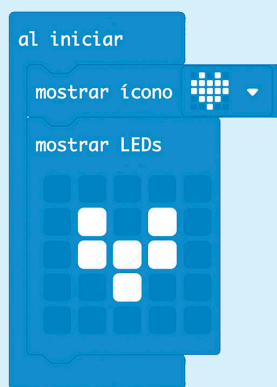
Los **procesadores** de los computadores, robots, lavadoras, celulares y muchos otros dispositivos siguen las instrucciones de un **programa** que ha sido escrito por una persona que se llama **programadora**.

La **micro:bit** tiene un procesador que debe ser programado con un lenguaje especial que aprenderás a utilizar.



Sesión 2

Conectadas: Manos a la Micro:bit

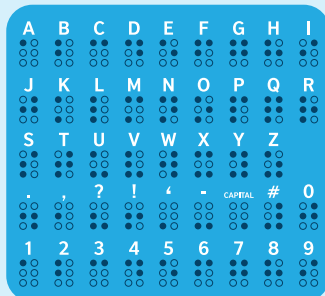


La tarjeta **micro:bit**, la cual se muestra a la izquierda, contiene un **procesador** (*processor*) el cual puede ejecutar las instrucciones de un **programa**, escritas por una persona **programadora** en un **lenguaje** especial, utilizando un **editor** especial que funciona en un computador o en un celular llamado **Make-Code**, el cual vas a aprender a utilizar. Este editor tiene un simulador de la **Micro:bit**, el cual permite **depurar** y **validar el programa**.

Depurar: corregir los errores que hacen que no funcione.
Validar: verificar que el programa hace lo que debe hacer.

Paso a paso, como en un programa para el procesador

1. Vas a ser primero un(a) **programador(a)**. Con la ayuda de tu profesor entra al editor. Deberás seleccionar y verás una pantalla como la siguiente:
2. Elige en la sección de Instrucciones la opción Básico. Aparecerán varios bloques, donde cada bloque representa una instrucción (como las flechas en nuestra actividad anterior)
3. Pon dentro del bloque que dice **para siempre** el bloque **mostrar número 0** y observa lo que pasa en la pantalla de LED. ¿Observas el 0?
4. Ahora vamos a poner a palpitar el corazón de la **micro:bit**. Para ello vas a “escribir” el programa que ves al lado izquierdo, tomando los bloques que requieres de la sección **Básico**.
5. Explora cómo hacer para que los bloques encajen correctamente. Cuando un bloque no encaja correctamente o no está ubicado en la secuencia de instrucciones cambiará.
6. ¡Si ves un corazón que palpita, lo has logrado! Es tu primer programa en la **micro:bit**.
7. Si no lo has logrado, deberías:
 1. Revisar lo que hiciste.
 2. Examinar cómo lo hizo alguna de las otras personas de tu grupo.
 3. Si aún no lo logras, pedirle ayuda al profesor.
8. Este corazón late muy rápido. Quisiéramos que no se fatigue tanto y vaya más lento. Utiliza ahora el bloque para colocar entre palpitaciones.
9. ¿Lograste que palpite más lento? Te has convertido en una persona que sabe programar en la **micro:bit**. De ahora en adelante aprenderás a manejar más bloques para hacer muchas cosas: hacer juegos, resolver problemas y ayudarle a la naturaleza.
10. Explora ahora el menú de comandos y trata de jugar con ellos. Algunos bloques se pueden conectar, otros no.
11. Te invitamos a revisar los diferentes bloques y su forma. Discute con tu colega de grupo sobre qué cosas ves en las diferentes pestañas. ¿Te imaginas para qué sirven algunos comandos- bloques? ¿Algunos te parecen completamente nuevos?
12. Para terminar, te invitamos a cambiar el bloque de **para siempre** por el de **al presionar el botón A**, que encontrarás en el menú en **Entrada**.
 1. ¿Qué sucede cuando se oprime con el ratón en el simulador el botón A una vez?
 2. ¿Qué pasa si lo oprimas muchas veces? ¿Cuántas veces palpita el corazón? Comparte tus hallazgos con tus compañeros y con el profesor. ¿Encontraron lo mismo?



Botón A: es una entrada de la *micro:bit* y es una **variable Booleana**, puede estar oprimido o no.

Variable booleana: toma solo uno de dos valores, **verdadero** o **falso**:

- **El botón B**, que también es una **entrada**, es una **variable booleana**.
- **Botón no oprimido** vale **Falso**, **Botón oprimido** vale **verdadero**
- Cada **LED** es una **salida booleana**: encendido es **verdadero** y apagado es **falso**.



Aplicando lo aprendido

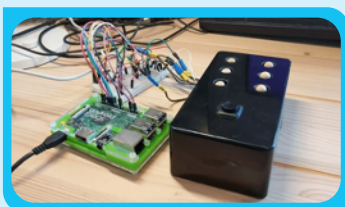
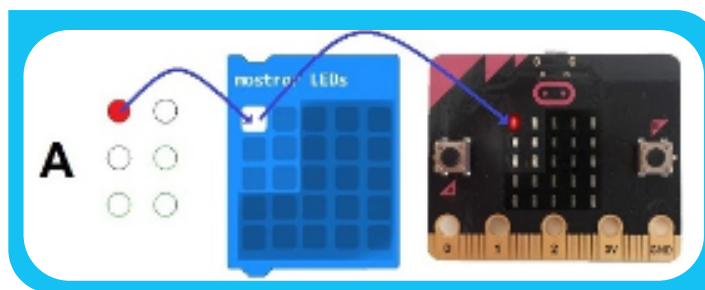


El objetivo del programa es transmitir información a través de un código de luces que se muestra al lado izquierdo. Vamos a utilizar la pantalla de la *micro:bit* (arreglo de LED, luces) para transmitir estos mensajes.

Se usará un arreglo de seis LED para representar cada letra del alfabeto como se muestra en la figura de la izquierda. Los caracteres se despliegan en el arreglo de LED de la *micro:bit* haciendo coincidir la esquina superior izquierda de cada letra con la esquina superior izquierda del arreglo de LED de la *micro:bit* como se ilustra a continuación.

Tu grupo está a cargo de crear un programa que transmita un mensaje por este medio. Para ello es importante tener en cuenta que toda consonante debe mostrarse 3 segundos. El comienzo de cada palabra debe ir separado por comas (,) y cada vocal se debe mostrar respetando los siguientes tiempos:

a: 1 segundo **e:** 2 segundos **i:** 4 segundos **o, u:** 5 segundos.



Si se repite una letra, entre ellas hay que transmitir el símbolo menos (-) para separarlas. Al comienzo y al final del envío de la información debe utilizar el símbolo número (#) por 1 segundo. Comparte con los demás grupos el resultado de tu dispositivo de transmisión, así mismo, ayuda a evaluar cómo funciona el de otros grupos.

Deberás transmitir tu primer nombre, seguido del primer nombre de las otras personas de tu mismo grupo. Cuando esté funcionando, verifica con otro grupo si pueden adivinar los nombres que fueron transmitidos.

La codificación del alfabeto que acabamos de utilizar es la que usan las personas con discapacidad visual para leer usando dispositivos como el que se presenta en la imagen a la izquierda. Estos generan los caracteres de manera que puedan ser leídos a través del tacto.



Para ampliar

En internet podrás encontrar una gran cantidad de sitios con ideas, videos y mucho más sobre programación, actividades desconectadas y programación de la **Micro:bit**.

Puedes comenzar explorar el sitio oficial de la **Micro:bit** en www.microbit.org

También podrás utilizar algunos aplicativos en línea como el siguiente, que tienen un lenguaje parecido al que utilizaste en la actividad desconectada:

www.lightbot.com

Hoy en día profesionales de todas las áreas utilizan la programación para resolver problemas en: biología, psicología, ingeniería, matemáticas y física. Muchas de las profesiones STEM (Ciencia, Tecnología, Ingeniería y Matemáticas) han contribuido al desarrollo de los dispositivos que usan **procesadores** y a los **programas** que hacen que dichos dispositivos sean muy útiles.

Desarrollar el pensamiento computacional, que es lo que estás haciendo, es aprender algo que te será muy provechoso para tu futuro dado que cada vez más artefactos y procesos utilizan procesadores. Por ejemplo, los autos que se conducen automáticamente tienen muchos **procesadores** programados por **programadores** o **programadoras**.



Para ir más lejos

¡Fantástico! Ahora te proponemos un reto más complejo para que hagas el programa para una lavadora automática.

Una lavadora típicamente tiene varios pasos como agregar agua, luego jabón, enjuagar por un tiempo, luego sacar el agua, cargar de nuevo agua y enjuagar. El reto del grupo es crear símbolos que representen cada etapa con el arreglo de **LEDs** y simular el funcionamiento de la lavadora mostrando esos símbolos en la pantalla. Los minutos de un paso se representa por los segundos en que la **micro:bit** muestra el ícono respectivo. Es decir que, por ejemplo, si el lavado toma 30 minutos, se mostrará el ícono de lavado 30 segundos.

Busca otro grupo y pídele que observe tu simulación y trate de decir qué está haciendo la lavadora y por cuanto tiempo hace cada etapa.



Qué hemos aprendido

Revisa y completa la siguiente tabla marcando con una X en la columna que mejor represente tu aprendizaje:

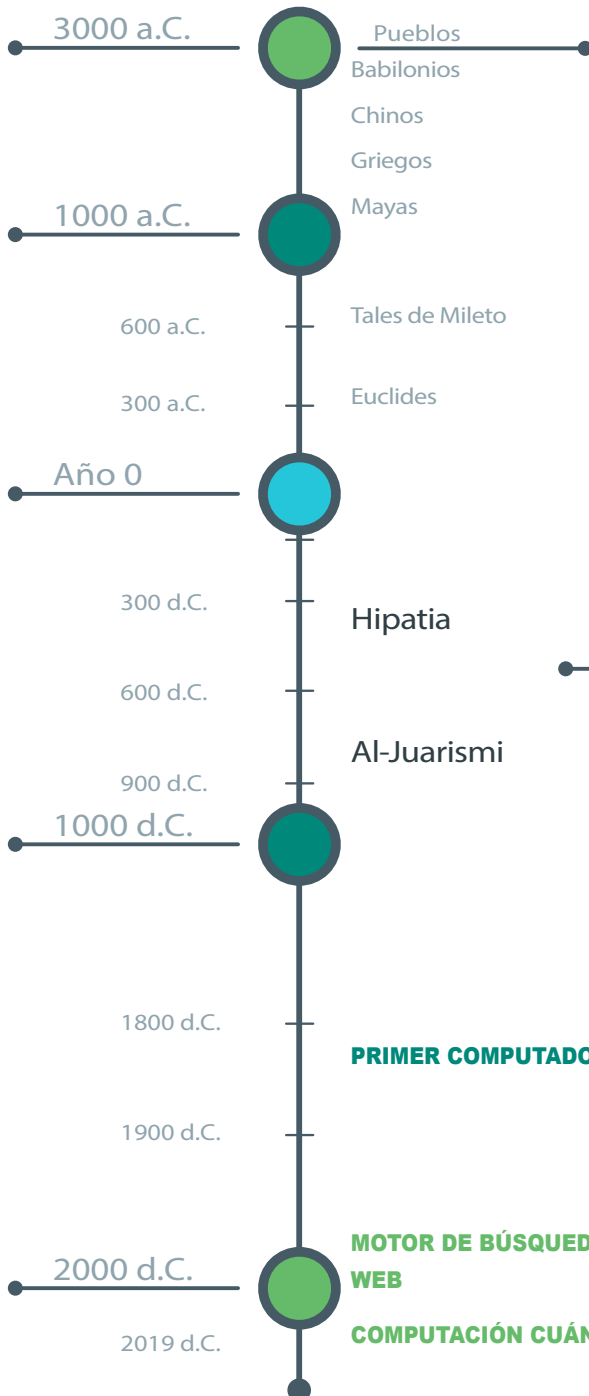
Verifica los aprendizajes logrados	Sí	Algo	No
Identifico y escribir un conjunto de pasos e instrucciones para realizar una tarea.			
Simulo la ejecución de ese conjunto de instrucciones y pasos para saber si funciona bien.			
Manejo el editor MakeCode de la micro:bit para escribir un programa y simular su funcionamiento.			
Utilizo variables booleanas.			
Reconozco que muchos artefactos tienen dentro un procesador.			
Describir qué es un programa, una persona que programa, un procesador, una entrada y una salida.			

Selecciona la opción que mejor represente tu opinión:

Contesta las siguientes preguntas	Sí	Algo	No
Las actividades realizadas fueron difíciles.			
Las actividades me motivaron.			
Siento que aprendí muchas cosas.			
Aún me quedan muchas dudas sobre lo que hice.			



Un poco de historia



Las bases del pensamiento computacional vienen desde la antigüedad y las mujeres han tenido un papel importante.



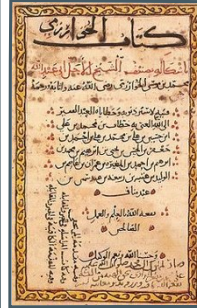
Hipatia (350~370d.C. - 415 d.C.)
 Fue una filósofa, matemática y astrónoma que vivió a comienzos del siglo V en Alejandría (hoy Egipto). Creó un algoritmo desconectado para la división. Mejoró un instrumento de cómputo de la antigüedad, el astrolabio, que en griego significa «buscador de estrellas» y se usa para determinar la posición y altura de las estrellas. Fue una pionera de las mujeres en las Ciencias. Murió víctima de la persecución religiosa al conocimiento científico y a la libertad de pensamiento.



Imagen de un astrolabio








Al-Juarismi (780 d.C. - 850 d.C.)
 Su nombre completo es Abu Abdallah Muḥammad ibn Mūsā al-Jwārizmī (Abu Yāffar) (رفعج وپا یم زراوخل ا یسوم زب نمم هلد ا دبج وپا). Fue un matemático, astrónomo y geógrafo musulmán que vivió en Persia (hoy Irán). Creó un algoritmo desconectado para resolver ecuaciones cuadráticas y lo publicó en su libro Álgebra. Este libro es considerado el primer libro de álgebra por lo que se le considera el fundador del álgebra como disciplina. Introdujo nuestro sistema de numeración denominado arábigo. De su nombre se deriva la palabra algoritmo.



Álgebra de Al-Juarismi










TARJETA UNO

				
				
				INICIO










TARJETA DOS

				
				
				
				INICIO



TARJETA TRES

				
				
				
				
				INICIO