Invernaderos

Grado 6º Guía 3

TIC



Estudiantes

Apoya:

Educación

BRITISH COUNCIL

Colombia Programa
{EL CÓDIGO A TU FUTURO}

Invernaderos

Grado 6º Guía 3

TIC

Estudiantes

Apoya:

Educación

BRITISH COUNCIL

Colombia Programa
{EL CÓDIGO A TU FUTURO}

MINISTERIO DE TECNOLOGÍAS DE LA INFORMACIÓN Y LAS COMUNICACIONES

Yeimi Carina Murcia Yela
Ministra TIC

Giovanny Andrés López Cabezas
Viceministro de Transformación Digital

Gloria Patricia Perdomo Rangel
Viceministra de Conectividad

William Alexander Sánchez Martínez
Director (e) de Apropiación de TIC

Lorena Paola Fortich Tulena
Jefe de la Oficina Asesora de Prensa

Equipo Comunicaciones DATIC
Manuel Alejandro Guerrero Aponte
Carlos Andrés Méndez Silva

Equipo Técnico
Cristhiam Fernando Jácome Jiménez
Miller Jimmy Alarcón
Ricardo Cañón Moreno

BRITISH COUNCIL COLOMBIA

Felipe Villar Stein
Director de país

Laura Barragán Montaña
Directora de programas de Educación, Inglés y Artes

Marianella Ortiz Montes
Jefe de Colegios

Óscar Daniel Barrios Díaz
Jefe de Implementación Colombia Programa

Equipo operativo
Juanita Camila Ruiz Díaz
Jesica Lubrano Lavadera
Dayra Maritza Paz Calderón
Saúl Fabián Torres
César Augusto Herrera Lozano
Myriam Nathalia Salinas Tarazona
Paula Álvarez Peña
Elizabeth Marín Giraldo

Equipo técnico
Ana Lorena Molina Castro
Alejandro Espinal Duque
Vanesa Abad Rendón
Raisa Marcela Ortiz Cardona
Juan Camilo Londoño Estrada

Edición y revisión técnica de versiones finales
Alejandro Espinal Duque
Ana Lorena Molina Castro
Heidy Esperanza Gordillo Bogota
Juan Camilo Londoño Estrada

Apoyo en la edición y revisión académica y técnica (versión 1)
Vanesa Abad Rendón
Raisa Marcela Ortiz Cardona

Consultora experta
Heidy Esperanza Gordillo Bogota

Edición
Juanita Camila Ruiz Díaz
Alexandra Ruiz Correa

BRITISH COMPUTER SOCIETY (BCS) – Consultoría internacional

Julia Adamson
Directora General de Educación y Beneficio Público

Niel McLean
Jefe de Educación

Claire Williams
Coordinadora de Alianzas

ASOCIACIÓN COLOMBIANA DE FACULTADES DE INGENIERÍA - ACOFI

Edición general
Mauricio Duque Escobar

Coordinación pedagógica
Margarita Gómez Sarmiento
Mariana Arboleda Flórez
Rafael Amador Rodríguez

Coordinación de producción
Harry Luque Camargo

Asesoría estrategia equidad
Paola González Valcárcel

Asesoría primera infancia
Juana Carrizosa Umaña

Autoría
Alex González Jurado
Arlet Orozco Marbello
Harry Luque Camargo
Isabella Estrada Reyes
Lucio Chávez Mariño
Margarita Gómez Sarmiento
Mariana Arboleda Flórez
Mauricio Duque Escobar
Paola González Valcárcel
Rafael Amador Rodríguez
Rocío Cardona Gómez
Saray Piñerez Zambrano
Yimzay Molina Ramo

.Puntoaparte Editores

Diseño, diagramación, ilustración, y revisión de estilo

CC BY-NC 4.0

Esta obra se encuentra bajo una Licencia Creative Commons Atribución-No Comercial 4.0 Internacional. https://creativecommons.org/licenses/by-nc/4.0/

"La presente guía corresponde a una versión modificada del material originalmente diseñado por el equipo de autores de ACOFI, en tanto incorpora ajustes, actualizaciones y contribuciones posteriores realizadas por el equipo del British Council y el Ministerio de Tecnologías de la Información y las Comunicaciones, bajo la autorización correspondiente, para su publicación final".

ISBN: 978-628-7934-71-9

Prólogo

Ministerio de Tecnologías de la Información y las Comunicaciones

Estimadas y estimados educadores, estudiantes y comunidad educativa:

Colombia es un país de talento y de oportunidades que no siempre han estado al alcance de todas y todos. Entre ellas, la tecnología, que en las zonas más apartadas aún se percibe como un privilegio lejano.

Con la convicción de conectar territorios y cerrar brechas, emprendimos una transformación en el Ministerio TIC para acercarlo más a las regiones y a sus realidades.

Creemos en una tecnología que llegue con propósito a la vida de las personas, que permita cerrar brechas, abrir caminos y acompañar el desarrollo de las comunidades en todo el país.

Hoy, en ese camino, en el que ya hemos formado a cerca de un millón de colombianas y colombianos en habilidades digitales, presentamos las **Guías de Pensamiento Computacional de Colombia Programa**, una apuesta concreta para llevar estas oportunidades a las aulas del país.

Este material es el resultado de un trabajo colectivo en el que más de 2.000 docentes de 420 sedes educativas aportaron su experiencia y conocimiento para su construcción.

Además, contó con la asesoría de la *British Computer Society (BCS)*, entidad del Reino Unido con más de 20 años de experiencia en la incorporación del pensamiento computacional en el currículo educativo, gracias al convenio de cooperación internacional suscrito con el British Council Colombia.

Estas guías trazan una ruta pedagógica para que niñas, niños y jóvenes desarrollen nuevas formas de pensar, aprendan a resolver problemas y construyan soluciones. Son un recurso pensado para acompañar la labor de los y las docentes que enseñan pensamiento computacional.

Su uso requiere la mirada y la adaptación de quienes enseñan, teniendo en cuenta el contexto y las particularidades de cada estudiante.

Con ellas buscamos que las nuevas generaciones pasen de ser usuarias de la tecnología a protagonistas de su creación y transformación; que puedan comprenderla, dominarla y ponerla al servicio de sus propias vidas y de sus comunidades.

Queremos que, sin importar el lugar donde se estudie, en una zona rural, en la Amazonía o en una ciudad, existan las mismas oportunidades para desarrollar talento digital.

El gobierno del presidente Petro le cumplió a los colombianos y las colombianas y, por eso, buscamos que más niñas y jóvenes se reconozcan como creadoras, y que cada docente cuente con herramientas para transformar su aula.

Porque cuando conectamos territorios, abrimos oportunidades para que Colombia avance con equidad, desarrollo y justicia social.

Invitamos a estudiantes, docentes y comunidades a apropiarse de estas guías, a disfrutarlas y a llevarlas a su vida diaria. Porque cuando la educación digital llega a los territorios, no solo transforma el aprendizaje: transforma comunidades enteras y abre caminos para construir un mejor país.

Carina Murcia Yela

Carina Murcia Yela
Ministra de Tecnologías de la Información y las Comunicaciones
Gobierno de Colombia

Prólogo

British Council

Por más de 87 años, el *British Council* ha tenido presencia en el país, buscando construir lazos de confianza y cooperación entre el Reino Unido y Colombia. Uno de los mejores ejemplos de lo que podemos lograr a través de estas alianzas es *Colombia Programa*. Esta iniciativa, que desde hace tres años y de la mano con el Ministerio de Tecnologías de la Información y las Comunicaciones nos ha permitido alcanzar un objetivo ambicioso: ofrecer espacios de formación y recursos de alta calidad para que docentes y estudiantes de los colegios públicos del país desarrollen sus habilidades digitales y el pensamiento computacional con un enfoque de género.

Esta publicación marca un hito particularmente importante, pues con ella entregamos al país un material sin precedentes en América Latina y comparable con muy pocos a nivel mundial. Se trata de 79 guías pedagógicas que suman más de 4.000 páginas y contienen 30 sesiones por cada grado, desde transición hasta undécimo. Con estas, maestras y maestros podrán facilitar al menos una hora de clase semanal orientada al desarrollo del pensamiento computacional de sus estudiantes. Basta preguntarse qué otra área o asignatura en Colombia cuenta con un material semejante —en términos de amplitud, rigurosidad y acceso gratuito— para dimensionar la relevancia de este esfuerzo para el sector educativo y tecnológico del país.

Al abrir estas páginas impresas o navegar su versión digital, encontrarán una gran variedad de actividades que proponen a niños, niñas y jóvenes nuevas formas de acercarse a los temas más relevantes del presente, como el uso de la inteligencia artificial desde un enfoque ético y seguro. También verán cómo esta y otras temáticas se articulan con diversas áreas del conocimiento, promoviendo el trabajo colaborativo entre docentes y estudiantes y convirtiendo el pensamiento computacional en un puente entre las matemáticas, las ciencias, el arte y el lenguaje.

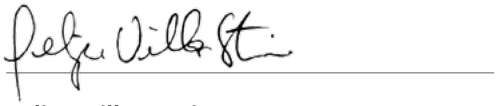
Encontrarán también en estas páginas orientaciones y sugerencias sobre cómo adaptar las actividades a diferentes contextos y necesidades. Por ejemplo, cómo desarrollar habilidades de pensamiento computacional en instituciones sin conectividad o incluso sin electricidad; cómo fomentar la equidad de género para que más niñas y jóvenes se interesen por la programación y las áreas STEM; o cómo ajustar ciertas actividades para que estudiantes en condición de discapacidad puedan participar plenamente.

Lo que quizá no sea visible a primera vista es el arduo proceso que hizo posible llegar a este punto. Un equipo de 25 personas se dedicó durante más de un año a diseñar y revisar cada una de las 79 guías. Contamos, además, con la asesoría de la *British Computer Society (BCS)*, a través de expertos que participaron y lideraron reformas curriculares en Inglaterra en el área de ciencias de la computación. Asimismo, realizamos 38 mesas de validación en cinco regiones del país, en las que 1.755 docentes revisaron y comentaron los primeros borradores. A esto se suman las innumerables horas que más de 2.100 docentes "pares expertos", pertenecientes a los 420 Nodos de Desarrollo de Pensamiento Computacional en todo el país, dedicaron en 2025 a pilotear las guías en sus aulas, asegurando así su pertinencia para los contextos educativos reales.

A cada uno de estos maestros y maestras, directivas docentes, mentores(as), tutores, expertos(as), académicos(as) y funcionarios(as) del *British Council* y de los Ministerios de las TIC y de Educación, así como a los y las estudiantes que nos acompañaron en este proceso, infinitas **gracias** por permitir que hoy Colombia cuente con esta herramienta tan poderosa.

Desde el *British Council*, sabemos que estas guías de pensamiento computacional abrirán nuevas posibilidades para que, a través del trabajo y esfuerzo de docentes en el aula, los y las estudiantes accedan a nuevas formas de pensar, de interactuar con la tecnología y de desarrollar capacidades alineadas con las demandas del mercado laboral presente y futuro. De lograrlo, conseguiremos el sueño que ha guiado esta labor y la alianza de *Colombia Programa*: que cada niña, niño y joven en Colombia pueda construir el futuro digital y material que desee.

Esperamos que las disfruten y que juntos podamos aprovechar al máximo su potencial.



Felipe Villar Stein
Director
British Council Colombia

El camino recorrido para llegar a estas guías

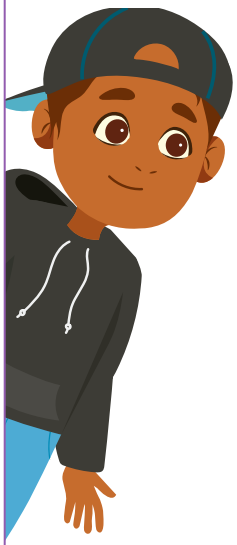
Creación

1 **8 de noviembre de 2023**
El Ministerio TIC y el *British Council* firman el convenio 1247 que da inicio a Colombia Programa.

Diciembre de 2023
La Asociación Colombiana de Facultades de Ingeniería (ACOFI) empieza a trabajar en la matriz de alcance y secuencia.

2 **Febrero de 2024**
Se aprueba la matriz de alcance y secuencia de las guías creadas por ACOFI, que incorpora retroalimentación de la *British Computer Society*. Con esta aprobación se inicia la creación de las guías de pensamiento computacional de Colombia Programa.

Validación y pilotaje



3 **Mayo a octubre 2024**
1.755 docentes en 38 mesas de trabajo de las 5 regiones del país, validan la versión inicial de las guías y proponen ajustes para hacerlas más adaptables e inclusivas.

Mayo a diciembre 2024
Se incorporan a las guías los ajustes provenientes de las mesas de validación.

4 **Agosto 2024 a mayo 2025**
Diagramación e impresión de las guías para entrega a los nodos de pensamiento computacional.

Marzo a septiembre 2025
2.100 docentes de los 420 nodos prueban las guías en sus aulas. Se recopilan sugerencias de ajustes a partir de este ejercicio.

Octubre a noviembre 2025
Análisis de resultados y recomendaciones para ajustes de la versión final de las guías.

Ajustes finales



5 **Diciembre 2025 a abril 2026**
Implementación de ajustes finales.

Enero a junio 2026
Diagramación de la versión final de las guías de pensamiento computacional de Colombia Programa.



Grado 6° Guía 3

TIC

Identificación de los aprendizajes

- Algoritmos, patrones, abstracción y descomposición
- Lógica, programación y depuración
- Computación física
- Prácticas de datos

Aprendizajes de la guía

Con las actividades de esta guía se espera que puedas avanzar en:

- Explicar el flujo de los algoritmos, cómo cambian las variables durante su ejecución y anticipar los resultados.
- Definir y utilizar variables numéricas para organizar información y realizar cálculos.
- Utilizar entradas y salidas de un dispositivo digital para resolver retos.

Resumen de la guía

Esta guía propone 5 sesiones de trabajo orientadas a avanzar en el desarrollo de habilidades de programación por bloques utilizando el editor *MakeCode*, con particular énfasis en la definición y uso de variables para almacenar y procesar información. Igualmente, se incorpora el uso de los sensores que trae la *micro:bit* y que se pueden simular en el editor de *MakeCode*.

Resumen de las sesiones

Sesión 1

Se continúa aprendiendo a utilizar los sensores de la *micro:bit* de forma simulada en *MakeCode* y a visualizar las lecturas que se obtienen de estos sensores.

Sesión 2

Se aprende a definir variables, a guardar en ellas información y a recuperarla para visualizarla.

Sesión 3

Se continúa aprendiendo a leer señales de sensores y se aprende a manejar los pines y a ajustar esas lecturas a una escala.

Si se requiere

En la Guía 1 de grado 5º se aborda por primera vez el uso de *MakeCode*.

Sesión 4

Se aprende a realizar operaciones aritméticas con los valores guardados en las variables.

Sesión 5

Se resuelve el reto propuesto con los conocimientos logrados en las sesiones anteriores.

Nota

Esta guía es una adaptación de la ficha “Invernaderos” ,desarrollada en el 2020 por ACOFI para el programa *Coding for Kids*, en el marco del convenio 838 entre el Ministerio TIC, Computadores para Educar y el *British Council*, y editada en el marco de los convenios 764 de 2021 y 698/002 de 2022 suscritos entre el MEN, el Ministerio TIC y el *British Council*.

Conexión con otras áreas

A continuación, se presenta la conexión con otras áreas:

Matemáticas

- En esta guía tendrás la oportunidad de utilizar saberes previos en matemáticas, particularmente al realizar operaciones aritméticas con valores y al representar datos.

Grado 6° Guía 3

TIC

Sesión 1

Aprendizajes esperados

Con las actividades de esta sesión se espera que puedas avanzar en:

- Usar *MakeCode* para tomar lecturas de algunos sensores de la *micro:bit*.
- Visualizar valores e íconos en la pantalla de LED de forma clara, en función de los datos medidos.

Duración sugerida



40% 40% 20%



Material para la sesión

Material por grupo:

- Dispositivo con acceso a *MakeCode*.
- Anexo 1.1.

Anexo

Anexo 1.1



Actualmente muchas plantas se cultivan en invernaderos especialmente acondicionados para asegurar condiciones controladas de temperatura, humedad y algunas veces de radiación solar.

Para mantener las condiciones bajo control es necesario medir estas variables físicas (humedad, temperatura y radiación solar), en algunos casos simplemente para indicar que existe un problema para que las personas a cargo tomen acciones, o en casos más automatizados, para que el sistema corrija automáticamente el problema.

Por ejemplo, si la humedad es muy baja, será necesario regar el cultivo.

El reto que se te propone es diseñar un programa prototipo de invernadero automatizado, que funcione con la micro:bit para que se pueda probar en simulación usando MakeCode. Este prototipo debería cumplir con las siguientes especificaciones:

- Tomar datos de temperatura, humedad y nivel de luz cada hora (tengan en cuenta que cada hora se simulará como un minuto).
- Al oprimir el botón B su programa debe comenzar a tomar datos.
- Al oprimir el botón A debe presentar los datos máximos, mínimos y el promedio de las variables hasta el momento.
- Si la humedad es inferior a un valor definido debe hacer sonar una alarma para avisar que es necesario regar el cultivo. Este valor debe poderse ajustar por programa, dado que puede variar de cultivo a cultivo.

Recuerda que al usar *MakeCode* puedes utilizar herramientas de accesibilidad si las requieres. Tu docente puede orientarte al respecto.

Lo que sabemos, lo que debemos saber

Esta sección corresponde al 40% de avance de la sesión

Empieza por conocer el reto que te proponemos en esta guía, ver *Anexo 1.1*, y que deberás poder resolver en la sesión 5. Recuerda que este reto se resuelve al final y en esta guía aprenderás lo que requieres para hacerlo.

En la primera guía de este grado ya trabajaste con el editor de *MakeCode* y con la programación en bloques, que es uno de los lenguajes de programación de la tarjeta *micro:bit*.

Recordemos que esta es una tarjeta que tiene un pequeño procesador (o microprocesador) capaz de ejecutar un programa que le demos, realizar cálculos a gran velocidad y almacenar información. Además, cuenta con entradas que le permiten capturar información, como los **sensores** que trae o los puntos de contacto (pines) donde se pueden conectar dispositivos externos. Igualmente, puede visualizar información en una cuadrícula o matriz de LED y emitir sonidos.

Figura 1. *Micro:bit* v2 y sus componentes

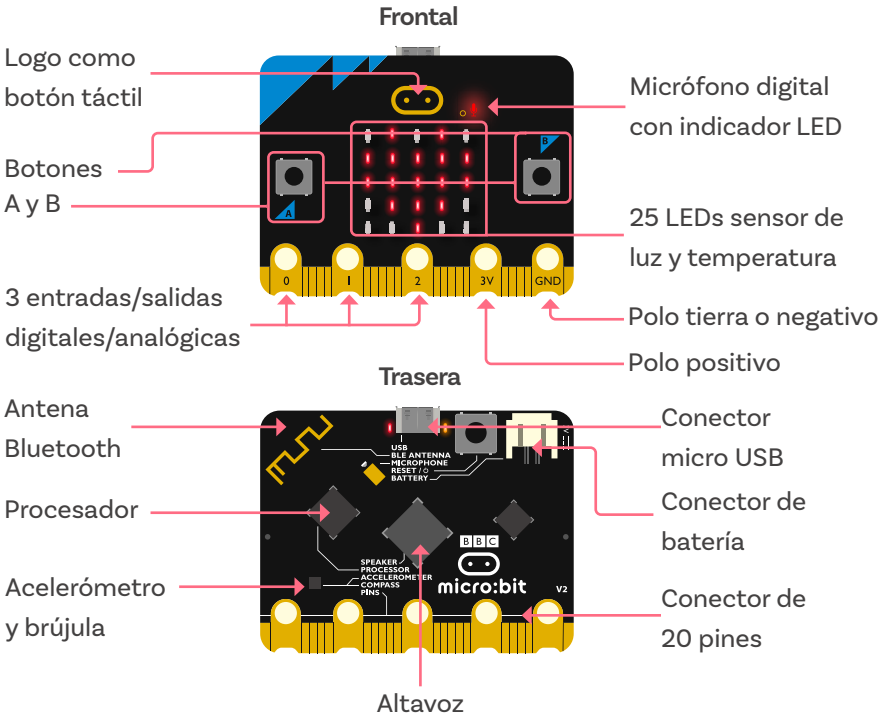
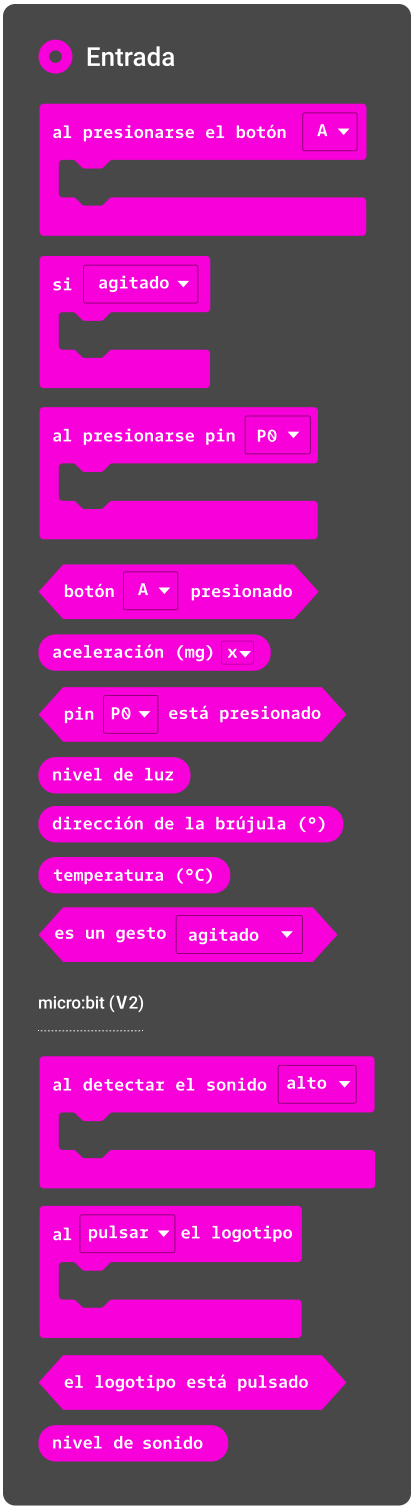


Figura 2. Bloques de entrada en *MakeCode*



Para utilizar los sensores, *MakeCode* propone un menú que se puede observar en la *Figura 2*.

Primero, podrás observar bloques que permiten lanzar un programa si algo sucede:

Al presionar el botón: lanza el programa que contenga si se oprime el botón A, el B o ambos al tiempo A+B.

Si agitado: lanza el programa contenido en el bloque si se agita la *micro:bit*.

Al presionarse pin: lanza el contenido del bloque si en un pin de entrada aparece un valor.

Adicionalmente, se presentan condiciones que podrán ser verdaderas o falsas dependiendo de si la condición se cumple o no:

- Botón A, B o A+B presionados (simultáneamente).
- Hay aceleración.
- Pin presionado.

Luego, se encuentran instrucciones de lectura del valor en una **entrada**:

- Nivel de luz.
- Dirección de la brújula.
- Temperatura.

Finalmente, se presentan bloques para el manejo de otras entradas:

- Bloque que activa un programa si hay sonido o no.
- Bloque que activa un programa si se toca el logotipo de la *micro:bit*.
- Condición de oprimido o no del logotipo.
- Nivel de sonido recibido.

Estas entradas permiten recoger información del entorno, pero existen otras posibilidades que podrás conocer cuando revises más adelante el manejo de los pines.

Ahora es momento de explorar las entradas de la *micro:bit* y usarlas en pequeños programas.

Enlace

Recuerda que puedes acceder al editor *MakeCode* en:

Glosario

- **Sensor:** dispositivo que es capaz de medir una variable física, como la temperatura, la cantidad de luz o la orientación en el campo magnético de la Tierra, por ejemplo.
- **Entrada:** corresponde a un medio por el cual un sistema de computación recoge información, por ejemplo, el teclado de un computador o los botones y sensores de la *micro:bit*.
- **Procesador:** unidad central de procesamiento de un programa y de los datos que maneja. Todo computador o dispositivo computarizado cuenta con un procesador.

Manos a la obra

Conectadas

Esta sección corresponde al 80% de avance de la sesión

En esta sesión vas a trabajar resolviendo varios retos cortos que te irán ayudando desde ahora a ganar habilidades para resolver el reto de invernadero que se plantea para el final de la guía.

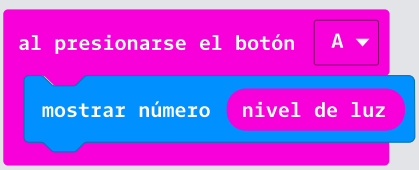
El primer reto es un ejemplo de lo que harás y el resto tendrás que resolverlos de forma más autónoma, haciendo los ajustes necesarios.

Trabajen en parejas, siguiendo las indicaciones de su docente.

Reto 1: revisen el programa de la *Figura 3*.

¿Qué creen que hace?

Figura 3. Código para el reto 1



Una vez analicen este pequeño programa y tengan una idea de lo que hace, introdúzcanlo en *MakeCode* y verifiquen su predicción.

Al editar este pequeño programa, igualmente verán un círculo amarillo y negro en el simulador de la *micro:bit* que aparece a la izquierda de la pantalla del *MakeCode*, tal como lo observan en la *Figura 4*.

Figura 4. Simulador de la *micro:bit*



Ejecuten el programa oprimiendo el botón A. En la parte superior izquierda de la imagen de la *micro:bit*, en el simulador verán que el círculo amarillo y negro tiene el valor de 128, que aparece por defecto. Podrán bajar o subir con el cursor para modificar los valores en una escala que va del 0 al 255, donde 255 representa el valor máximo de luz y 0 sería un ambiente completamente oscuro.

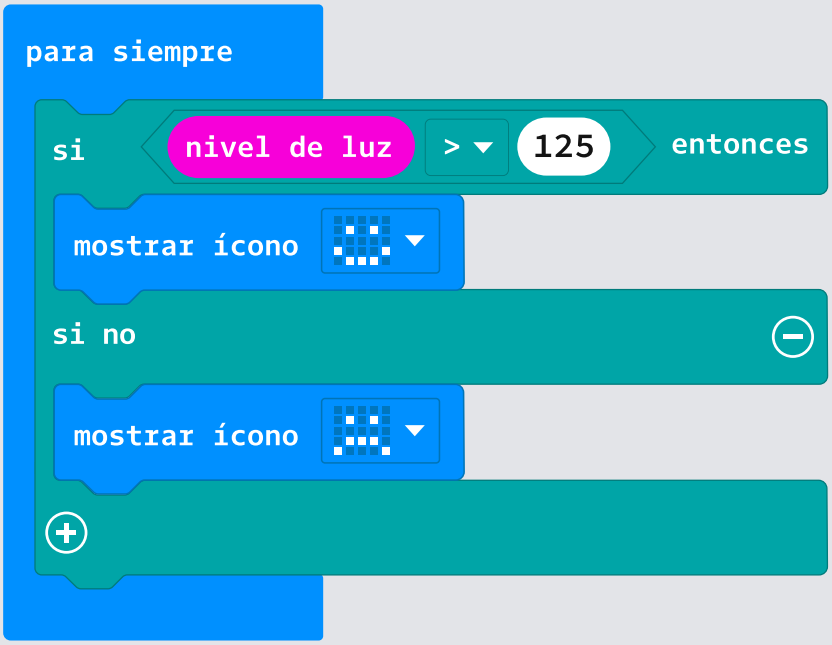
Ahora que ya comprenden el programa, cambien los valores de nivel de luz e imaginen en qué momento del día podrían cambiar estas medidas si correspondieran a valores reales tomados, por ejemplo, en el área de juegos de la escuela.

Observen que la pantalla muestra el valor escogido con el cursor en el ícono de iluminación, ver *Figura 4*.

Reto 2: analicen el programa de la *Figura 5*.

¿Qué esperan que haga?
¿Deben usar un botón para que se ejecute este código?
¿Cuándo funcionaría?

Figura 5. Código del reto 2



Una vez hayan analizado este programa, no duden en probarlo y verificar su uso cambiando los valores de los niveles de luz en el simulador. Observen cuándo aparece un ícono u otro.

¿Por qué podría ser útil en un invernadero el conocer el nivel de luz al que se exponen las plantas?
¿Cómo podría servir un programa como el que analizaron para apoyar el cuidado de las plantas del invernadero?

Anoten sus respuestas.

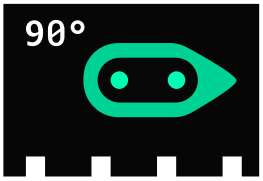
Nota

La dirección de la brújula en la *micro:bit* arroja un valor entre 0 y 359. Puedes asumir para este ejercicio que 0 o 359 son el norte.

La instrucción para leer la brújula es:

```
dirección de la brújula (°)
```

En el simulador de *MakeCode* la dirección se muestra encima y a la derecha de este ícono. Este ícono se puede girar para cambiar la dirección.



Si al tratar de resolver el reto 3 sienten que se bloquean y no pueden avanzar, pidan permiso a su docente para levantarse y observar de forma silenciosa el trabajo de otros grupos. Esto podría inspirarles y ayudarles a continuar.



Reto 3: a continuación, lean la siguiente situación.

Marianella y Saúl están junto con sus familiares de paseo por la Sierra Nevada de Santa Marta. Como allí las temperaturas pueden subir o bajar dependiendo del lugar específico que estén recorriendo, Marianella y Saúl creen que con la micro:bit se puede crear un dispositivo que les ayude a ubicarse y también a proteger a todas las personas de su familia de los cambios climáticos extremos. Por tanto, han determinado que necesitan un programa que cumpla cada uno de los siguientes ***requerimientos****:*

- *Al oprimir el botón B debe mostrar la temperatura.*
- *Al oprimir el botón A debe mostrar una cara alegre si la temperatura es superior a 15 grados, de lo contrario, debe mostrar una cara triste. Si la temperatura es mayor a 35 grados centígrados, debe mostrar un ícono que refleje mucho calor.*
- *En forma continua, debe tomar medición de la orientación de la brújula e indicar siempre la dirección norte, sin mostrar nada en la matriz de LED si se apunta hacia otra dirección diferente al norte.*

Su reto es crear el programa que necesitan Marianella y Saúl.

Cuando terminen su programa pueden proponer mejoras al código de mostrar la dirección de brújula en la *micro:bit*, pues como quizás lo comprobaron, mostrar exactamente el norte no es siempre fácil.

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 usar *MakeCode* para tomar lecturas de algunos sensores de la *micro:bit*?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

2. ¿Puedes visualizar valores e íconos en la pantalla de LED de forma clara en función de los datos medidos?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

> **Si tus respuestas fueron "Parcialmente" o "Aún no", vuelve a las actividades propuestas en *MakeCode*. Luego, discute con tu compañera o compañero de grupo lo que se hizo en cada momento de la actividad y el código de cada uno de los retos. Si todavía te quedan dudas, consúltale a tu docente.**

Ahora, en grupos de 4 personas, según la instrucción de tu docente, sigan la rutina llamada Pensar, Presentar e Integrar (P-P-I). Recuerden que para llevarla a cabo deben:

- ◯ Primero responder las preguntas propuestas individualmente.
- ◯ Después, cada persona en el grupo y en su turno le presenta al resto del equipo sus respuestas.
- ◯ Finalmente, el grupo integra una respuesta unificada que podrán comunicar a la clase.



Las preguntas para discutir son las siguientes:

Si consideran el reto del invernadero automatizado, ¿para qué serviría lo aprendido en esta sesión?

Como vieron, algunas medidas que toma la *micro:bit* se convierten en un número que tiene sentido. Por ejemplo, la temperatura se entrega en grados centígrados y la orientación de la brújula de 0 a 359 grados. Pero otras medidas se convierten en una escala que no es típica, por ejemplo, el nivel de iluminación corresponde a un número entre 0 y 255. Si se quiere convertir este valor a una escala de 0 a 100% ¿qué podrían hacer?



Aprovecha este espacio final para hacerte un esquema en el que resumas algo de lo que aprendiste, por ejemplo, colocando un ejemplo de programa que hiciste y lo que mostraba, como, el resultado del reto 3, además de algunas definiciones y palabras nuevas que aprendiste.

Actividad opcional para seguir consolidando aprendizajes

Completen las siguientes frases en un papelito por grupo y luego lo entregan a su docente. No es necesario que escriban sus nombres.

- ○ Lo más divertido que hicimos hoy fue...
- ○ Lo más difícil fue...
- ○ Hoy aprendimos que...
- ○ Sentimos que todavía necesitamos...

Grado 6° **Guía 3**

TIC

Sesión 2

Aprendizajes esperados

Con las actividades de esta sesión se espera que puedas avanzar en:

- Definir y asignar valores a variables en un lenguaje de programación en bloques.
- Visualizar el contenido de las variables por medio de la programación de contadores.

Duración sugerida



40% 40% 20%



Material para la sesión

Material por grupo:

- Dispositivo con acceso a *MakeCode*.

Lo que sabemos, lo que debemos saber

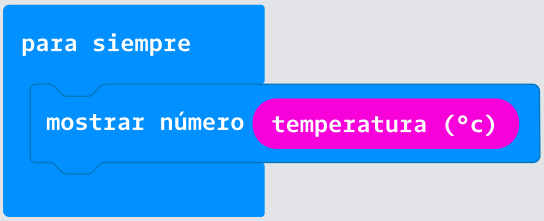
Esta sección corresponde al 40% de avance de la sesión

En la sesión anterior aprendiste a tomar algunas medidas físicas con la *micro:bit* (en simulación por ahora) y a mostrar en la pantalla de LED un dato o un ícono según lo leído.

Trabajaste con mediciones de dirección, nivel de luz y temperatura. Con estos datos se ejecutaron acciones inmediatas, como colocar una cara feliz o triste o mostrar el dato.

La *micro:bit* tiene otros *sensores* que podrías explorar, como movimiento o nivel de sonido. Te dejamos experimentar con ellos autónomamente. Por ejemplo, con estos bloques podrías ver la temperatura detectada por la micro:bit, como se muestra en la *Figura 1*.

Figura 1. Código en *MakeCode* para mostrar continuamente el valor medido de temperatura



O también generar un ícono cuando la *micro:bit* sea agitada, ver *Figura 2*.

Figura 2. Código en *MakeCode* para mostrar el ícono de una sombrilla cuando se agite la *micro:bit*



Estos son algunos ejemplos de lo que podrías realizar con los sensores de esta tarjeta. Sigue explorando otras posibilidades.

En muchos casos se requiere guardar la información de los sensores para realizar operaciones o acciones. Por ejemplo, si tomas una medición de ruido ahora y quieres hacer con ella dos cosas, como poner un signo de alerta si es alto y además mostrar el valor de la medición, tendrás que tomarla dos veces. Como resultado, el segundo valor que se tome podría ser diferente entre una medición y otra, y la señal de peligro podría no ser apropiada para ese nuevo valor. Por ello, es importante guardar el valor tomado para usarlo en varias operaciones.

Como has aprendido, para resolver este problema, en los programas se pueden crear espacios en la memoria del dispositivo denominados ***variables***, en los que se guarda información, dándole a cada espacio un nombre. Para examinar esto en más detalle, piensa que vas a guardar el dato del valor que tomó el sensor de ruido para poder ejecutar las dos acciones:

- Generar una alarma visual y sonora.
- Mostrar el valor medido del sonido que dispara la alarma.

Lo primero es crear una variable que se llamará nivel_sonido.

Figura 3. Representación de la variable nivel_sonido y el espacio para el almacenamiento de datos



En un algoritmo expresado en palabras, asignar un valor a esta variable podría escribirse de diferentes formas. Por ejemplo:

nivel_sonido ← *leer nivel sonido*
o
Asignar a nivel_sonido lectura nivel ruido

Y cuando se piense utilizar la variable, por ejemplo, para mostrar su valor se podría indicar:

Mostrar nivel_sonido

Figura 6. Bloques de instrucciones



Es hora de examinar cómo crear variables en *MakeCode*, si aún no lo has hecho.

1. Ingresa a *MakeCode* y crea un nuevo proyecto. Ya estando ahí, examina el menú de **variables,** ver *Figura 4*, y oprime el botón **Crear una variable** que aparece allí. Verás lo que aparece en la *Figura 5.*

Figura 4. Menú Variables en *MakeCode*

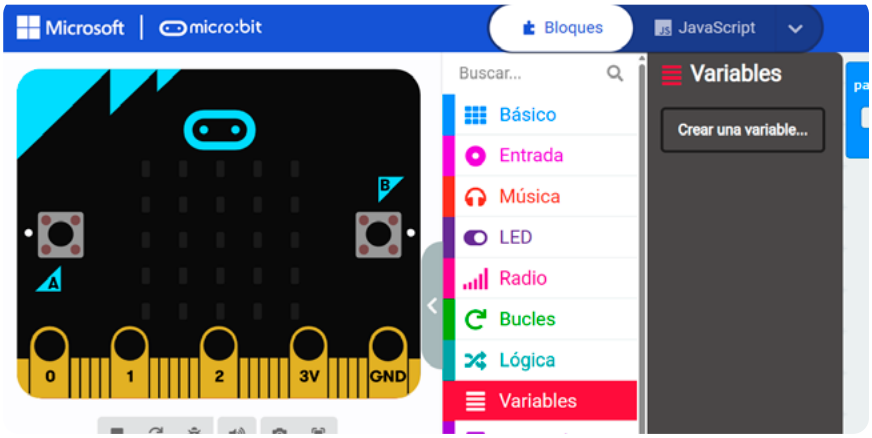
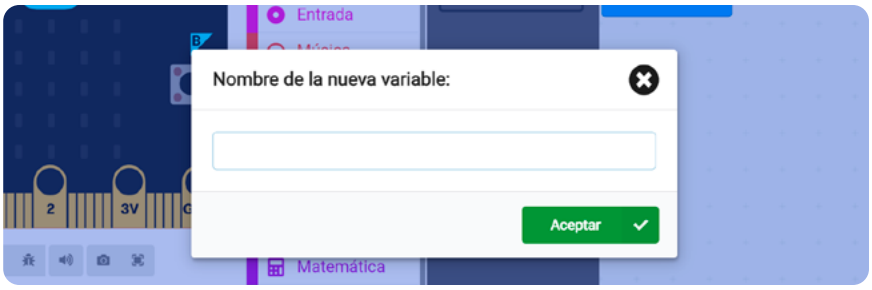


Figura 5. Pantallazo al dar clic en el botón Crear una variable



2. Digita nivel_sonido y dale aceptar. Observarás que en el menú Variables han aparecido 3 nuevos bloques de instrucciones, como se muestra en la *Figura 6.*
Estos bloques son:

- Fijar el valor de la variable a un valor específico, por defecto cero.
- Cambiar el valor de la variable adicionando el valor indicado, en este caso se suma 1.

- Y la variable misma, que podrás usar en diferentes contextos, para mostrar, hacer operaciones o comparar, por ejemplo.

Ya sabes lo necesario, es el momento de trabajar en *MakeCode* para realizar una actividad en parejas.

Ten en cuenta que en el código de la *Figura 7*, el bloque **nivel de sonido** corresponde a la lectura del sensor de ruido en un instante dado, mientras que el bloque nivel_sonido es la variable que se usa para almacenar el dato del valor medido.

Glosario

Variable: en computación es un espacio con un nombre donde se guardan uno o varios datos. Las variables en computación pueden ser de diferentes tipos (numérica, lógica, caracteres) y contener uno o muchísimos datos.

Dato: es una pieza de información, por ejemplo, un número.

Manos a la obra Conectadas

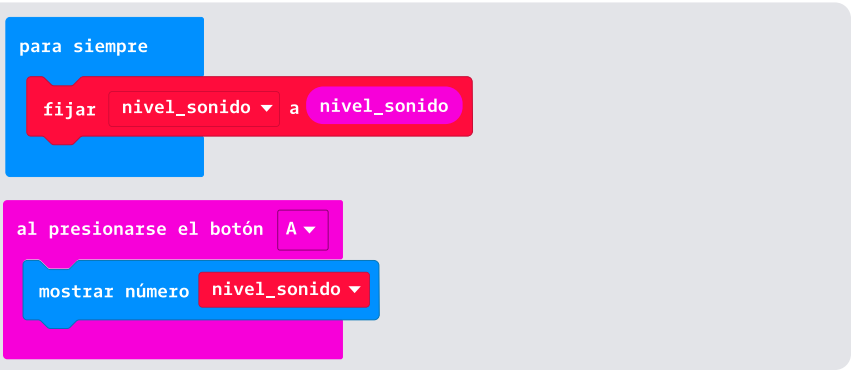
Esta sección corresponde al 80% de avance de la sesión



Organízate con una compañera o compañero, según te indique tu docente. Vayan resolviendo uno a uno los retos que se plantean a continuación.

Reto 1.
Como grupo, analicen el programa que aparece en la *Figura 7.*

Figura 7. Código usando la variable creada anteriormente



¿Qué hace? ¿Cómo lo saben?

Ahora codifiquen el programa y pruébenlo.

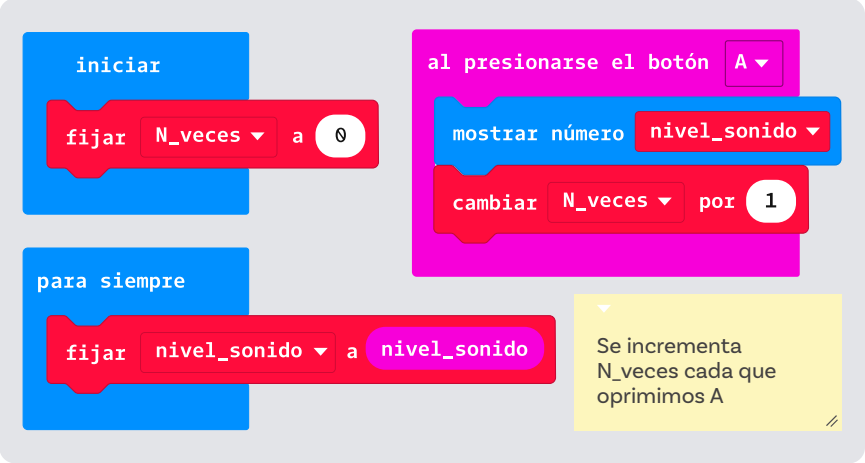
Reto 2

Un código como el de la *Figura 7* es la base para la creación de un control automatizado de niveles de ruido, como los que se utilizan en edificios inteligentes para detectar y reducir la contaminación auditiva.

Su reto ahora es modificar ese código para que no solo muestre el valor del nivel de sonido que está almacenado en la variable nivel_sonido, sino que, además, si se oprime el botón B, indique cuántas veces se ha presionado el botón A. Imaginen que eso equivale a crear un código que simule que un edificio inteligente lleva registro de las veces que se realizan las comprobaciones de niveles de ruido externo.

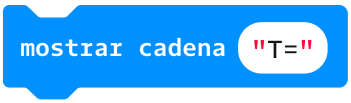
La *Figura 8* les dará una pequeña pista para que puedan modificar el código anterior y resolver este segundo reto.

Figura 8. Código de la figura 7 al que se agregó la variable N_veces



Nota

Puedes utilizar el bloque **Mostrar cadena** para indicar el tipo de dato que se mostrará en pantalla. Por ejemplo, puedes mostrar la letra T antes de indicar el valor de temperatura que se midió.



Como observan en la *Figura 8*, deben crear la variable N_veces. Lean el comentario que se agregó al código para explicar el uso del bloque **cambiar variable por 1**. Como la variable N_veces se va modificando cada vez que se presiona el botón A, se le puede considerar un **contador**.

Cuando su código muestre el número de veces que se oprimió el botón A, continúen con el siguiente reto.

Reto 3.

Imaginen que ahora van a programar otra característica del edificio inteligente, el control de temperatura. Para hacerlo deben diseñar un programa que tome datos de temperatura cada 30 segundos, los presente en pantalla y luego indique cuántos datos ha leído.

Este programa debe cumplir también con estos requerimientos:

- Al oprimir el botón A, se debe reiniciar en 0 el contador.
- Al presentar la información en la pantalla se debe indicar claramente cuándo se presenta la temperatura y cuándo se presenta el número total de datos leídos.

Finalicen los retos a tiempo para que tengan la oportunidad de revisar lo que han aprendido antes de terminar esta sesión de trabajo.

Glosario

Contador: este término se emplea para una variable numérica en la cual se va contando cuántas veces ha sucedido un hecho, evento o, en este caso, lectura de un valor.

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 definir y asignar valores a variables en un lenguaje de programación en bloques?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

2. ¿Puedes visualizar el contenido de las variables por medio de la programación de contadores?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

> **Si tus respuestas fueron "Parcialmente" o "Aún no", vuelve a las actividades propuestas en *MakeCode*. Luego, discute con otras compañeras y compañeros lo que se hizo al resolver cada reto. Si todavía te quedan dudas, consúltale a tu docente.**

Ahora te proponemos un reto con tu grupo usando una rutina llamada pensar, presentar e integrar (P-P-I):

- ◯ Primero, respondemos individualmente.
- ◯ Luego, cada persona en el grupo y en su turno le presenta al resto del equipo sus respuestas.
- ◯ Finalmente, el grupo integra una respuesta unificada.

Las preguntas que te proponemos:

- ◯ Si regresamos al reto, lo que hemos aprendido en esta sesión ¿para qué serviría?

- ◯ Contar cuántas veces se ha tomado una medida, ¿para qué podría servir? ¿En algún cálculo?

Actividad opcional para seguir consolidando aprendizajes

Aprovecha este espacio final para hacerte un esquema en el que resumas algo de lo que aprendiste, por ejemplo, colocando las definiciones de palabras y algún ejemplo de cómo se definen y usan variables. Piensa en conceptos como variable, dato, contador.

Crea un crucigrama con las definiciones de algunas palabras asociadas al uso de variables. No olvides incluir palabras como variable, dato y contador. También puedes agregar al crucigrama los nombres de algunos bloques de *MakeCode*. Luego, intercambia crucigramas con una compañera o compañero y rétense mutuamente a resolver sus propuestas.

Finaliza escribiendo una conclusión sobre lo que aprendiste en esta sesión y la forma en que te podría ayudar para resolver el reto de los invernaderos automatizados.

TIC

Sesión 3

Aprendizajes esperados

Con las actividades de esta sesión se espera que puedas avanzar en:

- Leer y escribir datos de dispositivos externos usando los pines.
- Cambiar la escala de las lecturas a una escala apropiada a la naturaleza de la variable.

Duración sugerida



40% 40% 20%



Material para la sesión

Material por grupo:

- Dispositivo con acceso a *MakeCode*.

Lo que sabemos, lo que debemos saber

Esta sección corresponde al 40% de avance de la sesión

Como probablemente sabes, todo circuito eléctrico requiere dos puntos de conexión para que funcione, con el fin de establecer un camino para la corriente eléctrica.



En las sesiones anteriores lograste leer datos de variables físicas como la temperatura y el nivel de sonido, y sabes cómo crear variables para guardar estos datos. También aprendiste a usar un contador para llevar un registro de cuántas mediciones has realizado.

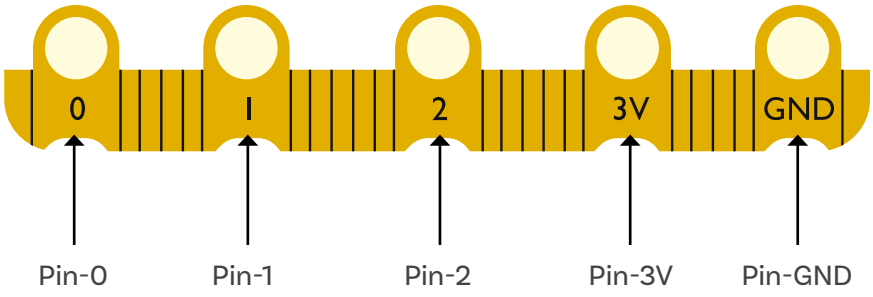
Además, descubriste cómo interpretar datos que la *micro:bit* obtiene en unidades para la temperatura o grados para la orientación, y cómo leer otras variables, como el nivel de sonido o de luz, en una escala del 0 al 255.

En esta sesión vas a explorar el uso los pines de la *micro:bit* y entender para qué sirven.

Igualmente, aprenderás a cambiar valores de 0 a 255 a una escala apropiada, por ejemplo, de 0 a 100.

La *micro:bit* cuenta con un conjunto de 25 pines o puntos de conexión eléctricos en la parte inferior, ver *Figura 1.*

Figura 1. Pines de la tarjeta *micro:bit*

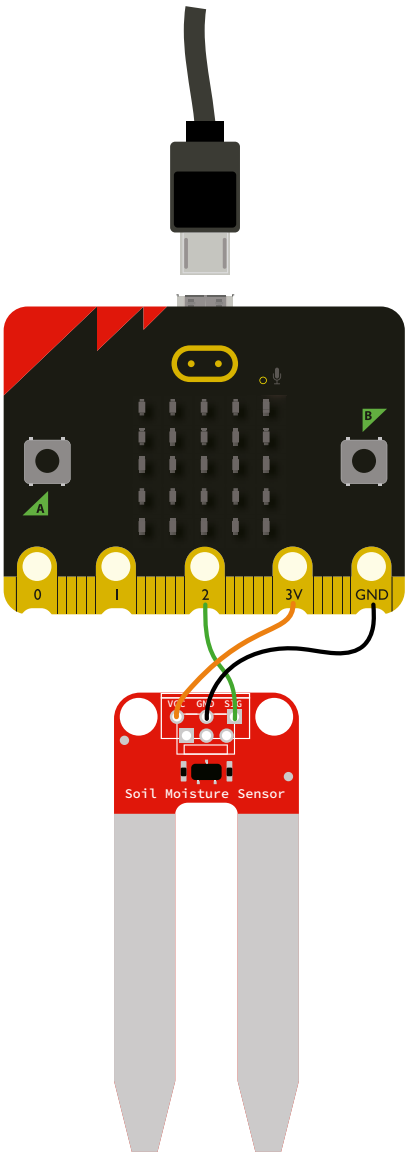


Hay 5 pines grandes que están conectados a los huecos de la tarjeta. Entre estos pines grandes, 3 que están numerados del 0 al 2, un pin etiquetado 3V (3 voltios) y un pin etiquetado GND, que permite conectar el circuito de retorno.

Nota

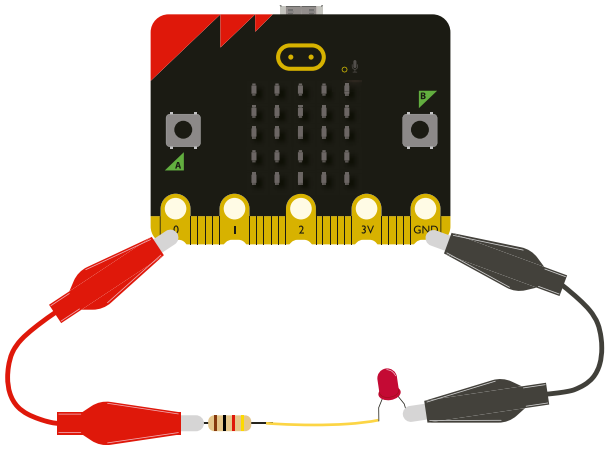
Mientras algunos sensores de la *micro:bit* trabajan con la escala 0 a 255, los pines trabajan con una escala que varía de 0 a 1023.

Figura 3. Conexión de un sensor de humedad a la *micro:bit*



Para conectar un elemento externo, como un nuevo sensor, este deberá conectarse entre un pin (0, 1 o 2) y el pin denominado GND. La *Figura 2* muestra cómo se conectaría un **LED** a los pines.

Figura 2. Conexión de un LED a la *micro:bit*



En este caso, el pin-0 no se usa como entrada, sino como **salida**. El circuito eléctrico se completa al colocar en el pin-0 un valor de tres voltios, que es el máximo voltaje que emite la *micro:bit*. Esto se logra escribiendo sobre dicho pin el número 1023.

La pequeña **resistencia** que se coloca en este circuito limita la corriente eléctrica para que no se dañe el pequeño LED rojo que se muestra en la *Figura 2*. Finalmente, se conecta a GND para completar el circuito.

Si se fuera a conectar un sensor, por ejemplo, de humedad, la conexión sería como se muestra en la *Figura 3*. Una ampliación de la conexión de este sensor se muestra en la *Figura 4*.

Figura 4. Detalle de la conexión del sensor de humedad a la *micro:bit*

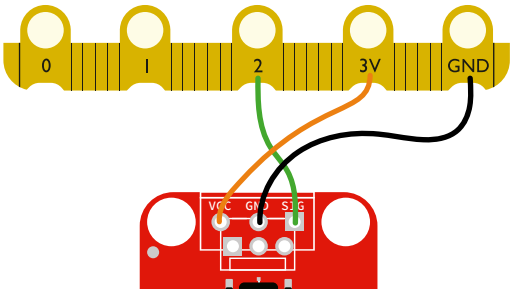
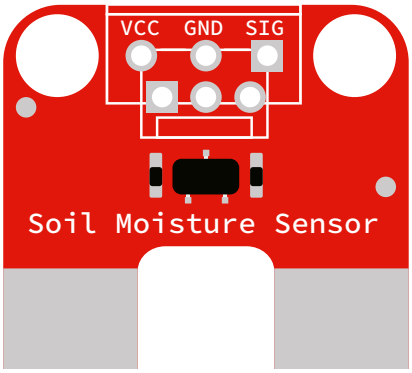


Figura 5. Puntos de conexión del sensor de humedad



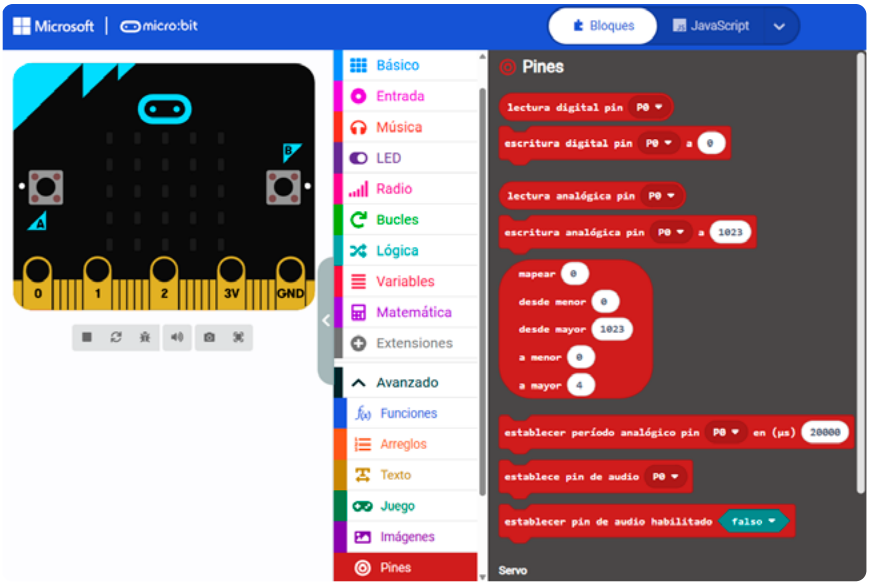
El sensor requiere tres puntos de conexión:

- **VCC:** donde se debe conectar a un **voltaje** de 3 voltios para que funcione.
- **GND:** que es tierra y representa en punto de cierre del circuito eléctrico que se establece.
- **SIG:** que es la abreviación de la palabra en inglés *"signal"* o señal en español, que es donde se podrá "leer" el valor de humedad medido de un suelo de un cultivo, por ejemplo.

Con este ejemplo del sensor de humedad, veamos ahora como usar el PIN-2 para "leer" el valor de humedad detectado.

En la *Figura 6* se muestra dónde se encuentran en *MakeCode* las instrucciones para usar los pines para "leer" un dato.

Figura 6. Bloques del menú de Pines en *MakeCode*



Nota

Si no cuentas con un sensor de humedad como el que se muestra aquí, puedes improvisar uno usando un par de puntillas y unos cables cocodrilo. Podrás encontrar instrucciones detalladas para hacerlo dando clic sobre el código QR o escaneándolo:

Tendrás que oprimir en el panel de herramientas la opción **avanzado** para que aparezca el menú de pines y, al hacerlo, verás los bloques de instrucciones de uso de pines.

Es hora de trabajar con *MakeCode* y conocer el funcionamiento de algunas de estas instrucciones.

Glosario

- **Pin:** punto de conexión eléctrica.
- **GND:** es la abreviación de la palabra en inglés "*ground*", que en español se traduce como tierra en un circuito eléctrico.
- **Voltio:** uno de los términos usados para describir la electricidad.
- **LED:** pequeño bombillo que se puede encender o apagar. La palabra LED viene de su nombre en inglés *(Light Emitting Diode)* que corresponde a diodo emisor de luz.
- **Salida:** corresponde a un medio por el cual un sistema de computación comunica información, por ejemplo, la pantalla de LED o la emisión de un sonido en la *micro:bit*.
- **Resistencia:** componente electrónico hecho de carbón u otros materiales que se opone al paso de la corriente eléctrica. Las franjas de colores de una resistencia indican su valor.
- **Voltaje:** se relaciona con la energía que hace que un circuito eléctrico funcione, se mide en voltios.

Nota

Para colocar comentarios en un código, basta ubicar el cursor sobre el lugar deseado en el espacio de programación de *MakeCode* y oprimir el botón derecho del ratón. Al hacerlo, aparecerá la opción **añadir comentario**.

Enlace

Recuerda que puedes acceder al editor *MakeCode* en:

Manos a la obra

Conectadas

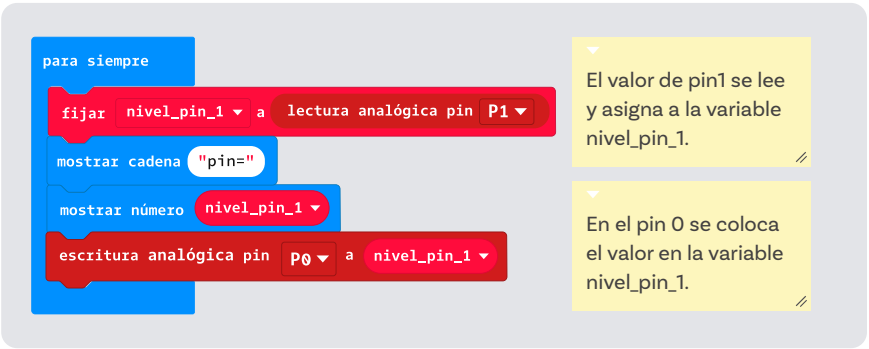
Esta sección corresponde al 80% de avance de la sesión

Por ahora no vas a trabajar con un sensor. Te ocuparás de hacer funcionar los pines como entradas, para leer valores en ellos, y como salidas, colocando valores que podrían permitir, por ejemplo, encender un LED, hacer funcionar un motor o hacer sonar una alarma.

Organícense en parejas, siguiendo las indicaciones de su docente. Accedan a *MakeCode*.

Empiecen por examinar el programa de la *Figura 7*.

Figura 7. Programa de lectura y escritura analógica de pines



¿Qué hace?

Los comentarios les ayudarán a inferir lo que hace este programa. Colocar comentarios en un programa es importante para que puedan recordar lo que han hecho y también para que otras personas puedan entender más fácilmente sus programas.

Ahora introduzcan en *MakeCode* el programa de la *Figura 7* y verifiquen su funcionamiento.

Para cambiar el valor en el pin 1 pueden usar el cursor sobre ese pin. En la *Figura 8* se ha colocado el valor 258 en el pin 1, pero como observan aún no aparece el mismo valor en el pin 0. Sin embargo,

Figura 10. Bloques de uso de los pines: lectura y escritura digital y analógica



Como quizás notaron, al ejecutar el código de la *Figura 7*, mientras la *micro:bit* está mostrando un dato o unos caracteres en pantalla, el programa no continúa con la siguiente instrucción. Por eso toma un tiempo en salir en el pin 0 el valor almacenado en la variable Nivel_pin_1.



al esperar un momento, se observa algo parecido a lo que muestra en la *Figura 9*. Esto sucede porque el código asigna o "escribe" en el pin 0 el valor que había tomado o "leído" del pin 1.

En este caso, el pin 0 es un pin de salida, porque se coloca o escribe un valor en él, y el pin 1 es un pin de entrada, porque se lee un valor de él.

Figura 8. Pines 0 y 1 de la *micro:bit* mostrando valores diferentes

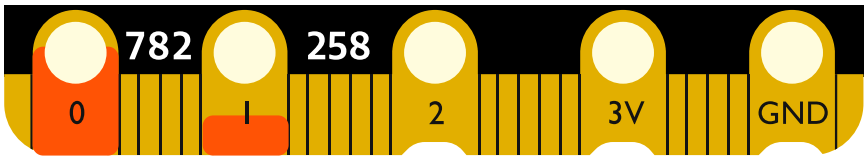
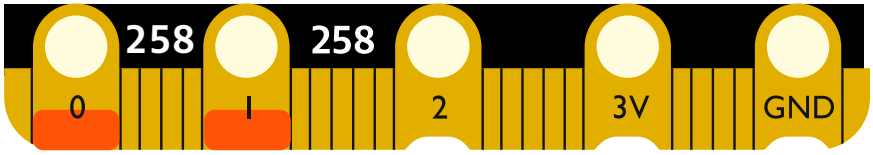


Figura 9. Pines 0 y 1 de la *micro:bit* mostrando el mismo valor



Los pines 0, 1 y 2 de la *micro:bit* son flexibles en su uso y tienen 4 formas de trabajar:

- Como entrada, leyendo un valor de 0 a 3 voltios, que se convierte en un número de 0 a 1023.
- Como salida, escribiendo un número entre 0 y 1023, que se convierte en un voltaje de 0 a 3 voltios.
- Como entrada digital, que solo puede tener 2 valores: 0 o 1, que corresponden a 0 voltios o 3 voltios.
- Como salida digital, que solo puede tener 2 valores: 0 o 1, que coloca a la salida 0 voltios o 3 voltios.

Cada pin se configura automáticamente en función de la instrucción o bloque que se utilice. Si se usa una instrucción de entrada, ver *Figura 10*, quedará configurado como entrada.

Ahora, trabajen como equipo en la resolución del primer reto.

Reto 1.
Lean la situación a solucionar.

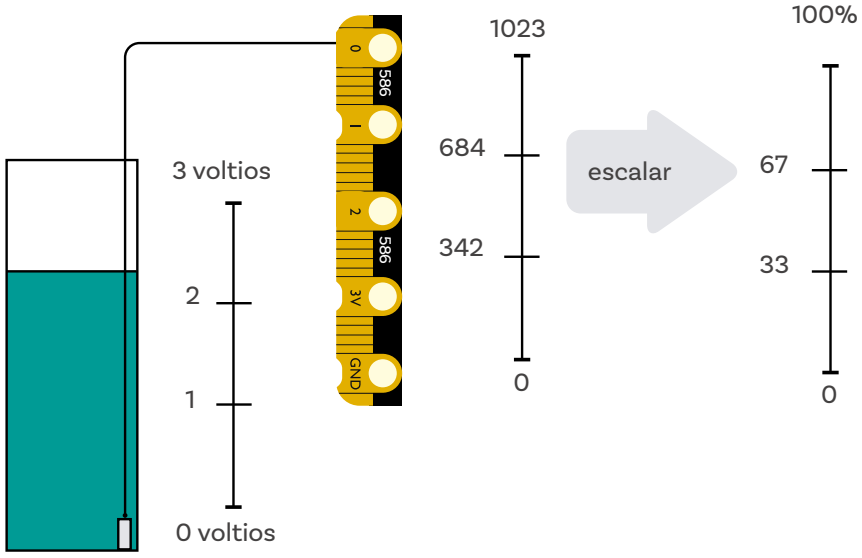
En tu escuela se ha instalado un tanque que recolecta el agua lluvia para aprovecharla en los baños y lavamanos. Para poder monitorear la cantidad de agua recolectada, se ha instalado un sensor de nivel de agua en el fondo del tanque. Este sensor se ha conectado al pin 0 de una micro:bit.

Si el sensor entrega 0 voltios, esto indica que el tanque está vacío, y si entrega 3 voltios, el tanque está lleno. Si, por ejemplo, el tanque se encuentra lleno solo hasta la mitad, el sensor colocará 1,5 voltios en el pin 0.

Sin embargo, estos valores en voltios no son claros para todas las personas. Por eso se requiere crear un programa que muestre el porcentaje de llenado del tanque en una escala de 0 a 100%. Su misión es crear este programa.

Antes de empezar, miren el diagrama que se presenta en la *Figura 11*.

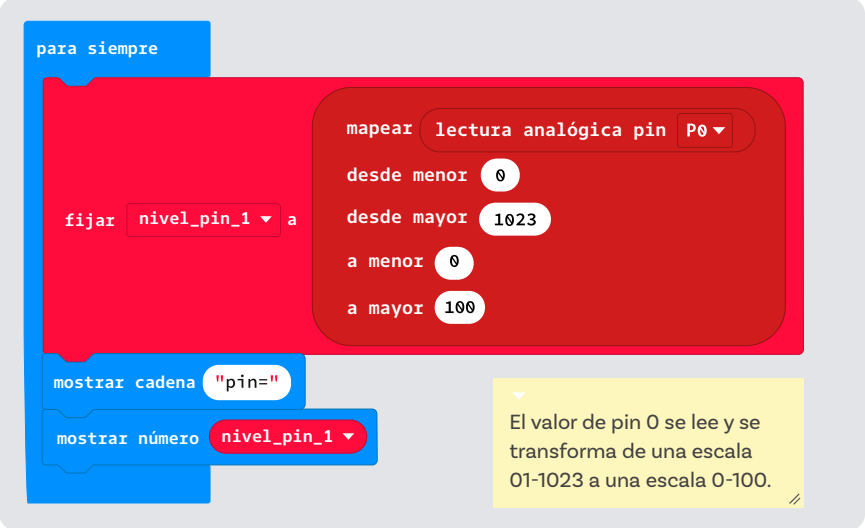
Figura 11. Diagrama Reto 1



Como se observa en la *Figura 11*, el sensor que se coloca al fondo del tanque mide el agua que hay encima de él y da una señal que varía de los 0 voltios a los 3 voltios. Al conectar este sensor al pin 0 de la *micro:bit*, se lee un número entre 0 y 1023, que en este caso es el 586. Este valor indica que el tanque está algo más de medio lleno. Así que el reto es convertir ese valor en algo que se pueda mostrar en pantalla y que vaya en porcentajes de 0 a 100, que es una escala más usual.

Examinen ahora el programa de la *Figura 12* y utilicen el comentario agregado para entender su funcionamiento.

Figura 12. Código con bloque del menú pines llamado map



¿Qué hace?

Ahora prueben este programa y jueguen con el valor de entrada al pin 0, esperando cada vez un momento hasta que se muestre el porcentaje en la pantalla.

¿Podrían hacer que salga el valor seguido del símbolo %?

Si ya lograron lo anterior, resuelvan este segundo reto.

Reto 2

Lean la situación problema.

El código anterior se utilizó también para programar el tanque de recolección de aguas lluvias que se utiliza para regar las plantas en un invernadero. Sin embargo, se debe ajustar el programa para que avise cuando quede menos del 25% de agua en el tanque, a fin de llenarlo con agua de un pozo.

Su misión es hacer un programa que muestre en la pantalla un ícono que indique que el tanque tiene suficiente agua y otro ícono que indique que es hora de buscar otra fuente de agua, dado que el agua lluvia no es suficiente. Utilicen su creatividad para diseñar estos dos íconos.

Recuerden terminar con tiempo para que logren reflexionar y organizar las ideas sobre lo que aprendieron resolviendo los retos.

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 leer y escribir datos de dispositivos externos usando los pines?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

2. ¿Puedes cambiar de escala de las lecturas a una escala apropiada a la naturaleza de la variable?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

Si tus respuestas fueron "Parcialmente" o "Aún no", vuelve a las actividades propuestas en *MakeCode*. Luego, discute con algunas compañeras y compañeros de tu clase lo que hace el código de *MakeCode* que se presenta en las diferentes figuras. Si todavía te quedan dudas, consúltale a tu docente.

Actividad opcional para seguir consolidando aprendizajes

Ahora te proponemos un reto con tu grupo usando una rutina llamada pensar, presentar e integrar (P-P-I):

- Primero, respondemos individualmente.
- Luego, cada persona en el grupo y en su turno le presenta al resto del equipo sus respuestas.
- Finalmente, el grupo integra una respuesta unificada.

Las preguntas que te proponemos, son:



Si regresas al reto del invernadero automatizado, ¿cómo te sirve lo que aprendiste en esta sesión?

Si en lugar de querer mostrar un ícono que indique que se debe llenar el tanque, lo que se quiere es encender una bomba que envíe agua al tanque, ¿cómo te imaginas que podría hacerse esto? Haz un dibujo con tu idea.

Aprovecha este espacio final para hacerte un esquema en el que resumas algo de lo que aprendiste, indicando para qué sirven los pines y proponiendo un ejemplo de su uso y del ajuste de la escala de una señal.

Grado 6° **Guía 3**

TIC

Sesión 4

Aprendizajes esperados

Con las actividades de esta sesión se espera que puedas avanzar en:

- Realizar operaciones matemáticas de mayor complejidad (ecuaciones) entre variables.
- Definir y asignar valores a variables de ecuaciones matemáticas.

Duración sugerida



15% 70% 15%

Material para la sesión

Material por grupo:

- ○ Dispositivo con acceso a MakeCode.
- ○ Anexo 1.1.



Anexo

Anexo 1.1



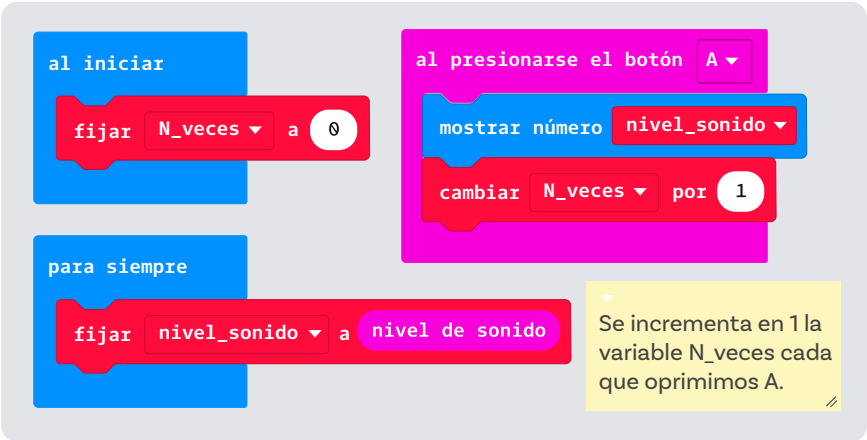
Lo que sabemos, lo que debemos saber

Esta sección corresponde al 15% de avance de la sesión

Después de trabajar las sesiones anteriores ya tienes varios aprendizajes relacionados con variables, sensores y pines, pero ¿ya sabes todo lo necesario para resolver el reto planteado al comienzo de la guía? Te sugerimos revisar el reto en el *Anexo 1.1* e identificar lo que falta por aprender. Tómate unos minutos y luego continúa.

Notarás que se requiere realizar operaciones con los valores que se guardan en las variables. Ya en sesiones pasadas has trabajado con algunas operaciones básicas:

Figura 1. Código que aumenta el valor de la variable N_veces cada vez que se oprime el botón A



En el programa de la *Figura 1* sumaste 1 al valor almacenado en la variable N_veces.

En el programa de la *Figura 2*, por otro lado, realizaste una operación de escalamiento para cambiar un valor que viene entre 0 y 1023 a una escala de 0 a 100(%).

Figura 3. Bloque map y cambio de escala de 0 a 1023 por 0 al 4

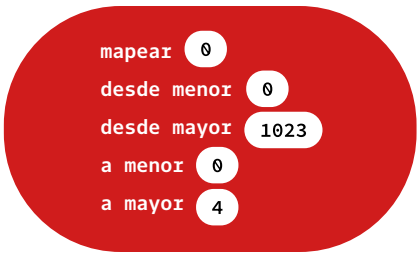
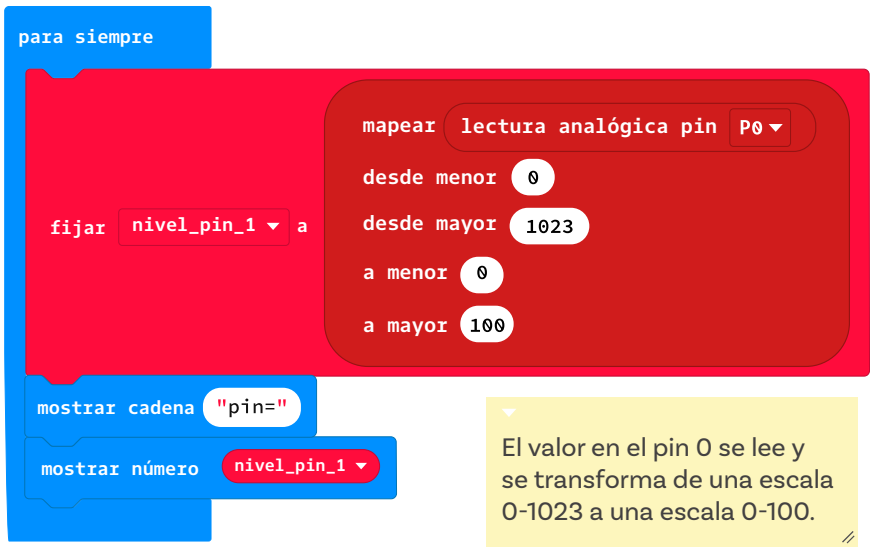


Figura 2. Código para cambiar valores de una escala analógica a porcentajes



Enlace

Recuerda que puedes acceder al editor *MakeCode* en:

Manos a la obra Conectadas

Esta sección corresponde al 85% de avance de la sesión

Organízate en parejas, siguiendo las indicaciones de tu docente.

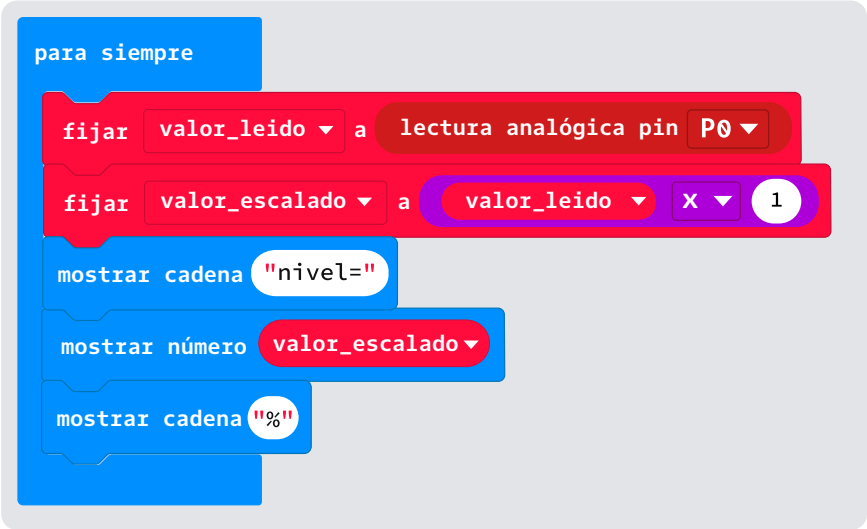
Van a trabajar nuevamente utilizando *MakeCode*.

Observen la *Figura 3*. Este bloque o instrucción de escalamiento es equivalente a la siguiente expresión matemática:

$$Valor_{obtenido} = \frac{(valor_{entrada} - from_low) * (to_high - to_low) + to_low}{(from_high - from_low)}$$

Examinen ahora el programa que se presenta en la *Figura 4*.

Figura 4. Código que incorpora variables para almacenar un valor leído del pin 0 y otro para almacenar el valor al que corresponde en otra escala



Piensen en la siguiente pregunta:

¿Qué mostrará en pantalla ese programa?

Como notarán, lo único que se hizo fue multiplicar por 1 el valor leído en el pin 0, donde se supone que hay un sensor de nivel.

Reto 1

El primer reto que tienen es realizar el escalamiento sin utilizar la instrucción **map**, que aprendieron en la sesión pasada y que se muestra en la *Figura 3*.

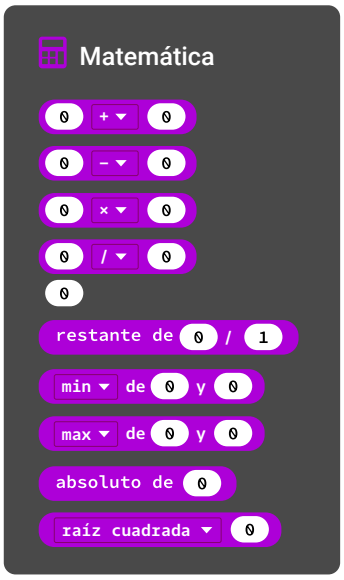
Este escalamiento es una relación de proporcionalidad, que a veces se conoce como regla de 3[1].

$$Valor_{escalado} = \frac{valor_{leido} * 100}{1023}$$

Modifiquen el programa para que haga este cálculo. Encontrarán las operaciones matemáticas en el menú **Matemática**, ver *Figura 5*.

1. La regla de tres que se presenta solo es válida si ambas escalas comienzan en cero.

Figura 5. Menú de bloques de matemáticas en *MakeCode*



Comparen el resultado que obtuvieron con el obtenido mediante el programa creado en la sesión anterior.

¡Es hora de abordar un nuevo reto!

Reto 2

Van a simular ahora, con ayuda de la *micro:bit*, una de las funciones de un invernadero inteligente, la toma periódica de niveles de temperatura. Por esta razón, deberán crear un programa que mida la temperatura cada 10 segundos y que repita esta acción 10 veces, es decir, que haga un proceso de toma de datos de 100 segundos.

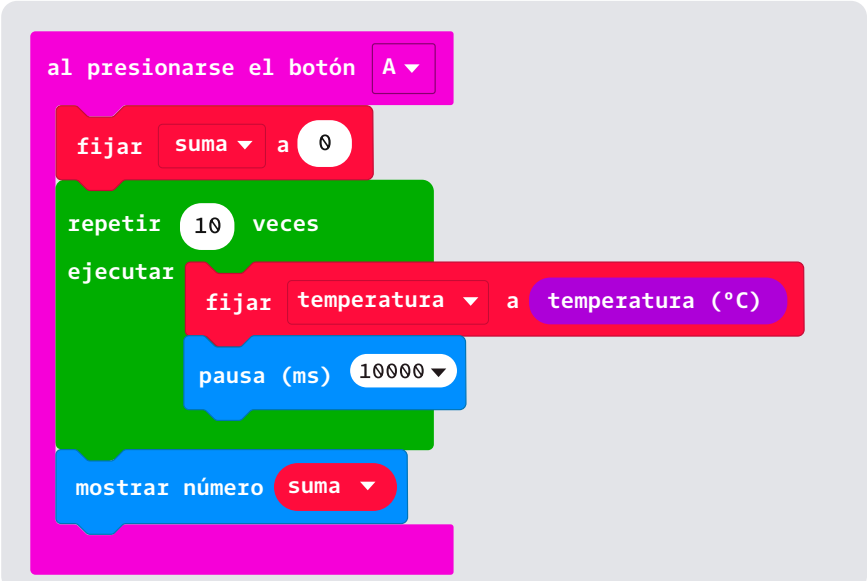
Para realizar este programa, deberán sumar cada temperatura medida a una variable llamada suma.

Al final de las 10 medidas, deberán dividir lo que hay en la variable suma entre 10 y mostrar el resultado.

¿Tienen idea de lo que se está calculando acá?

La *Figura 6* muestra una parte del código que deberán utilizar.

Figura 6. Código para tomar 10 valores de temperatura y almacenarlos en la variable temperatura



Si prueban este programa, después de pasados 100 segundos, notarán que muestra 0, dado que la variable suma se inicia en cero y no está acumulando ningún valor.

Su reto es completar este programa para que, al final, muestre la media aritmética o promedio de los valores leídos.

¡Éxitos en este trabajo!

Una vez terminen, reserven tiempo para realizar las actividades de cierre de la sesión.

Glosario

Promedio: también conocido como media aritmética, es un valor único que representa el resultado de sumar todos los valores de un conjunto de datos y dividirlos por la cantidad de valores que se promedian.

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 realizar operaciones matemáticas de mayor complejidad (ecuaciones) entre variables?

- ◯ Sí
- ◯ Parcialmente
- ◯ Aún no

2 ¿Puedes definir y asignar valores a variables de ecuaciones matemáticas?

- ◯ Sí
- ◯ Parcialmente
- ◯ Aún no

> **Si tus respuestas fueron "Parcialmente" o "Aún no", vuelve a las actividades propuestas en *MakeCode*. Luego, discute con tu compañera o compañero de grupo lo que hicieron durante la actividad anterior, prestando especial atención a las operaciones matemáticas realizadas con el código. Si todavía te quedan dudas, consúltale a tu docente.**

Finaliza haciendo una gráfica o esquema que te permita recordar lo aprendido sobre pines, variables y operaciones matemáticas en *MakeCode*. Cuando tu docente lo indique, comparte tu gráfica con otras personas y escúchales presentar las suyas. Si lo crees pertinente, agrega detalles finales a tu gráfica después de este espacio de socialización.

Sesión 5

Aprendizajes esperados

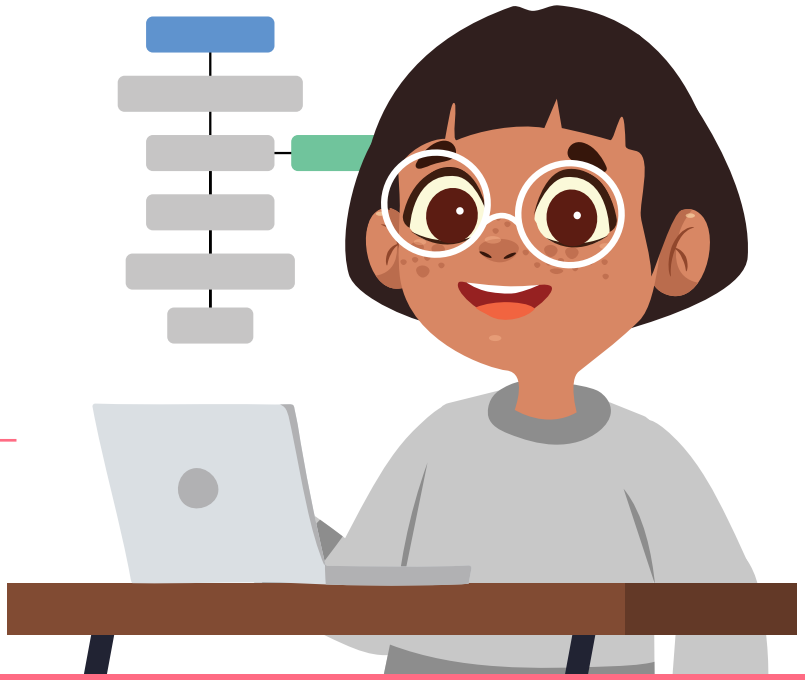
Con las actividades de esta sesión se espera que puedas avanzar en:

- Definir, asignar valores y realizar operaciones matemáticas con variables.
- Hacer lecturas de los sensores de la *micro:bit* y cambiar las escalas de las lecturas según sea necesario.
- Mostrar información en la pantalla de LED y gestionar valores a través de los pines.

Duración sugerida



15% 70% 15%



Material para la sesión

Material por grupo:

- ○ Dispositivo con acceso a *MakeCode*.
- ○ Anexo 1.1.

Anexo

Anexo 1.1



Actualmente muchas plantas se cultivan en invernaderos especialmente acondicionados para asegurar condiciones controladas de temperatura, humedad y algunas veces de radiación solar.

Para mantener las condiciones bajo control es necesario medir estas variables físicas (humedad, temperatura y radiación solar), en algunos casos simplemente para indicar que existe un problema para que las personas a cargo tomen acciones, o en casos más automatizados, para que el sistema corrija automáticamente el problema.

Por ejemplo, si la humedad es muy baja, será necesario regar el cultivo.

El reto que se te propone es diseñar un programa prototipo de invernadero automatizado, que funcione con la micro:bit para que se pueda probar en simulación usando MakeCode. Este prototipo debería cumplir con las siguientes especificaciones:

- Tomar datos de temperatura, humedad y nivel de luz cada hora (tengan en cuenta que cada hora se simulará como un minuto).
- Al oprimir el botón B su programa debe comenzar a tomar datos.
- Al oprimir el botón A debe presentar los datos máximos, mínimos y el promedio de las variables hasta el momento.
- Si la humedad es inferior a un *valor* definido debe hacer sonar una alarma para avisar que es necesario regar el cultivo. Este valor debe poderse ajustar por programa, dado que puede variar de cultivo a cultivo.

Lo que sabemos, lo que debemos saber

Esta sección corresponde al 15% de avance de la sesión

Ahora puedes crear y trabajar con variables, leer información de los sensores de la *micro:bit*, realizar operaciones matemáticas entre datos, cambiar escalas para interpretar mejor las lecturas, mostrar resultados en la pantalla de LED y hasta interactuar con los pines para enviar o recibir información.

Es hora de leer de nuevo el reto que se encuentra en el *Anexo 1.1*.

Manos a la obra Desconectadas

Esta sección corresponde al 85% de avance de la sesión

Organízate en grupos de 2 a 3 personas, siguiendo las indicaciones de tu docente. Trabajarán como equipo con el fin de resolver el reto planteado.

Además de calcular el promedio, lo cual ya saben hacer, deberán encontrar los valores máximos y mínimos de las variables leídas. Revisen el siguiente algoritmo en diagrama de flujo y pruébenlo a lápiz y papel con los datos que se proporcionan luego.



Figura 1. Diagrama de flujo del algoritmo para encontrar valores máximos de temperatura

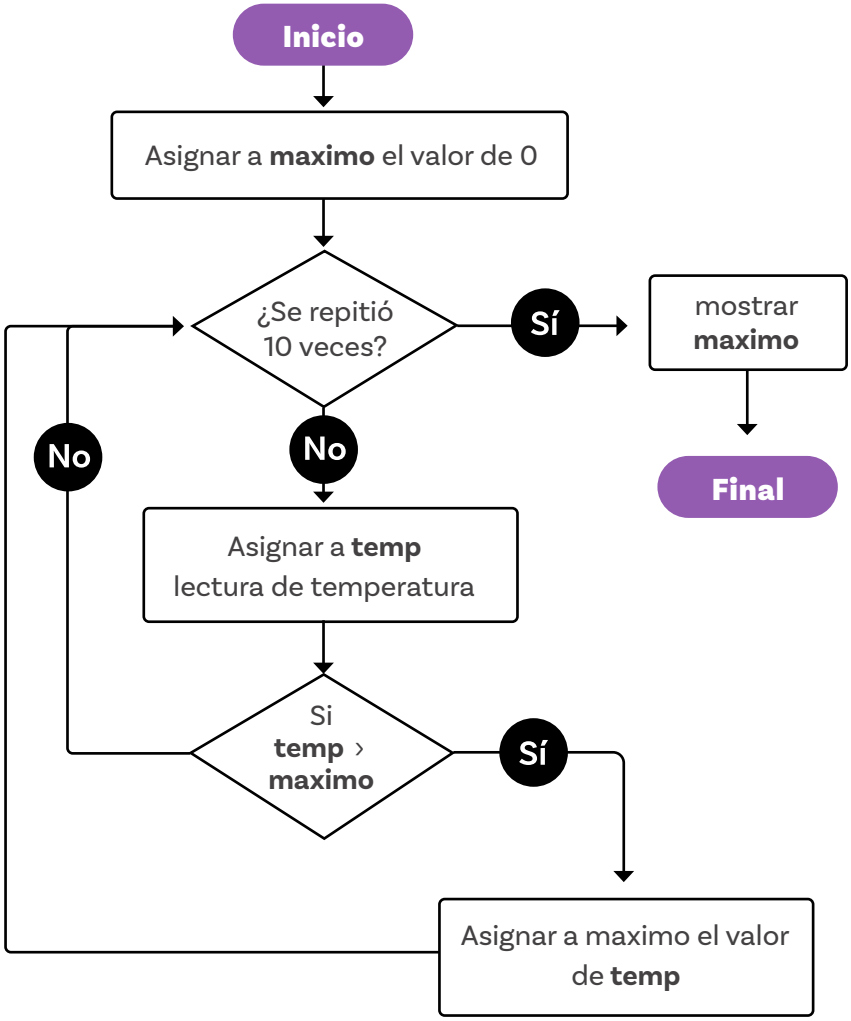


Figura 2. Datos de un caso de prueba obtenidos al lanzar un dado

2	5	3	1	4	5	3	6	6	1

Si todo funcionó bien, al seguir el diagrama de flujo de la *Figura 1* deberían haber obtenido 6 como valor máximo, pues este es el valor más grande de entre los números del caso de prueba.

¿Qué deberían cambiar en ese diagrama de flujo para que les ayude a identificar el valor mínimo?

Nota

El reto les pide incluir una alarma sonora. Para generarla pueden utilizar este bloque, que encontrarán en el menú de Música.



Anexo

Anexo 1.1



Escriban su algoritmo y pruébenlo a mano con el mismo conjunto de datos que se utiliza en la *Figura 2*.

Es hora de codificar en *MakeCode*.

Manos a la obra

Conectadas

Ya saben todo lo que requieren para resolver el reto planteado en el *Anexo 1.1*.

Como grupo diseñen su programa y pruébenlo en *MakeCode*.

Es probable que su código tenga errores, deberán encontrarlos y corregirlos. Este proceso en programación se denomina depuración.

No duden en examinar qué hacen los demás grupos si se encuentran sin saber cómo continuar.

¡Éxitos!

Computación y sociedad

En esta guía pudiste ver cómo la programación ayuda en la producción de alimentos.

La computación se utiliza cada vez más en el sector agrícola, por ejemplo, para dosificar los agroquímicos en la cantidad justa y necesaria para evitar que contaminen.

Existe un creciente desarrollo de tecnologías para lo que se ha llamado agricultura de precisión, en la cual se usan drones y robots que permiten tener zonas agrícolas más sostenibles.

Las personas que producen y usan estas tecnologías deben saber de computación.

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 definir, asignar valores y realizar operaciones matemáticas con variables?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

2. ¿Puedes hacer lecturas de los sensores de la *micro:bit* y cambiar las escalas de las lecturas según sea necesario?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

3. ¿Puedes mostrar información en la pantalla de LED y gestionar valores a través de los pines?

   - ◯ Sí
   - ◯ Parcialmente
   - ◯ Aún no

Si tus respuestas fueron "Parcialmente" o "Aún no", vuelve a las actividades propuestas. Luego, discute con tus compañeras y compañeros de grupo lo que se hicieron al ir resolviendo el reto. Si todavía te quedan dudas, consúltale a tu docente.

Ahora te proponemos un reto con tu grupo usando una rutina llamada pensar, presentar e integrar (P-P-I):

- Primero, respondemos individualmente.
- Luego, cada persona en el grupo y en su turno le presenta al resto del equipo sus respuestas.
- Finalmente, el grupo integra una respuesta unificada.

Las preguntas que te proponemos:

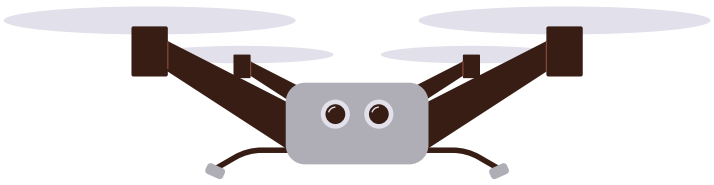
¿?

¿Qué resultó más difícil?

¿Qué más fácil?

Si el reto pidiera abrir un grifo en lugar de hacer sonar una alarma, ¿qué deberías cambiar?





Anexo 1.1 Reto: Invernadero automatizado

Actualmente, muchas plantas se cultivan en invernaderos especialmente acondicionados para asegurar condiciones controladas de temperatura, humedad y, algunas veces, de radiación solar.

Para mantener las condiciones bajo control es necesario medir estas variables físicas (humedad, temperatura y radiación solar), en algunos casos simplemente para indicar que existe un problema y que las personas a cargo tomen acciones, o en casos más automatizados, para que el sistema corrija automáticamente el problema.

Por ejemplo, si la humedad es muy baja, será necesario regar el cultivo.

El reto que se te propone es diseñar un programa prototipo de invernadero automatizado, que funcione con la *micro:bit* para que se pueda probar en simulación usando *MakeCode*. Este prototipo debería cumplir con las siguientes especificaciones:

- Tomar datos de temperatura, humedad y nivel de luz cada hora (tengan en cuenta que cada hora se simulará como un minuto).
- Al oprimir el botón B su programa debe comenzar a tomar datos cada minuto.
- Al oprimir el botón A debe presentar los datos máximos, mínimos y el promedio de las variables hasta el momento.
- Si la humedad es inferior a un valor definido debe hacer sonar una alarma para avisar que es necesario regar el cultivo. Este valor debe poderse ajustar por programa, dado que puede variar de cultivo a cultivo.



TIC

Apoya:

Educación

BRITISH COUNCIL

Colombia Programa

{EL CÓDIGO A TU FUTURO}