

# Curso F

El último curso de Fundamentos de Ciencias de la Computación se ha adaptado a las necesidades de los/as estudiantes de sexto básico.

En estas lecciones, los/as estudiantes crearán programas con diferentes tipos de loops, eventos, funciones y condicionales. También investigarán diferentes técnicas de resolución de problemas y discutirán los impactos sociales de la computación y de Internet. Al final del currículo, los/as estudiantes crearán historias interactivas y juegos que puedan compartir con sus amigos y familiares.

Enlaces para los/as profesores: [Teacher Videos Playlist](#)

## Lección 1: Mis Amigos Robóticos Loopy

Algorithms | Debugging | Unplugged

¡Convierte a tus amigos en robots y diles qué hacer usando loops!

## Lección 2: Secuencias en Laberinto

Sequencing | Debugging | Loop

¡En esta lección, aprenderás a escribir tus propios programas!

## Lección 3: Construyendo los Cimientos

Unplugged | Persistence | Frustration

Construye una estructura que pueda sostener un libro. Puede que te sientas frustrado/a, ¡recuerda ser perseverante!

## Lección 4: Depuración con Scrat

Bug | Debugging | Scrat | Ice Age

Encuentra problemas en los desafíos de La Era del Hielo y practica tus habilidades de debugging.

## Lección 5: Programación en Artista

Artist | Programming | Revise | Private Information

Crear imágenes hermosas programando al Artista.

## Lección 6: Mis Amigos Robóticos Loopy

Unplugged | Loop | Repeat

¡Convierte a tus amigos en robots y diles qué hacer usando loops!

## Lección 7: Loops en Artista

Loop | Artist

En esta lección, ¡los loops facilitan el hacer imágenes incluso más geniales con artista!

## Lección 8: Loops Anidados

Nested Loops | Loops | Bee | Maze

Loops dentro de loops dentro de loops. ¿Qué significa? Esta lección te enseñará qué sucede cuando colocas un loop dentro de otro loop.

## Lección 9: Loops Anidados con Frozen

Loop | Nested Loop | Artist

Anna y Elsa tienen excelentes habilidades de patinaje sobre hielo, pero necesitan tu ayuda para crear patrones en el hielo. Utiliza loops anidados para crear algo súper COOL.

## Lección 10: Algoritmos: Tangramas

Unplugged | Algorithms

Aquí aprenderás acerca de algoritmos usando desafíos llamados tangramas.

## Lección 11: Introducción a los Desafíos Online

Algorithms | Loop | Repeat

Esta lección te dará la práctica en las habilidades que necesitarás para este curso.

## Lección 12: Ciudadanía Digital

Common Sense Education | Cyberbullying

El Bullying nunca está bien. Esta lección te enseñará lo que está bien decir en línea y lo que NO.

## Lección 13: Eventos en La Era de Hielo

Event | Ice Age

Piensa en tu videojuego favorito. ¿Te has preguntado alguna vez cómo fue hecho? ¡En estos desafíos desarrollarás tu propio videojuego con amigos de la Era del Hielo!

## Lección 14: Condicionales en Minecraft

Conditional | Minecraft

¡Esquiva la lava! Aquí aprenderás acerca de los condicionales en el mundo de Minecraft.

## Lección 15: Variables: Variables en Sobres

Unplugged | Variable

Los sobres y las variables tienen algo en común: ambos pueden contener cosas valiosas. Aquí aprenderás lo que son las variables y las cosas geniales que éstas pueden hacer.

## Lección 16: Variables en Artista

Variable | Artist

¡No te olvides de traer la creatividad a la clase! En estos desafíos harás dibujos fantásticos usando variables.

## Lección 17: Variables en Play Lab

Variable | Play Lab

Pronto aprenderás a hacer que los personajes interactúen en un juego usando variables.

## Lección 18: Loops Para: Diversión con Loops Para

Unplugged | For Loops

¡Vas a entretenerte muchísimo aprendiendo sobre los Loops "Para"!

## Lección 19: Loops Para en Abeja

For Loop | Bee

Bsss...bssss... En estos desafíos guiarás a una abeja hacia el néctar y la miel usando Loops "Para".

## Lección 20: Loops Para en Artista

For Loop | Artist

Prepárate para hacer tu próxima obra maestra. Aquí usarás los loops "para" para hacer dibujos que te dejarán con la boca abierta.

## Lección 21: Funciones: Componer Canciones con Parámetros

Unplugged | Function | Parameter

¡Podrías lanzar el próximo gran éxito! En esta lección, aprenderás qué son los parámetros y cómo hacen fantásticas canciones.

## Lección 22: Funciones en Abeja

Function | Bee

¡La abeja necesita tu ayuda otra vez! Aquí usarás funciones para conseguir néctar y hacer miel.

## Lección 23: Funciones con Parámetros en Artista

Function | Parameter | Artist

Prepara tus dedos programadores. En estos desafíos harás dibujos impresionantes en Artista usando funciones con parámetros.

## Lección 24: Funciones con Parámetros en Abeja

Function | Parameter | Bee

Ya has podido practicar un poco usando funciones con parámetros. ¡Esta lección continuará tu práctica con Abeja!

## Lección 25: Explorar Ideas de Proyectos

Project | Define | Prepare | Try | Revise | Reflect

¡Es hora de inspirarse! Estos desafíos te mostrarán muchos juegos e ilustraciones pre armadas que te ayudarán a desarrollar tu plan para el GRAN proyecto.

## Lección 26: El Proceso de Diseño

Project

Los proyectos así de grandes toman tiempo y mucha planificación. Aquí, aprenderás acerca del proceso de diseño que usarás para construir tu propia creación.

## Lección 27: Construye Tu Proyecto

Project

¡Finalmente, puedes empezar a construir tu proyecto!

## Lección 28: Revisa Tu Proyecto

Project

Roma no se construyó en un día y tu proyecto tampoco debería. Tómate el tiempo para editar y revisar tu proyecto para que sea lo mejor que pueda ser.

## Lección 29: Presenta Tu Proyecto

Project

¡Es hora de mostrar tu trabajo! Aquí presentarás tu fantástico proyecto a tus pares.



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 1: Mis Amigos Robóticos Loopy

Algoritmos | Depurar | Sin conexión

## Descripción

Basándose en la actividad inicial "Mis amigos Robóticos", los/as estudiantes aprenden a usar loops cuando programan sus robots para construir estructuras más grandes de manera más eficiente.

## Propósito

Esta lección sirve como una re introducción a los loops, usando el ahora familiar set de instrucciones de programación para el/a "robot". Los/as estudiantes desarrollarán habilidades de pensamiento crítico buscando patrones de repetición en los movimientos de los/as compañeros/as y determinando cómo simplificar esos patrones repetidos usando loops.

## Agenda

### Actividad previa (5 min)

Revisión de Mis Amigos Robóticos

### Actividad Principal (45 min)

Mis Amigos Robóticos

### Actividad de Cierre (10 min)

Registrar en un diario

### Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Comprender la necesidad de precisión en la programación.
- Aprender a reconocer un error y depurar el código que funciona mal.

## Preparación

- ▣ Vea **Mis Amigos Robóticos Loopy - Video para el docente**.
- ▣ (Opcional) Imprima un **Programación: Mis Amigos Robóticos - Clave de símbolo** para cada grupo o cada 4 estudiantes. Alternativamente, encuentre un ligar donde pueda desplegar esta información, que puede ser usado como referencia para los/as estudiantes durante la lección.
- ▣ Prepare una pila de 20 vasos de papel por cada grupo de 4 estudiantes. O
- ▣ (Opcional) imprima y corte un **resource paper-trapezoid-template not found** para cada grupo si su clase no va a usar los vasos.
- ▣ Imprima un set de **Apilando Ideas - Manipuladores** por grupo.
- ▣ Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Programación: Mis Amigos Robóticos - Guía de preparación para el docente**

Para los estudiantes

- **Programación: Mis Amigos Robóticos - Video sin conexión (descargar)**
- **Programación: Mis Amigos Robóticos - Clave de símbolo**

- **Apilando Ideas** - Manipuladores
- **Diario Think Spot** - Diario de reflexiones

Hacer una copia ▾

## Vocabulario

- **Algoritmo** - una lista de tareas para finalizar una tarea.
- **Bug o error** - parte de un programa que no funciona correctamente.
- **Depurar** - encontrar y solucionar problemas en un algoritmo o programa.
- **Programa** - un algoritmo que se ha programado como algo que puede ser ejecutado por una máquina.

# Guía docente

## Actividad previa (5 min)

### Revisión de Mis Amigos Robóticos

#### Programación: Mis Amigos Robóticos - Guía de preparación para el docente

**Objetivo:** Esta revisión refrescará el conocimiento de los alumnos acerca de cuán rápido los programas para la actividad de "Mis Amigos Robóticos" se pueden volver intensos.

**Mostrar:** Muestre el [symbol-key][1] que usamos en Mis Amigos Robóticos. Para cada uno de los seis símbolos, pida a los alumnos que muestren cómo luce para un robot seguir esa instrucción.

**Modelar:** Con la clase ubicada como un grupo completo, saque un desafío fácil del Paquete de Pilas de Vasos de "Mis Amigos Robóticos" y programe con cada uno como para recordar las reglas y la terminología.

Luego, saque un desafío levemente más difícil, pero que requiera muchos pasos como el de más abajo.

[][2]

**Voluntario/a:** Pida un voluntario/a (o un grupo de voluntarios) que salgan adelante para ayudar a programar este en la pizarra. Si los hace apegarse estrictamente a la regla de "no más símbolos que aquellos en la clave", ¡probablemente le tomará un tiempo!

**Mostrar:** Ahora, muestre esta imagen:

[][3]

¿Cuál es la reacción de la clase?

**Idea:** Dé a los alumnos la oportunidad de hacer una lluvia de ideas sobre formas más cortas de transmitir el código que están a punto de crear. (Se puede saltar esta parte si sus alumnos comienzan a decir cosas como: "Mover hacia adelante 6 veces". Debido a que eso abrirá la discusión de cómo mostrar "seis veces" con símbolos).

Una vez que los alumnos hayan entendido la idea de código de "repetición", deles el vocabulario que incluye.

Asegúrese de compartir con ellos que a menudo los términos "repetir algo" y "loop algo" se usan indistintamente.

## Actividad Principal (45 min)

### Mis Amigos Robóticos

Muestre una copia de **Programación: Mis Amigos Robóticos - Clave de símbolo** (o escriba los símbolos en la pizarra). Párese a un lado dígame a la clase que estos serán los únicos seis símbolos que usarán en este ejercicio. Para esta tarea, le darán a su amigo/a "robot" la instrucción de construir una pila de vasos específica usando solamente las flechas descritas en **Programación: Mis Amigos Robóticos - Clave de símbolo**.

1. Agrupe la clase en pares o grupos de 3 estudiantes. Pida a cada grupo que elija el primer "robot" (debería haber suficiente tiempo para que todos tengan su turno). Pida a los "robots" que salgan de la sala hasta que se les llame de vuelta.
2. Muestre al resto de la clase las **Apilando Ideas - Manipuladores**. Pida a cada grupo que elija que idea le gustaría que hiciera su robot. Intente motivarlos a que hagan una idea fácil primero y luego elijan una más compleja.

#### 💡 Lesson Tip:

Mientras el robot está trabajando en la pila asegúrese de que la clase sepa:

- A los programadores no se les permite hablar cuando el/a robot está trabajando. Esto incluye dar las respuestas o señalar cuando el/a robot ha cometido un error.
- Los programadores deberían levantar su mano si ven un bug.

3. Deje que cada grupo discuta sobre cómo se debe construir cada apilamiento. Luego indique a cada grupo que traduzca el algoritmo en símbolos. Asegúrese de que lo escriban en alguna parte para que lo lea el “robot” después.
4. Una vez que los grupos hayan decidido sus algoritmos, pida a los “robots” que vuelvan a la sala. Es recomendable que continúe mostrando la **Programación: Mis Amigos Robóticos - Clave de símbolo** mientras los robots construyen su pila de vasos, para que puedan recordar lo que significa cada comando.
1. Si un/a estudiante detecta un bug y levanta su mano, permita que el robot termine la instrucción lo mejor que pueda. Luego, pida a los/as estudiantes que discutan sobre el bug y piensen en una solución. Continúe hasta que la pila esté construida correctamente.

**Lesson Tip:**

Esta actividad se puede hacer en un solo grupo grande en caso que su clase no se sienta cómoda trabajando en grupos pequeños.

Una vez que la pila esté lista, puede repetir la actividad eligiendo otro robot.

## Actividad de Cierre (10 min)

### Registrar en un diario

Objetivo: hacer que los alumnos reflexionen sobre la actividad que acaban de experimentar.

Conversación rápida:

Estos son algunos temas posibles:

- ¿Sientes que los bucles hacen la programación más fácil o más difícil?
- ¿Qué otro tipos de cosas de la vida repetimos?
  - Comer: llevarse la comida a la boca, mascar 20 veces
  - Cepillarse el pelo: cepillar el pelo 35 veces
  - Rutinas: despertar, ir a la escuela, volver a casa, acostarse

Sugerencias para el diario:

- ¡Tiempo para el diario! Pide a los alumnos que dibujen una cara de sentimientos en la esquina de la página de su diario para recordarles cómo se sintieron en esta lección.
- Pida a los alumnos que escriban o dibujen algo en su diario que más adelante les recordará lo que son los bucles. Esto puede venir de una idea como:
  - ¿Qué significa repetir para ti?
  - Haz un dibujo de ti mismo repitiendo algo.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.



# Lección 2: Secuencias en Laberinto

Secuencia | Depurar | Bucle

## Descripción

En este conjunto de desafíos, los/as estudiantes comenzarán con una introducción (o revisión dependiendo de la experiencia de su clase) del espacio de trabajo en línea de Code.org. Habrá videos que señalarán la funcionalidad básica del espacio de trabajo, incluyendo los botones Ejecutar, Restablecer y Paso. También se explica en estos videos: Arrastrar, eliminar y conectar bloques de Blockly. A continuación, los/as estudiantes practicarán sus habilidades de *secuenciación* y debugging en el laberinto.

## Propósito

Reconocemos que cada sala de clases tiene un espectro de entendimiento para cada materia. Algunos/as estudiantes de su clase pueden ser magos/as en la computación, mientras que otros/as no tienen mucha experiencia. Con el fin de crear un campo de juego (y aprendizaje) igualitario, hemos desarrollado esta "Etapa de Reforzamiento". Esto puede ser usado como una introducción o como una revisión sobre cómo usar Code.org y los conceptos básicos de Ciencias de la Computación.

## Agenda

### Actividad previa (10 min)

Introducción  
Vocabulario

### Actividades Puente - Programación (10 min)

Actividad Sin Conexión Usando Bloques de Papel  
Previsualización de los Desafíos Online con la clase

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (10 min)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Ordenar comandos de movimiento como pasos secuenciales en un programa.
- Modificar un programa existente para solucionar errores.
- Dividir una secuencia larga de instrucciones en la secuencia repetible más grande.

## Preparación

- ☐ Recorra el **Curso F Desafíos Online - 2018 - Sitio web** para encontrar cualquier área potencial de problema para su clase.
- ☐ Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación** - Recomendaciones de lecciones [Hacer una copia](#)

Para los estudiantes

- **Bloques Blockly Sin Conexión (Cursos 2-5)** - Manipuladores
- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#)

## Vocabulario

- **Bug o error** - parte de un programa que no funciona correctamente.
- **Depurar** - encontrar y solucionar problemas en un algoritmo o programa.

- **Bucle** - la acción de hacer algo una y otra vez.
- **Programa** - un algoritmo que se ha programado como algo que puede ser ejecutado por una máquina.
- **Programación** - el arte de crear un programa.

# Guía docente

## Actividad previa (10 min)

### Introducción

Los/as estudiantes aprenderán muchos conceptos nuevos o revisarán muchos conceptos básicos. De acuerdo con la experiencia de su clase, puede cubrir el siguiente vocabulario o pasar a una actividad de puente. Recomendamos usar las siguientes palabras en oraciones si las definiciones no están cubiertas explícitamente.

### Vocabulario

Esta lección tiene cuatro palabras nuevas e importantes de vocabulario:

- **\*\* Programa \*\*** - Dilo conmigo: Pro - Gra - ma    Un algoritmo que se ha codificado en algo que puede ser ejecutado por una máquina.
- **\*\* Programación \*\*** - Dilo conmigo: Pro - Gra - ma - ción    El arte de crear un programa.
- **\*\* Bug \*\*** - Dilo conmigo: Bug    Un error en un programa que impide que el programa se ejecute como se espera.
- **\*\* Debugging \*\*** - Dilo conmigo: De - Bugging    Búsqueda y corrección de errores en programas.
- **\*\* Loop \*\*** - Dilo conmigo: Loop    La acción de hacer algo una y otra vez.

## Actividades Puente - Programación (10 min)

Esta actividad ayudará a llevar los conceptos sin conexión de "Mis amigos Robóticos" al mundo en línea al que se están moviendo los/as estudiantes. Elija una de las siguientes actividades para su clase:

### Actividad Sin Conexión Usando Bloques de Papel

Similar a "Mis Amigos Robóticos", haga que los/as estudiantes de su clase se formen en parejas. Distribuya a cada pareja varios bloques "complete 1" y "mueva \_" de los **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores. Pida a cada pareja de estudiantes que dibujen un diseño en una Programación con Papel Cuadrulado - Ficha. Luego, pida a los/as estudiantes que trabajen juntos para escribir el programa necesario para dibujar este diseño usando los bloques de papel Blockly. Los/as estudiantes tendrán que escribir "arriba", "abajo", "derecha", "izquierda" en el bloque "mueva \_". Asegúrese de que los/as estudiantes sepan que el programa va de arriba a abajo y que los bloques necesitan tocarse.**

### Previsualización de los Desafíos Online con la clase

Elija un desafío de **Curso F Desafíos Online - 2018 - Sitio web**. Recomendamos el desafío número 6 para esta actividad. Divida a sus estudiantes en grupos de tres o cuatro. Pídales que "programen" a Red, el Angry Bird, para que llegue al cerdo, usando las flechas de "Mis Amigos Robóticos".



Una vez que todos los grupos tengan una respuesta, discuta con la clase sobre el camino.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Los/as docentes juegan un papel vital en la educación de Ciencias de la Computación y en el apoyo a un ambiente en la sala de clases que sea colaborativo y dinámico. Durante las actividades en línea, el papel del/la profesor/a es principalmente de estímulo y apoyo. Las lecciones en línea están pensadas para centrarse en los/as estudiantes, por lo que los/as profesores/as deben evitar intervenir cuando los/as estudiantes se quedan estancados/as. Algunas ideas sobre cómo hacer esto son:

#### 🔔 Teacher Tip:

Enseñe a los/as estudiantes la forma correcta de ayudar a los/as compañeros/as:

- No te sientes en el asiento de tu compañero/a.
- No uses el teclado del/la compañero/a.
- No toques el mouse del/a compañero/a.
- Asegúrate que el/la compañero/a pueda describirte la solución antes de que te retires.

- Utilizar **Programación en Parejas - Video para el estudiante** cada vez que sea posible durante la actividad.
- Anime a los/as estudiantes con preguntas/desafíos para comenzar preguntándole a su pareja.
- Las preguntas sin respuesta pueden ser preguntadas a un grupo cercano, que tal vez ya conozca la solución.
- Recuerde a los/as estudiantes que deben utilizar el proceso de debugging antes de que usted se acerque.
- Pida a los/as estudiantes que describan el problema que están viendo. ¿Qué se supone que haga? ¿Qué es lo que hace? ¿Qué te dice eso?
- Recuerde a los/as estudiantes que se sienten frustrados, que la frustración es un paso en el camino hacia el aprendizaje, y que la perseverancia valdrá la pena. Si un/a estudiante sigue estancado/a después de todo esto, haga preguntas para que el/a estudiante descubra el error por sí mismo/a.

## Actividad de Cierre (10 min)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- Enumera algunos de los bugs que encontraste en tus programas hoy.
- ¿Cuál era su desafío favorito para completar? Dibuja a tu personaje favorito completando un desafío.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ **AP** - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 3: Construyendo los Cimientos

Sin conexión | Perseverancia | Frustración

## Descripción

Los problemas nuevos y sin resolver, son a menudo muy difíciles. Si queremos tener alguna posibilidad de hacer algo creativo, útil e inteligente, entonces necesitamos estar dispuestos a encarar los problemas difíciles incluso si esto significa fracasar algunas veces antes de tener éxito. En esta lección, los/as estudiantes construirán una estructura con materiales comunes. La estructura será probada en su habilidad de sostener un libro por más de diez segundos. La mayoría de los/as estudiantes no podrán lograrlo la primera vez, pero es importante que sigan adelante y continúen intentándolo.

## Propósito

Esta lección enseña que el fracaso no es el fin del viaje, sino un indicio de cómo tener éxito. La mayoría de los/as estudiantes se sentirán frustrados/as en algún momento de esta lección, pero es importante enfatizar que el fracaso y la frustración son pasos comunes hacia la creatividad y el éxito.

## Agenda

### Actividad previa (20 min)

Vocabulario

Intentar, Intentar de Nuevo

### Actividad Principal (20 min)

Construyendo los Cimientos

### Actividad de Cierre (10 min)

Charla Exprés: ¿Qué aprendimos?

Escribir en el Diario

### Aprendizaje Extendido

## Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Describir los pasos para completar un desafío de ingeniería estructural.
- Predecir y analizar posibles problemas en la creación de la estructura.
- Crear una estructura basada en el plan del equipo.
- Revisar el plan y la estructura hasta cumplir el desafío.

## Preparación

- ▢Vea [teacher-video-practicing-persistence][0].
- ▢Vea el [lesson-in-action-foundation][1].
- ▢Imprima [csf-building-a-foundation-activity][2].
- ▢Junte suficientes elementos de construcción (marshmallows o gomitas con mondadientes o palitos de helado) para cada grupo. No tiene que dar ninguna cantidad fija, solamente asegúrese de poner límite a la cantidad de materiales.
- ▢Entregue un [think-spot-journal][3] a cada estudiante.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Construyendo los Cimientos** - Video sin conexión ([descargar](#))
- **Construyendo los Cimientos** - Video de lección en acción
- **Construyendo los Cimientos** - Guía de preparación para el docente

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones

[Hacer una copia](#) ▾

# Vocabulario

- **Frustrado** - sentirse molesto o enojado porque algo no sale como uno quiere.
- **Perseverancia** - intentar algo una y otra vez, incluso si es muy difícil.

# Guía docente

## Actividad previa (20 min)

### Vocabulario

Esta lección tiene una palabra nueva: Perseverancia - Díganlo conmigo: Per - se - ve - ran- cia Intentar una y otra vez, incluso cuando algo es muy difícil.

### Intentar, Intentar de Nuevo

- ¿Todos/as hicieron todo bien la primera vez?
- Cuando era un bebé aprendiendo a caminar, ¿Me paré y corrí en mi primer intento?
- Sometimes, the best and most useful things to do are the hardest to learn.
  - Puede tomar un tiempo aprender cosas difíciles.
  - Si no haces algo bien al principio, ¿Significa eso que nunca lo harás?
  - ¿Puedes pensar en algo que fue difícil al principio, pero que ahora puedes hacer de manera bastante fácil?
    - Caminar
    - Hablar
    - Andar en bicicleta
- Cuando fracasas al hacer algo, tienes un indicio de lo que salió mal. Sólo tienes que buscar.
  - Si tu bicicleta se vuelca, la próxima vez debes trabajar en el equilibrio.
  - Si estás inflando un globo y se revienta, la próxima vez necesitas menos aire.
- Piensa en los errores como oportunidades de aprender o cómo hacer algo mejor, la siguiente vez.

#### 💡 Lesson Tip

Aquí hay grandes recursos para preparar su clase con el concepto de la perseverancia. Antes, dejarlos trabajar en este proyecto:

- **Mouse Wants a Cracker**
- **Fall 7 Times, Stand Up 8**
- **Never Ever Give Up**
- **If You Quit Too Soon**

## Actividad Principal (20 min)

### Construyendo los Cimientos

¿Alguna vez has comenzado una tarea, para luego descubrir que era mucho más difícil de lo que pensaste que sería? Las tareas difíciles pueden hacernos querer renunciar, pero si nos mantenemos apegados a nuestro objetivo y continuamos intentándolo, entonces ¡podríamos hacer algo mejor de lo que nunca hemos hecho!

En este desafío, trabajaremos en construir torres que sean lo suficientemente fuertes para sostener un libro por al menos 10 segundos, usando materiales de la vida diaria.

Reglas:

- Usa sólo los materiales proporcionados para construir la torre.
- Las torres pueden tener cualquier forma, pero tienen que ser al menos del alto de un vaso de papel.
- La torre debe soportar el peso de un libro por 10 segundos.

#### Instrucciones:

1. Divida a los/as estudiantes en grupos de tres o cuatro.
2. Explique las reglas del desafío, mencionadas más arriba.
3. Proporcione a cada grupo una cantidad limitada de materiales e infórmeles que no podrán obtener más.

#### 💡 Lesson Tip

La etapa de planificación puede ser difícil para estudiantes jóvenes. Puede ser útil ubicar algunas ideas de "ejemplo" al frente del aula. No avise que están ahí. Simplemente, anime a los/as estudiantes a dar una vuelta si se frustran. Intente animarlos/as a ubicar las sugerencias por sí mismos, si es posible.

4. Desafíe a la clase a adelantarse al problema y planificar su método de construir su primera torre.
  5. Anime a los/as estudiantes a comenzar la construcción, y luego pídale que le avisen cuando crean que han logrado cumplir con el desafío descrito en las reglas.
  6. Pruebe cada estructura. ¿Es más alta que un vaso? ¿Sostiene el libro?
  7. Si no es así, haga que los/as estudiantes entren en un ciclo de planificación, arreglo, prueba, y planificación nuevamente, hasta que cumplan el desafío.
  8. Felicite a los/as estudiantes cuando tengan éxito y ¡tome fotos de las torres exitosas!
- coursee/building-a-foundation/Main Activity -> content File: Activity. json!

## Actividad de Cierre (10 min)

### Charla Exprés: ¿Qué aprendimos?

- ¿Te sentiste orgulloso/a de lo que hiciste?
- ¿Crees que podrías hacer una torre del alto de una silla que pudiera sostener a una persona?
  - ¿Cuántas gomas de mascar crees que necesitarías?
- ¿Hubo momentos en que pensaste darte por vencido/a?
  - ¿Cómo superaste ese sentimiento?

#### Lesson Tip

Las preguntas de la Charla Exprés están orientadas para que se produzca pensamiento general y saber cómo la lección se relaciona al gran mundo y al aún más gran futuro de los/as estudiantes. Use el conocimiento de su aula para decidir si quiere discutir esto como clase, en grupos, o con un/a compañero/a de banco.

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil, y cómo se sienten puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro. Proporcionamos un Diario Think Spot como una plantilla básica para que los/as estudiantes la usen como su diario.

#### Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- Haz un dibujo de tu estructura.
- ¿Cuáles fueron algunos problemas que encontraste durante la construcción? ¿Cómo solucionaste esos problemas?

## Aprendizaje Extendido

Use estas actividades para mejorar el aprendizaje de los/as estudiantes. Pueden usarse como actividades fuera del aula u otro tipo de enriquecimientos.

### ¡Inténtalo de Nuevo!

Intenta hacer la misma actividad con materiales diferentes.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼



**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 4: Depuración con Scrat

Bug o error | Depurar | Scrat | Era de Hielo

## Descripción

Debugging es un elemento esencial para aprender a programar. En esta lección, los/as estudiantes encontrarán desafíos que han sido resueltos incorrectamente. Necesitarán recorrer el código existente para identificar errores, incluyendo loops incorrectos, bloques faltantes, bloques adicionales y bloques que están fuera de servicio.

## Propósito

Los/as estudiantes de tu clase pueden sentirse frustrados/as con esta lección debido a la esencia del debugging. Debugging es un concepto muy importante para la programación de computadores. Los/as informáticos/as tienen que ser muy buenos/as para enfrentarse a los errores de sus propios programas. Debugging obliga a los/as estudiantes a reconocer los problemas y a superarlos al mismo tiempo que desarrollan el pensamiento crítico y las habilidades de resolución de problemas.

## Agenda

### Actividad previa (15 mín)

Introducción  
Vocabulario

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

### Aprendizaje Extendido

### Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Predecir dónde fallará un programa.
- Modificar un programa existente para solucionar errores.
- Reflexionar sobre el proceso de depuración de manera apropiada para cada edad.

## Preparación

- Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
  - (Opcional) Elija algunos desafíos para hacer con la clase.
    - Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
      - Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.
        - Revise **Receta Debugging - Guía para el estudiante** con la clase.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Diario Think Spot - Diario de reflexiones** [Hacer una copia](#)
- **Receta Debugging - Guía para el estudiante** [Hacer una copia](#)

## Vocabulario

- **Bug o error** - parte de un programa que no funciona correctamente.
- **Depurar** - encontrar y solucionar problemas en un algoritmo o programa.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Pregunte a los/as estudiantes sobre problemas que tienen que resolver en su diario vivir.

- ¿Cómo arreglas algo que no está funcionando?
- ¿Sigues una serie de pasos específicos?
- Los desafíos en esta unidad ya han sido resueltos para ti (¡Yupi!), pero no parecen estar funcionando bien (¡buuu!).
- A los problemas en estos programas los llamamos “Bugs” y será tu trabajo resolverlos.

### Vocabulario

Esta lección tiene tres nuevas e importantes palabras de vocabulario:

- **Bug** - Repitan conmigo: Bag. Algo que está mal. Un error.
- **Debugging** - Repitan conmigo: Di-ba-gin. Encontrar y reparar errores.
- **\_Perseverancia\_** - Repitan conmigo: Per-se-ve-ran-cia. No rendirse. La perseverancia funciona mejor cuando intentas las cosas de diferentes formas, repetidas veces.

Diga:

Debugging es un proceso. Primero, deben reconocer que hay un error en su programa. A continuación, trabajen en el programa paso a paso para encontrar el error. Prueben el primer paso, ¿funcionó? Luego el segundo, ¿qué tal ahora? Si se aseguran de que todo funciona línea por línea, entonces cuando lleguen al lugar donde su código no está haciendo lo que se supone que debe hacer, saben que han encontrado un error. ¡Una vez que hayan descubierto su error, pueden trabajar para arreglarlo (o hacer "debugging")!

Si cree que logrará emocionar a la clase, puede presentar al personaje de los desafíos de hoy, Scrat de La Era del Hielo. Si los/as estudiantes no están familiarizados/as con Scrat, **show some videos** de la peculiar ardilla metiéndose en problemas.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Antes de que los/as estudiantes empiecen a trabajar con el computador, recuérdelos las ventajas de la **Programación en Parejas - Video para el estudiante** y sobre pedir ayuda a sus compañeros/as. Siente a los/as estudiantes en parejas y recomíéndelos que pidan ayuda por lo menos a dos compañeros/as antes de acudir al/la profesor/a.

Como se menciona en el propósito de esta lección, asegúrese de que los/as estudiantes sepan que se enfrentarán a desafíos que pueden resultar frustrantes. Dígalos que está bien sentirse frustrado/a, pero que es importante resolver el problema y pedir ayuda. A medida que los/as estudiantes trabajan en los desafíos, camine por alrededor de ellos/as, para asegurarse de que ningún estudiante se sienta tan estancado/a que no esté dispuesto/a a continuar más.

## Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que han aprendido, por qué es útil, y cómo se sienten puede ayudar a solidificar cualquier conocimiento obtenido hoy y construir una hoja de revisión que pueden revisar en el futuro.

**Sugerencias para el Diario:**

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué tipo de errores encontraste hoy?
- Dibuja un error que hayas encontrado en uno de los desafíos hoy. ¿Qué hiciste para "debug" el programa?

## Aprendizaje Extendido

Utilice estas actividades para mejorar el aprendizaje de los/as estudiantes. Pueden ser utilizados como actividades fuera de la clase u otro enriquecimiento.

### Plantar bugs

Pida a los/as estudiantes que vuelvan a los niveles anteriores, añadiendo errores a sus soluciones. A continuación, pueden pedir a otros/as estudiantes que apliquen debug a su trabajo. Esto también se puede hacer con desafíos de papel.

Cuando otros/as estudiantes estén debugging, asegúrese de que las críticas sean constructivas. Si esto fuera un problema para su clase, repase acerca del debugging respetuoso antes de esta actividad jugando con otro/a estudiante.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 5: Programación en Artista

Artista | Programación | Revisión | Información privada

## Descripción

En esta lección, los/as estudiantes tomarán el control del Artista para completar los dibujos en pantalla. Esta etapa del Artista permitirá a los/as estudiantes crear imágenes de creciente complejidad usando nuevos bloques como moverse hacia adelante por 100 píxeles y girar a la derecha por 90 grados.

## Propósito

Partiendo de la experiencia previa de los/as estudiantes con la secuenciación, esta lección trabajará para inspirar más creatividad con la programación. El propósito de esta lección es consolidar el conocimiento de la secuenciación mediante la introducción de nuevos bloques y objetivos. En este caso, los/as estudiantes aprenden más sobre los píxeles y ángulos usando los nuevos bloques, mientras practican sus habilidades de secuenciación. Además, los/as estudiantes podrán visualizar nuevas metas, tales como programar al Artista para dibujar un cuadrado.

## Agenda

### Actividad previa (10 min)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (10 - 15 min)

Escribir en el Diario

### Aprendizaje Extendido

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Crear un programa para completar una imagen mediante pasos secuenciales.
- Dividir formas complejas en partes simples.

## Preparación

- ☐ Recorra los **Curso F Desafíos Online - 2018 - Sitio web** en la etapa 6 para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ (Opcional) Consiga transportadores para su clase para visualizar los ángulos que deben usar para completar el desafío.
- ☐ Imprima un **Giros & Ángulos - Guía para el estudiante** para cada estudiante.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#)
- **Introducción a Artista** - Video para el estudiante
- **Giros & Ángulos** - Video para el estudiante
- **Giros & Ángulos** - Guía para el estudiante [Hacer una copia](#)

# Guía docente

## Actividad previa (10 min)

### Introducción

Muestre a los/as estudiantes uno o ambos de los siguientes videos como introducción a los ángulos:

**Introducción a Artista - Video para el estudiante** (1,5 minutos de duración)

**Giros & Ángulos - Video para el estudiante** (2 minutos de duración)

Use el **Giros & Ángulos - Guía para el estudiante** para mostrar los ángulos interiores y exteriores de los alumnos para diferentes formas. Este documento se puede usar como una entrega o puede elegir imprimirlo como un póster para que los estudiantes consulten.

Pedir:

Discuta las formas cuadradas y triangulares del documento.

- ¿Cómo codificarías una computadora para dibujar esa forma?
- ¿En qué orden deben estar las instrucciones?

Díales a los/as estudiantes que en estos desafíos moverán a un personaje que deja una línea donde quiera que vaya. Los estudiantes escribirán un código que hace que el personaje dibuje varias formas, incluyendo un cuadrado.

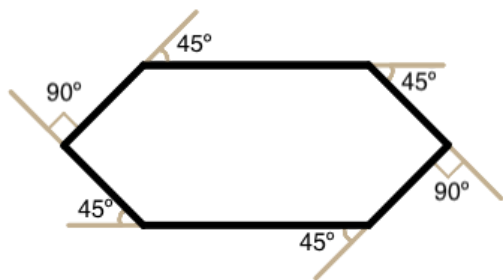
## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

En este conjunto de desafíos, el Artista ya no estará limitado a ángulos de 90 grados. Tener transportadores disponibles puede ayudar a los/as estudiantes a visualizar mejor los ángulos que necesitan. De lo contrario, la etapa proporciona imágenes de los ángulos a medida que el/a estudiante selecciona cual usar. (Nota: Las opciones de ángulo están limitadas a dos dentro del menú desplegable, lo que reduce el número de opciones en las que los/as estudiantes tienen que trabajar).

Antes de enviar a los/as estudiantes a los computadores para trabajar en los desafíos, puede ser útil hacer una breve presentación de cómo usar las herramientas en este nivel. Recomendamos el Desafío 5 como un buen ejemplo, para mostrar cómo usar el transportador en línea.

El octavo desafío pide a los/as estudiantes que dibujen un polígono de 6 lados. Esto puede ser un reto para algunos/as estudiantes. Recomendamos que los/as estudiantes lo intenten unas cuantas veces, que le pregunten a un/a compañero/a y que luego le pidan ayuda al/a profesor/a. A continuación, hay una imagen que puede ser útil para los/as estudiantes.



## Actividad de Cierre (10 - 15 min)

Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

**Sugerencias para el Diario:**

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Cuáles son los ángulos interiores que forman un cuadrado? ¿Qué hay de un triángulo?
- Dibuja una forma simple en tu papel e imagina el código utilizado para dibujarla. ¿Puedes escribir ese código junto a la forma?

## Aprendizaje Extendido

Use las siguientes actividades para mejorar el aprendizaje de sus estudiantes. Pueden ser utilizados como actividades fuera de la clase o en otra oportunidad.

### La fotocopidora

- Entregue a los/as estudiantes dos hojas de papel.
- En una de las hojas, dibuje una imagen sencilla, usando sólo líneas rectas.
- En la segunda hoja, dibuje las instrucciones para recrear la imagen usando comandos para moverse en línea recta y girar en distintos ángulos.
- Intercambie las hojas de instrucciones e intente recrear la imagen usando sólo las instrucciones entregadas.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.



# Lección 6: Mis Amigos Robóticos Loopy

Sin conexión | Bucle | Repetir

## Descripción

Basándose en la actividad inicial "Mis amigos Robóticos", los/as estudiantes aprenden a usar loops cuando programan sus robots para construir estructuras más grandes de manera más eficiente.

## Propósito

Esta lección sirve como una re introducción a los loops, usando el ahora familiar set de instrucciones de programación para el/a "robot". Los/as estudiantes desarrollarán habilidades de pensamiento crítico buscando patrones de repetición en los movimientos de los/as compañeros/as y determinando cómo simplificar esos patrones repetidos usando loops.

## Agenda

Actividad previa (5 min)

Actividad (30 min)

Actividad de Cierre (10 min)

Actividades de Extensión

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Identificar patrones repetidos en el código que podrían reemplazarse con un bucle
- Escribir instrucciones que usen bucles para repetir patrones

## Preparación

- ▢ Vea **Mis Amigos Robóticos Loopy - Video para el docente**.
- ▢ (Opcional) Imprima un **Programación: Mis Amigos Robóticos - Clave de símbolo** para cada grupo o cada 4 estudiantes. Alternativamente, encuentre un ligar donde pueda desplegar esta información, que puede ser usado como referencia para los/as estudiantes durante la lección.
- ▢ Prepare una pila de 20 vasos de papel por cada grupo de 4 estudiantes. O
- ▢ (Opcional) imprima y corte un **resource paper-trapezoid-template not found** para cada grupo si su clase no va a usar los vasos.
- ▢ Imprima un set de **Apilando Ideas - Manipuladores** por grupo.
- ▢ Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Mis Amigos Robóticos Loopy** - Video para el docente

Para los estudiantes

- **Programación: Mis Amigos Robóticos** - Plantilla de trapecios en papel
- **Apilando Ideas** - Manipuladores

- **Programación: Mis Amigos Robóticos** -  
Clave de símbolo

## Vocabulario

- **Bucle** - la acción de hacer algo una y otra vez.
- **Programa** - un algoritmo que se ha programado como algo que puede ser ejecutado por una máquina.
- **Programación** - el arte de crear un programa.
- **Repetir** - hacer algo de nuevo.

# Guía docente

## Actividad previa (5 min)

### Revisión de Mis Amigos Robóticos

**Programación:** Mis Amigos Robóticos - Guía de preparación para el docente

**Objetivo:** Esta revisión refrescará el conocimiento de los alumnos acerca de cuán rápido los programas para la actividad de "Mis Amigos Robóticos" se pueden volver intensos.

**Mostrar:** Muestre el [symbol-key][1] que usamos en Mis Amigos Robóticos. Para cada uno de los seis símbolos, pida a los alumnos que muestren cómo luce para un robot seguir esa instrucción.

**Modelar:** Con la clase ubicada como un grupo completo, saque un desafío fácil del Paquete de Pilas de Vasos de "Mis Amigos Robóticos" y programe con cada uno como para recordar las reglas y la terminología.

Luego, saque un desafío levemente más difícil, pero que requiera muchos pasos como el de más abajo.

[][2]

**Voluntario/a:** Pida un voluntario/a (o un grupo de voluntarios) que salgan adelante para ayudar a programar este en la pizarra. Si los hace apegarse estrictamente a la regla de "no más símbolos que aquellos en la clave", ¡probablemente le tomará un tiempo!

**Mostrar:** Ahora, muestre esta imagen:

[][3]

¿Cuál es la reacción de la clase?

**Idea:** Dé a los alumnos la oportunidad de hacer una lluvia de ideas sobre formas más cortas de transmitir el código que están a punto de crear. (Se puede saltar esta parte si sus alumnos comienzan a decir cosas como: "Mover hacia adelante 6 veces". Debido a que eso abrirá la discusión de cómo mostrar "seis veces" con símbolos).

Una vez que los alumnos hayan entendido la idea de código de "repetición", deles el vocabulario que incluye.

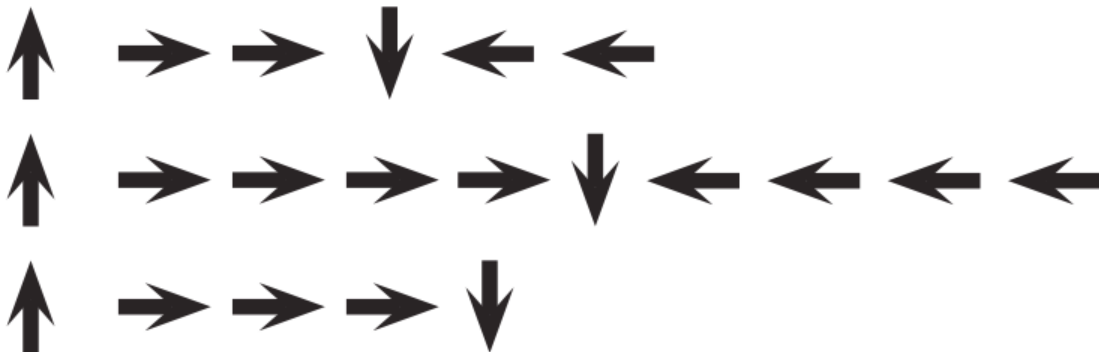
Asegúrese de compartir con ellos que a menudo los términos "repetir algo" y "loop algo" se usan indistintamente.

## Actividad (30 min)

### Introducción y Modelado

**Configurar:** Tenga disponibles pilas de vasos o papel cortado en forma de trapecoide para los grupos.

**Mostrar:** Toma el programa de una de sus pilas de vasos previas y muéstrelo a la clase, o use la que está más abajo.

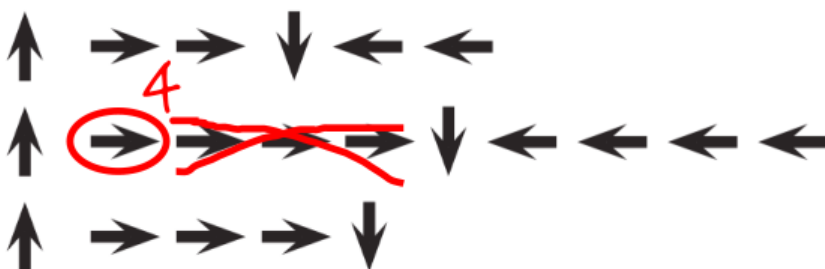


**Pensar:** Pida a los alumnos que piensen en silencio de dónde pueden encontrar en este programa un patrón de instrucciones que se repitan ininterrumpidamente (una repetición después de la otra).

**Parejas:** Vuélvete hacia un vecino y comparte los patrones de repetidos que encuentre.

**Compartir:** Pida a algunos alumnos compartir los patrones que identificaron. Trate de sacar enfoque diferentes de patrones agrupados. Para cada patrón, pida a los alumnos identificar cuántas veces se repite el patrón.

**Modelar:** Usando uno de los patrones repetidos que la clase identificó, modele cómo Encerrar en un círculo la instrucción o patrón que se repite, escriba el número de bucles cerca del círculo, luego haga una cruz sobre el resto de las flechas.



Repita esto hasta que el programa completo haya sido acertado, luego re-escriba el programa de una forma que los alumnos puedan ver cuánto más simple son las instrucciones resultantes.

## Looping Tus Robots

**Agrupar:** Separe a los alumnos en grupos de 4. Cada grupo debería luego subdividirse en dos parejas - cada pareja desarrollará su propio programa "ejecutar" para la otra pareja.

**Distribuir:** Entregue a cada grupo una pila de vasos o recortes de papel.

**Mostrar:** Muestre **Apilando Ideas - Manipuladores** a la clase o entregue copias individuales para que los grupos las usen. Pida a cada pareja (no grupos) que escojan cuál idea les gustaría que su robot hiciera. Incentive a las parejas a que seleccionen un patrón más complejo esta vez.



**Discutir:** Permita que cada grupo discuta cómo debería construirse la pila, luego dé instrucciones a cada grupo para traducir el algoritmo a símbolos. Asegúrese que cada grupo escriba el algoritmo símbolo en algún lugar para que el "robot" lo lea más tarde. A medida que los alumnos trabajan en sus programas, recuérdelos estar alertas a las oportunidades de reemplazar un patrón repetido por un bucle.

**Hacer:** Cuando los grupos hayan terminado sus instrucciones, haga que cada pareja se gire "ejecutando" su código con otra pareja. Recuérdelos estar alertas por errores en su código, pero no interrumpir un robo hasta que termine de ejecutar el programa.

**Discutir:** Cuando todas las parejas hayan tenido la posibilidad de ejecutar sus programas, pídale a algunos que compartan sus soluciones con el curso.

### Teaching Tip

**Buscando Loops:** Asegúrese de estar bien atento cuando los alumnos usen los loops. Trate de evitar corregir sus algoritmos completos o darles una solución, pero sienta la libertad de dirigir a los alumnos hacia los patrones que podrían ser acertados usando un círculo de repetición.

Observe a los alumnos mientras trabajan con el código. ¿Hay errores? Use las preguntas de depuración para ayudarlos a encontrar la solución.

- ¿Qué hace?
- ¿Qué se supone debiera hacer?
- ¿qué te dice eso?
- ¿Funciona en el primer paso?
- ¿Funciona en e segundo paso?
- ¿Dónde deja de funcionar?

Use esta oportunidad para discutir cómo a los grupo se les ocurrieron las diferentes soluciones para el mismo desafío. En particular, debería preguntar de cada programa:

- ¿Cómo identificaron los loops?
- ¿Hay otras formas en que esos bucles pudieran haber sido escritos?
- ¿Cuánto más corto es el programa con loops de lo que sería sin ellos?
- ¿Es el programa más fácil de entender con loops, o escrito a mano? ¿Por qué?

## Actividad de Cierre (10 min)

### Registrar en un diario

Objetivo: hacer que los alumnos reflexionen sobre la actividad que acaban de experimentar.

Conversación rápida:

Estos son algunos temas posibles:

- ¿Sientes que los bucles hacen la programación más fácil o más difícil?
- ¿Qué otro tipos de cosas de la vida repetimos?
  - Comer: llevarse la comida a la boca, masticar 20 veces
  - Cepillarse el pelo: cepillar el pelo 35 veces
  - Rutinas: despertar, ir a la escuela, volver a casa, acostarse

Sugerencias para el diario:

- ¡Tiempo para el diario! Pide a los alumnos que dibujen una cara de sentimientos en la esquina de la página de su diario para recordarles cómo se sintieron en esta lección.
- Pida a los alumnos que escriban o dibujen algo en su diario que más adelante les recordará lo que son los bucles. Esto puede venir de una idea como:
  - ¿Qué significa repetir para ti?
  - Haz un dibujo de ti mismo repitiendo algo.

## Actividades de Extensión

- Pida a sus alumnos que dibujen sus propias pilas de vasos para que alguien las modifique.
- Proporcione a los alumnos algoritmos que utilicen repeticiones, luego pídale que amplíen el programa a una versión completa paso a paso.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 7: Loops en Artista

Bucle | Artista

## Descripción

¡Mira a la cara de los/as estudiantes mientras hacen sus propios diseños maravillosos usando un pequeño número de bloques y pegatinas digitales! Esta lección se basa en la comprensión de loops de lecciones anteriores y da a los/as estudiantes la oportunidad de ser verdaderamente creativos. Esta actividad es fantástica para producir artefactos para portafolios o para reuniones de padres/profesores.

## Propósito

Esta serie destaca el poder de los loops con diseños creativos y personales.

Ofrecido como una secuencia respaldada por el proyecto, esta progresión permitirá a los/as estudiantes construir sobre su propio trabajo y crear artefactos asombrosos.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Identificar los beneficios de usar una estructura de bucles en lugar de la repetición manual.
- Diferenciar entre comandos que deben repetirse en bucles y comandos que deben usarse solos.

## Preparación

- Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**

Para los estudiantes

- **Diario Think Spot - Diario de reflexiones**

## Vocabulario

- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Los/as estudiantes deberían haber tenido mucha introducción en relación a los loops hasta este punto. Basándose en lo usted cree que podría beneficiar a su clase, le recomendamos lo siguiente:

- Crear un nuevo grupo de diseños con loops como en “Mis Amigos Robóticos Loopy”.
- Revisar cómo usar Artista con el desafío “Programando con Artista”.
- Vista previa de un desafío de esta lección.

Todas estas opciones revisarán ya sea los loops o el Artista, lo que le ayudará a preparar su clase para divertirse con los desafíos en línea.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Algunos/as estudiantes pueden descubrir dónde añadir “loops de repetición” escribiendo el programa sin loops y encerrando en un círculo las secciones donde se repite. Si cree que los/as estudiantes de su clase se pudieran beneficiar con esto, pídale que tengan papel y lápiz consigo. Puede que también disfruten dibujando algunas de las formas y figuras en papel antes de programarlo en línea. (Al dibujar estampillas, puede ser más fácil simbolizar aquellos con formas simples como círculos y cuadrados).

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los alumnos escriban acerca de lo que han aprendido, por qué es útil, y cómo se sienten puede ayudar a solidificar cualquier conocimiento obtenido hoy y construir una hoja de revisión que pueden revisar en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Cuál fue la forma o figura más genial que programaste hoy? ¡Dibújala!
- ¿Cuál otra forma o figura te gustaría programar? ¿Puedes crear el código para crearla?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 8: Loops Anidados

Loops Anidados | Bucles | Abeja | Laberinto

## Descripción

En esta actividad online, los/as estudiantes tendrán la oportunidad de comprender acerca de los loops a un nivel completamente nuevo. Jugando con Bee y las Plantas vs Zombies, los/as estudiantes aprenderán a programar un loop dentro de otro loop. También se les animará a averiguar cómo pequeños cambios en cada loop afectarán su programa cuando hagan clic en Ejecutar .

## Propósito

En esta introducción a los loops anidados, los/as estudiantes saldrán de su zona de confort para crear soluciones más eficientes para los desafíos.

En los desafíos anteriores, los loops empujaban a los/as estudiantes a reconocer la repetición. Aquí, los estudiantes aprenderán a reconocer patrones dentro de patrones repetidos para desarrollar estos loops anidados. Esta etapa comienza al animar a los/as estudiantes a que intenten resolver un enigma donde el código es irritante y complejo para escribir. Después de que un video presenta los loops anidados, se muestra a los/as estudiantes un ejemplo y se les pide que predigan lo que sucederá cuando se coloque un loop dentro de otro loop. Esta progresión lleva a mucha práctica para que los/as estudiantes se solidifiquen y desarrollen su comprensión de los loops en la programación. .

## Agenda

### Actividad previa (10 min)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 min)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Dividir las tareas complejas en secciones repetibles más pequeñas.
- Reconocer patrones grandes repetidos conformados de patrones repetidos más pequeños.
- Identificar los beneficios de usar una estructura de bucles en lugar de la repetición manual.

## Preparación

- Recorra **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018** - Sitio web
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación** - Recomendaciones de lecciones

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones

## Vocabulario

- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.



# Guía docente

## Actividad previa (10 min)

### Introducción

Revise brevemente con la clase lo que son los loops y por qué los usamos.

- ¿Qué hacen los loops?
  - Los loops repiten un grupo de comandos. (Vea el vocabulario sobre comando si los/as estudiantes no lo reconocen).
- ¿Cómo usamos los loops?
  - Usamos los loops para crear un patrón hecho de acciones repetidas.

Cuéntele a la clase que ahora harán algo súper genial: usar unos loops dentro de otros loops. Pídales que hagan una predicción sobre qué tipos de cosas estaríamos usando un loop dentro de otro loop.

"¡Si un loop repite un patrón, entonces hacer loops en un loop repetirá un patrón de patrones!"

Los/as estudiantes no necesitan comprender esto de inmediato, así que puede avanzar a los desafíos online incluso si los/as estudiantes parecen un poco confundidos/as.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Recomendamos altamente el **Programación en Parejas - Video para el estudiante** en esta lección. Este puede no ser un tema fácil para la mayoría de sus estudiantes. Trabajar con un/a compañero/a y discutir sobre las posibles soluciones a los desafíos podría aliviar a los/as estudiantes.

También, tenga a mano lápices y papel para que los/as estudiantes anoten sus planes antes de codificar. Algunos desafíos tienen un límite en el número de ciertos bloques que pueden usar, de esta manera, si a los/as estudiantes les gusta anotar la respuesta larga para encontrar repeticiones, el papel puede ser útil.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué es un loop anidado?
- ¿Puedes dibujar un desafío que utilice un loop anidado? Intenta codificar la solución a tu propio desafío.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a  
Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▾

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 9: Loops Anidados con Frozen

Bucle | Bucle anidado | Artista

## Descripción

Ahora que los/as estudiantes saben cómo hacer sus loops, pueden crear muchas cosas hermosas. Esta lección llevará a los/as estudiantes a través de una serie de ejercicios para ayudarles a crear su propio portafolio de imágenes ¡usando las excelentes habilidades de patinaje en el hielo de Anna y Elsa!

## Propósito

En esta serie, los/as estudiantes practicarán anidando loops mientras crean imágenes que estarán emocionados de compartir.

Comenzando con algunas instrucciones, los/as estudiantes harán sus propias decisiones en lo que respecta a la creación de diseños para la repetición. Luego las girarán de diferentes maneras para terminar con una obra de arte verdaderamente única.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Describir cuándo se necesita un bucle o un bucle anidado, o cuándo no se necesita ningún bucle.
- Reconocer la diferencia entre usar un bucle y un bucle anidado.
- Separar el código en secuencias repetibles más grandes mediante bucles y bucles anidados.

## Preparación

- ☐ Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Diario Think Spot - Diario de reflexiones** [Hacer una copia](#)

## Vocabulario

- **Bucle** - la acción de hacer algo una y otra vez.

- **Repetir** - hacer algo de nuevo.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Pida a sus estudiantes que discutan sobre el último grupo de desafíos.

- ¿Qué les gustó o no les gustó?
- ¿Qué desafíos fueron difíciles? ¿Por qué?
- ¿Qué desafíos fueron fáciles? ¿Por qué?
- Si fueras a enseñar los loops anidados a un/a amigo/a, ¿qué dirías para ayudarlos a entender?

Si hay tiempo, de una introducción a los personajes principales de los desafíos de hoy, Anna y Elsa de Frozen. Cuénteles la historia de las hermanas, si es que la clase no la sabe. Para crear emoción, cuénteles que ¡usarán loops anidados para hacer algunos dibujos fantásticos con los patines de hielo de Anna y Elsa!

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Este grupo de desafíos están preparados como una progresión. Esto significa que cada desafío crea la base para el desafío siguiente. Los/as estudiantes disfrutarán creando diseños más y más interesantes, haciendo cambios simples y pequeños al código que ya han escrito.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿De qué trató la lección de hoy?
- ¿Cuándo usas un loop? ¿Cuándo usas un loop anidado?
- Ejercicio para pensar: ¿Puedes hacer de todo un loop anidado? ¿Puedes sólo con un loop normal? ¿Puedes dibujar un ejemplo?
  - Respuesta: Sí, puedes, pero es mucho más difícil. Los loops anidados hacen los programas más simples.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 10: Algoritmos: Tangramas

Sin conexión | Algoritmos

## Descripción

Esta lección nos muestra algo importante acerca de los algoritmos. Mientras mantengas un algoritmo simple, hay muchas maneras de usarlo. Sin embargo, si quieres asegurarte que todos produzcan el mismo resultado, entonces tu algoritmo necesita más detalle. Los/as estudiantes aprenderán la diferencia entre un algoritmo detallado y uno general, mientras juegan con tangramas.

## Propósito

Introduciendo un concepto básico como los *algoritmos* a la clase en una actividad sin conexión, los/as estudiantes que se sienten intimidados con los computadores pueden aun construir los cimientos de la comprensión de estos temas. Los algoritmos son esenciales para la ciencia de la computación. En esta lección, los/as estudiantes aprenderán como traducir las instrucciones en algoritmos y cómo tener un rol en la programación.

## Agenda

### Actividad previa (10 min)

Vocabulario  
Introducción

### Actividad Principal (20 min)

Algoritmos

### Actividad de Cierre (15 min)

Conversación Rápida: ¿Qué aprendimos?  
Escribir en el Diario

### Evaluación (10 min)

Algoritmos de la Vida Real: Tangramas - Evaluación

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Afrontar el desafío de convertir una imagen en instrucciones procesables.
- Transmitir instrucciones a los compañeros de equipo para reproducir una imagen.
- Analizar el trabajo de los compañeros de equipo para determinar si un resultado fue exitoso.

## Preparación

- ▢ Vea el [teacher-video-tangrams-2][0].
- ▢ Reparta a cada estudiante un [think-spot-journal][1].
- ▢ Imprima un [csf-tangram-packet][2] para cada estudiante.
- ▢ Imprima una [csf-tangram-assessment][3] para cada estudiante.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Algoritmos Tangrama** - Video sin conexión ([descargar](#))
- **Set de Tangrama & Paquete de Tarjetas de Imágenes** - Manipuladores
- **Algoritmos de la Vida Real: Tangramas** - Evaluación
- **Algoritmos de la Vida Real: Tangramas** - Guía de respuestas de la evaluación

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones

[Hacer una copia](#) ▾

## Vocabulario

- **Algoritmo** - una lista de tareas para finalizar una tarea.

# Guía docente

## Actividad previa (10 min)

### Vocabulario

- **Algoritmo**

Una lista de pasos para terminar una tarea.

### Introducción

Sus estudiantes puede que hayan o no jugado con tangramas antes. Si han jugado se puede saltar esta parte, y explicar directamente la actividad principal.

Explique a los/as estudiantes que los tangramas se usan usualmente para resolver rompecabezas.

Recibirá un set de 7 “tans” y deberá usarlos todos

(sin sobreponer ninguno) para recrear una imagen que se te entregará. A menudo, esto se hace como actividad individual, y se le permite al/la jugador/a ver la imagen que están tratando de recrear. Muchas veces, puedes poner tus piezas sobre la silueta de la imagen, para asegurarse que la solución es la correcta.

#### 💡 Lesson Tip

Si su curso nunca ha usado los tangramas, puede darles un ejemplo o incluso tener una lección completa con tangramas. Hay varios, muy buenos en internet. **Here** hay una lección que puede hacer en la sala y **here** hay un juego para jugar online.

## Actividad Principal (20 min)

### Algoritmos

Vamos a usar nuestros tangramas en una forma levemente diferente. En lugar de mirar los desafíos e intentar adivinar cuál forma va dónde, vamos a tener desafíos que ya muestran dónde va cada pieza.

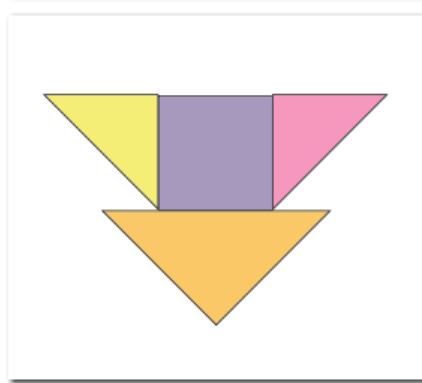
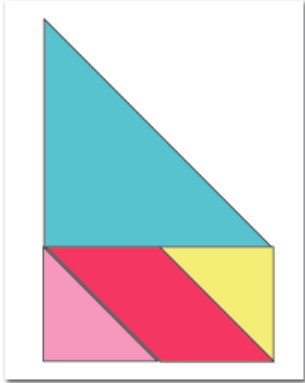
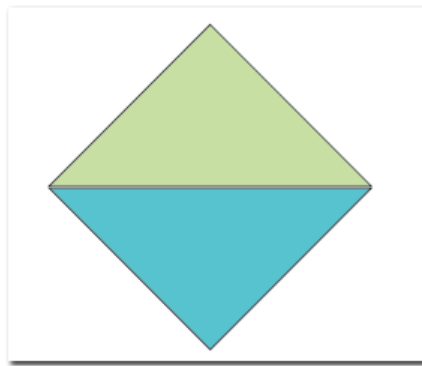
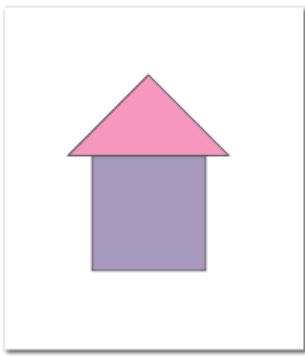
Puede pensar que esto lo hará más fácil, pero no será así, porque los/as estudiantes no verán la imagen que ¡tratarán de recrear!, en lugar de eso, un/a compañero/a le describirá la imagen.

Para evitar que sea muy difícil, usaremos desafíos que requieran usar las 7 piezas.

#### Instrucciones:

1. Divídalos en grupos de 3 a 5 estudiantes.
2. Cada jugador corta su propio set de tangramas.
3. Pida a un miembro de cada grupo seleccionar una Tarjeta de Algoritmo, sin mostrársela a nadie.
4. La persona con la Tarjeta Algoritmo intentará explicar la imagen a los demás sin dejarlos que la vean.
5. Los/as otros/as jugadores/as construirán sus imágenes con la descripción dada por quien tiene la Tarjeta.
6. Cuando el poseedor de la Tarjeta ha terminado, todos los demás mostrarán sus imágenes para ver si todos/as han terminado con la misma imagen.
7. Si todos tienen el mismo dibujo, quien tiene la Tarjeta, la mostrará y verá si todos coinciden con la tarjeta.
8. Si alguna de las imágenes en el grupo son diferentes, pida a quien tiene la Tarjeta, que intente describir la imagen nuevamente, usando más detalle.
9. escoja a un/a nuevo miembro para que tenga una nueva Tarjeta Algoritmo y repítalo hasta que todos hayan tenido la posibilidad de describir una imagen.

Haga el juego varias veces, con imágenes cada vez un poco más difíciles.



## Actividad de Cierre (15 mín)

Conversación Rápida: ¿Qué aprendimos?

- ¿Qué aprendimos hoy?
- ¿Fue más fácil o más difícil de lo que pensaste que sería describir una imagen a otros?
- ¿Terminó algún grupo teniendo coincidencia en todas sus piezas?
- ¿Puedes compartir algunos trucos que se te ocurrieron para ayudar al grupo a hacer coincidir la imagen de la Tarjeta de forma exacta?

Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy, y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el diario:

- ¿Qué aprendimos hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Puedes pensar en algunos trucos para hacer que la descripción de los tangramas a un/a compañero/a, sea más fácil?
- Describe por qué sería necesario ser muy detallista cuando crees algoritmos para escribir un código.

## Evaluación (10 min)

### Algoritmos de la Vida Real: Tangramas - Evaluación

Distribuya la hoja de trabajo de la evaluación y dele tiempo a los/as estudiantes que completen la actividad. Si hay tiempo extra, revise las respuestas con la clase.

## Normas de aprendizaje





This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 11: Introducción a los Desafíos Online

Algoritmos | Bucle | Repetir

## Descripción

En esta lección, los/as estudiantes serán introducidos a secuencias, loops y loops anidados para prepararlos para conceptos más complejos en la siguiente parte del curso. Esta lección de "reforzamiento" equipara el campo de juego entre los/as expertos/as y los/as principiantes de su clase.

## Propósito

Code.org trabaja duro para asegurarse de que todos tengan la oportunidad de aprender Ciencias de la Computación. Estas lecciones "introductorias" crean un nivel de campo de juego para todos/as los/as estudiantes de su clase. Desde principiantes a expertos/as, todos aprenderán los conceptos básicos de la programación en Code.org así como algunos conceptos básicos sobre los que los/as estudiantes se construirán más tarde.

## Agenda

### Actividad previa (5 min)

Introducción

### Actividad Puente (15 mín)

Pre visualización de los desafíos online con la clase

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Categorizar y generalizar código en funciones útiles.
- Dividir una secuencia larga de instrucciones en la secuencia repetible más grande.

## Preparación

- Revise los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- Lea **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**

Para los estudiantes

- **Diario Think Spot - Diario de reflexiones**

## Vocabulario

- **Algoritmo** - una lista de tareas para finalizar una tarea.
- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.

# Guía docente

## Actividad previa (5 min)

### Introducción

Pregunte a la clase lo que aprendieron de la última actividad, "Algoritmo Sin Conexión: Tangramas" y como creen que se relaciona con los computadores.

Explique que los computadores usan algoritmos para resolver problemas básicos y muy complejos. Hoy, crearán algoritmos para resolver desafíos online.

Si su clase se muestra interesada, puede continuar con la discusión sobre la última lección. De lo contrario, continúe a la actividad puente.

## Actividad Puente (15 mín)

Esta actividad ayudará a traer los conceptos sin conexión de "Algoritmos sin conexión: Tangramas" al mundo online al que los estudiantes están entrando. Haga lo siguiente con su clase:

### Pre visualización de los desafíos online con la clase

Muestre un desafío de **Curso F Desafíos Online - 2018 - Sitio web** asociado a esta lección. recomendamos usar el desafío número 5. Pida a la clase que indique cuales son los comandos necesarios para llevar a Scrat hacia la bellota. Es posible que muchos/as estudiantes eviten usar el loop, así que asegúrese de señalar que hay una cantidad limitada de bloques "avanzar" disponibles.

Complete el desafío con la clase y comience una discusión sobre las diferentes formas en las que podría haberse resuelto.

- ¿Habían otros caminos que Scrat pudo haber tomado?
  - ¿Por qué decidimos no seguir esos caminos?
- ¿Por qué usamos un loop? ¿Deberíamos usar loops siempre?

Si se muestran interesados, permita a la clase continuar la discusión con un/a compañero/a a medida que van hacia los computadores. Haga que trabajen con el/la mismo/a compañero/a en los desafíos online.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Los/as docentes juegan un rol vital en la educación de Ciencias de la Computación y en su contribución para un ambiente de clases colaborativo y dinámico. Durante las actividades en línea, el rol del/a profesor/a es principalmente motivacional y de soporte. Las lecciones en línea están pensadas para centrarse en el/la estudiante, por lo que los/as profesores/as deben evitar intervenir cuando los/as estudiantes no pueden avanzar. Algunas ideas sobre cómo hacer esto son:

- Usa la programación en parejas cada vez que sea posible durante esta actividad.
- Anime a los/as estudiantes con desafíos/preguntas para comenzar preguntándole a su compañero/a.
- Las preguntas sin responder pueden ser escaladas a un grupo cercano, que podría ya tener la respuesta.
- Recuerde a los/as estudiantes usar el proceso de debugging antes que usted se acerque.
- Pida a

#### 💡 Recomendación para el/la profesor/a

Muestre a los/as estudiantes la forma **correcta** de ayudar a sus compañeros/as:

- No sentarse en la silla de el/la compañero/a
- No usar el teclado de el/la compañero/a
- No tocar el mouse de el/la compañero/a
- Asegurarse de que el/la compañero/a puede describir la solución en voz alta antes de retirarse.

los/as estudiantes que describan el problema que están viendo. ¿Qué se supone que debe hacer? ¿Qué hace? ¿Qué te dice eso? • Recuerde a los/as estudiantes frustrados/as que la frustración es un paso en el camino del aprendizaje, y que la perseverancia tendrá sus frutos. • Si un/a estudiante aún está estancado después de todo esto, hágale preguntas claves para hacer que los/as estudiantes identifiquen el error por ellos/as mismos/as.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué es un loop y cuándo se utilizan?
- ¿Puedes dar un ejemplo de loops anidados?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ **AP** - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 12: Ciudadanía Digital

Common Sense Education | Acosar en línea

## Descripción

Los/as estudiantes consideran que es posible que mientras disfrutan sus sitios webs favoritos, encuentren mensajes de otros niños/as que los pueden hacer sentir molestos, heridos, tristes o con miedo. Exploran formas de manejar el ciberbullying y cómo responder cuando enfrentan un lenguaje perturbador online.

Los/as estudiantes discuten todas las formas en las que usan la tecnología para comunicarse y exploran las similitudes, y diferencias entre comunicación en persona y online. Los/as estudiantes hacen una lluvia de ideas para responder al ciberbullying.

## Propósito

Esta lección le entregará a los/as estudiantes las herramientas que necesitan para manejar el ciberbullying, si es que se encuentran alguna vez en la situación de tener a alguien que responda negativamente a sus posts online.

Es posible que los/as estudiantes no hayan tenido la mala suerte de experimentar el ciberbullying, pero deberían comprender lo que es, para que puedan identificarlo cuando estén online. Los/as estudiantes aprenderán cómo identificar el ciberbullying y qué pasos tomar para detenerlo. Esto puede ser útil en desafíos posteriores cuando los/as estudiantes tengan la posibilidad de compartir su trabajo.

## Agenda

### Actividad previa (5 min)

Introducción

### Actividad Principal (35 min)

¿Cuál es el problema?

Cruzando la Línea

Hablar y Actuar

### Actividad de Cierre (15 min)

Conversación Rápida: ¿Qué aprendimos hoy?

Pregunte:

### Evaluación (10 min)

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Empatizar con aquellos que han recibido mensajes malos e hirientes.
- Juzgar lo que significa cruzar la línea de comunicación inofensiva a dañina en línea.
- Generar soluciones para lidiar con el ciberacoso.

## Preparación

- ☐ Revise la **Common Sense Education - El Poder de las Palabras - Sitio web de Common Sense Education** y muéstrelo a la clase.
- ☐ Imprima la **Guía Las Palabras Pueden Herir del Common Sense Education - El Poder de las Palabras - Sitio web de Common Sense Education** (pág. 7) para cada grupo de 4.
- ☐ Imprima la **Guía Hablar y Actuar del Common Sense Education - El Poder de las Palabras - Sitio web de Common Sense Education** (pág. 6) para cada estudiante.
- ☐ Imprima la evaluación en las pág. 8-9 del **Common Sense Education - El Poder de las Palabras - Sitio web de Common Sense Education**.
- ☐ Tenga lápices de colores y una cuerda del largo de la sala.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Common Sense Education - El Poder de las Palabras - Sitio web de Common Sense Education**
- **Common Sense Education - Sitio web**
- **Ciudadanía Digital de Fundamentos de Ciencias de la Computación - Lista de**

recursos

#### Para los estudiantes

- **El poder de las Palabras** - Video de la lección
- **Diario Think Spot** - Diario de reflexiones
- **Caras de Emociones** - Imágenes de emociones

## Vocabulario

- **Ciberacoso** - usar herramientas tecnológicas para molestar deliberadamente a alguien.

# Guía docente

## Actividad previa (5 min)

### Introducción

**Dibuje** una serie de caras con expresión en la pizarra. Revise **Caras de Emociones - Imágenes de emociones** para obtener ejemplos.

**Invite** a los/as estudiantes a sugerir emociones que se relacionen con las expresiones en la pizarra. Con cada sugerencia, escriba la emoción al lado de la cara. Las respuestas pueden variar.

**Diga** a los/as estudiantes que no todos reaccionarán de la misma forma a una situación particular, pero sólo porque una reacción es diferente de la propia, no significa que debamos descartar los sentimientos de otros.

**Explique** a los/as estudiantes que van a ver un video acerca de cómo las palabras, ya sean escritas o habladas, pueden impactar cómo alguien se siente.

**Muestre** a los estudiantes el **El poder de las Palabras - Video de la lección**.

Pregunte:

- ¿Quién ha oído el dicho: "Palos y piedras pueden quebrar mis huesos, pero las palabras nunca me dañarán"? (En inglés: "Sticks and stones may break my bones, but words will never hurt me")
- ¿Qué quiso decir Guts en su texto con las palabras a veces pueden herir?
  - Las palabras son poderosas. A veces es difícil ignorar lo que alguien dice, cuando es un sobrenombre. Los nombres *pueden* hacerte sentir triste o herido/a.

**Recuerde** a los/as estudiantes de mantener la pregunta en mente durante esta lección: *¿Cómo tratas a otros/as online?*

## Actividad Principal (35 min)



### ¿Cuál es el problema?

**Organice** a los/as estudiantes en grupos de cuatro y pida a cada grupo que elija una persona para anotar las ideas.

**Distribuya** la **Guía Las Palabras Pueden Herir**. Pida a los/as estudiantes que lean la situación de Rani y Aruna recibiendo mensajes ofensivos en un sitio de juego para niños.

**Pida** a cada grupo responder las preguntas, luego pídale compartir las respuestas con la clase. Busque respuestas que muestren empatía con Rani y Aruna y reconozcan que los mensajes enviados a ellos eran ofensivos y dañinos. Pregunte a los/as estudiantes que lean la sección "¡Usa el Sentido Común!" de la **Guía del/a Estudiante - Las Palabras Pueden Herir**.

**Invite** a los/as estudiantes a compartir sus historias.

Pregunte:

- ¿Qué razones pudo tener la persona para escribir "Eres raro/a"?
  - Siguen una broma interna: el primero/a hizo algo tonto antes; un grupo de niños/as está molestando al/la niño/a; la persona que lo envió, realmente piensa que la persona es rara, pero le da temor decirlo cara a cara.
- ¿Cómo se siente el/a compañero/a cuando le llaman "raro/a"?
  - Posiblemente como la otra persona estaba bromeando, pero quizás esa persona estaba molestando o siendo hiriente.

**Diga** a un integrante de cada pareja que le diga al otro, "Eres raro/a" con una sonrisa en la cara.

Pregunte:

- ¿Por qué te sentirías diferente si pudieras ver al otro?
  - La gente da señales no verbales, por medio de expresiones faciales y lenguaje corporal.

## Cruzando la Línea

**Ubique** la cuerda a lo largo de la sala. Pida a los/as estudiantes que se paren a un lado de la línea. Luego pídeles que imaginen que están online y alguien les ha enviado un mensaje, que usted les va a leer. Pídale a los/as estudiantes que se queden dónde están si creen que el mensaje está bien; cruzar al otro lado de la línea si creen que el mensaje no está bien; o pararse en la línea si creen que el mensaje puede estar en medio.

Lea:

- Eres mi amigo/a.
- Eres un/a idiota.
- Tengo una fiesta y no estás invitado/a.
- Me gusta tu nuevo corte de pelo.
- Eres feo/a.
- Gracias por el consejo. La próxima vez, ¿Me lo dirías en persona en vez de por mensaje?
- ¿Terminaste tu tarea?
- ¿Por qué te estás demorando tanto en terminarla?
- Eres muy extraño/a.

**Revise** con los/as estudiantes que a los/as niños/as les gusta estar online y usar los celulares para enviar emails, chatear, ver videos, enviar mensajes, jugar y hacer tareas. Pero a veces, el lenguaje se puede volver ofensivo o puede asustar. Los mensajes que hacen que la gente se sienta mal, cruzan la línea. A veces, esa maldad es sin intención, pero cuando la gente usa herramientas como internet y los celulares para molestar a la gente de forma deliberada una y otra vez, eso es ciberbullying.

## Hablar y Actuar

**Pida** a los/as estudiantes que vuelvan a sus asientos.

**Discuta** lo fácil que es enojarse cuando alguien les envía mensajes online ofensivos o que asustan.

Defina:

- **Ciberbullying:** Usar las herramientas tecnológicas como internet o los celulares para molestar deliberadamente a alguien.

**Explique** que los abusadores intentan de forma deliberada hacerte sentir de esa forma, igual que los abusadores en la vida real.

Discuta:

- Calmarse puede ser un primer paso cuando recibes un mensaje ofensivo online. Respirar profundo, contar de 10 a 1, o hacer una pausa para pensar acerca de lo que harás a continuación, puede darte tiempo de pensar en la MEJOR manera de manejar la situación.
- Encontrar ayuda o contarle a un adulto de confianza o un/a amigo/a, puede ser una buena forma de actuar. No deberías lidiar con una situación de ciberbullying solo/a. A la persona que le cuentes, debe ser una persona que quiera escuchar lo que tienes que decir y que te ayudará a encontrar una solución. Los adultos pueden ser especialmente buenos, porque a veces ellos tienen el poder de influenciar la situación o pueden darte consejos de lo que puedes hacer.
- Ignorar a la persona que está haciendo ciberbullying puede ser efectivo. A los que hacen bullying generalmente les gusta la atención.
- Lo que sea que hagas, recuerda guardar una copia de tu comunicación con la persona que te está haciendo ciberbullying. Si borras los mensajes, no habrá prueba de cómo el que hace bullying te trató, si es que necesitas mostrarlo a algún adulto de confianza.

**Distribuya** la **Guía Del/la Estudiante Hablar y Actuar** a cada estudiante. Anímelos a representar el argumento y una posible solución. Pueden usar lápices y papel para hacer comics.

## Actividad de Cierre (15 mín)



## Conversación Rápida: ¿Qué aprendimos hoy?

Puede usar estas preguntas para evaluar la comprensión de sus estudiantes de los objetivos de la lección. Puede pedirle a sus estudiantes que reflexionen escribiendo acerca de estas preguntas en sus **Diario Think Spot - Diario de reflexiones**.

### Pregunte:

- ¿Por qué es una mala idea enviar mensajes ofensivos o que asusten online?
  - Porque pueden hacer sentir: molesto/a, enojado/a o asustado/a, a la persona que recibe el mensaje.
- ¿Por qué podría haber mal entendidos entre la gente cuando se envían mensajes online a diferencia de las discusiones cara a cara?
  - Los mensajes online pueden ser más confusos o producir más temor que los mensajes cara a cara, porque no hay señales que ayuden a comprender las intenciones de la gente.
- ¿Qué pueden hacer los/as niños/as cuando reciben mensajes de ciberbullying online?
  1. Permanecer calmados y respirar profundamente.
  2. Contarle a un/a amigo/a o adulto de confianza que pueda desarrollar un plan para manejar la situación.
  3. Ignorar al/la que hace bullying.
  4. Guardar una copia de las comunicaciones que tiene con el abusador.

### Sugerencias para el diario

- ¿De qué se trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué es ciberbullying?
- ¿A quiénes puedes recurrir si alguna vez te hacen bullying online o en persona?

## Evaluación (10 min)

Imprima la evaluación de **Common Sense Education - El Poder de las Palabras - Sitio web de Common Sense Education** (páginas 8-9) y distribúyala a la clase. Dé a los estudiantes tiempo suficiente para completar la evaluación, pero asegúrese de que hay tiempo suficiente para pasar las respuestas.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ NI - Redes y la Internet



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 13: Eventos en La Era de Hielo

Evento | Era de Hielo

## Descripción

En esta lección, los/as estudiantes son llevados a una historia que presenta personajes de la Era del Hielo. Los/as estudiantes trabajarán con eventos y loops para hacer que sus personajes se muevan por la pantalla, y tendrán la oportunidad de crear su propio juego o historia después de los niveles guiados.

## Propósito

Los/as estudiantes usarán eventos para hacer que los personajes de La Era del Hielo se muevan por la pantalla, muestren mensajes e interactúen con otros personajes según el usuario. Esta lección ofrece una entretenida introducción a los eventos en la programación, al mismo tiempo que proporciona la oportunidad de mostrar su creatividad. Al término de esta secuencia de desafíos, los/as estudiantes podrán compartir sus proyectos con sus amigos y familia.

## Agenda

### Actividad previa (15 mín)

Introducción

Revisión de "El Poder de las Palabras"

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Identificar acciones que se correlacionan con eventos de entrada.
- Crear un juego animado e interactivo mediante secuencia y eventos.

## Preparación

- ☐ Recorra **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones.**
- ☐ Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones.**

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018** - Sitio web
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación** - Recomendaciones de lecciones [Hacer una copia](#) ▾

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#) ▾

# Guía docente

## Actividad previa (15 mín)

### Introducción

Hoy, los/as estudiantes aprenderán a usar *eventos* en programación.

- **Evento:** Una acción que provoca que otra cosa pase.

Pida a los/as estudiantes que levanten la mano.

Lo que hizo fue declarar un evento. Cuando dice “levanten sus manos”, los/as estudiantes responden levantando la mano. En programación, usted declararía esto diciendo “cuando yo diga levanten la mano, ustedes levantan sus manos”.

También puede pensar en ciudades que declaran eventos. Hay leyes que dicen “cuando hay una luz verde, los autos pueden pasar la intersección”. Pregunte a los/as estudiantes por qué creen que esto es un evento.

Hoy, los/as estudiantes usarán los eventos para hacer un juego con personajes de la Era del Hielo, pero los eventos que usarán serán más como video juegos. Los eventos tomarán forma de acciones, tales como hacer clic en la flecha hacia arriba o dos personajes encontrándose.

### Revisión de “El Poder de las Palabras”

Recuerde a los/as estudiantes lo que significa el ciberbullying, mediante una lista de cosas que están bien decir en línea y las que no.

<b>Está bien decir en línea</b>	<b>Ciberbullying - NO está bien decir</b>
Eres mi amigo/a. Me gusta tu corte de pelo. ¿Terminaste tu tarea?	Eres un/a idiota. Voy a hacer una fiesta y tú no estás invitado/a. Eres feo/a. Eres tan raro/a.

Discuta sobre otros ejemplos para las dos categorías anteriores.

Los/as estudiantes tendrán la oportunidad de escribir mensajes que los personajes dirán. Asegúrese de que sepan que nunca está bien decir cosas malas en línea.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Esta es una actividad muy creativa con grandes oportunidades para la creatividad. Es posible que quiera proporcionar pautas estructuradas sobre el tipo de juego que se debe crear, especialmente para los estudiantes que se sienten abrumados por demasiadas opciones.

#### 💡 Teacher Tip

Recuerde a sus estudiantes que deben compartir su trabajo sólo con sus amigos cercanos y familiares. Para mayor información, vea o muestre a su clase **Detente y Piensa Online - Video**.

## Actividad de Cierre (15 mín)

Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

**Sugerencias para el Diario:**

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué es un evento?
- ¿Cómo hacen los eventos que los programas sean súper geniales?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ **AP** - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 14: Condicionales en Minecraft

Condicional | Minecraft

## Descripción

Esta lección da a los/as estudiantes la oportunidad de aprender y practicar los condicionales. Presenta personajes y escenarios de Minecraft, donde los/as estudiantes completan tareas tales como la minería y la construcción de estructuras usando sus programas.

## Propósito

Este conjunto de desafíos trabajará para solidificar y construir sobre el conocimiento de los condicionales y loops. Al combinar estos dos conceptos, los/as estudiantes podrán explorar el potencial para crear programas complejos e innovadores en un ambiente nuevo y emocionante.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

### Aprendizaje Extendido

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Definir las circunstancias en que ciertas partes de un programa deben ejecutarse y cuándo no.
- Determinar si se cumple un condicional según los criterios.

## Preparación

- Recorra **Curso F Desafíos Online - 2018 - Sitio web** relacionados a esta lección para encontrar posibles áreas problemáticas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- Curso F Desafíos Online - 2018 - Sitio web**
- Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- Diario Think Spot - Diario de reflexiones** [Hacer una copia](#)

## Vocabulario

- Condición** - una sentencia que un programa verifica si es verdadera o falsa. Si es verdadera, se realiza una acción. De lo contrario, se ignora.

- **Condicionales** - sentencias que solo se ejecutan bajo ciertas condiciones.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Reúna a toda la clase y pida dos voluntarios/as para que caminen en línea recta hacia alguna dirección dentro de la sala. Si se encuentran una silla fuera de lugar deben pasar sobre ella, si llegan a una muralla deben sentarse.

Una vez que todos/as estén sentados/as, pregunte cómo programarían un robot para responder a una muralla o a una silla. Recuérdele que no pueden decir simplemente “Pasa sobre la silla”, a menos que ustedes sepan que hay una silla, y no siempre sabrán que hay una silla. Podría ser útil traducir la tarea en instrucciones como:

- Mientras haya un camino hacia adelante
  - Caminar hacia adelante.
  - Si hay una silla, pasar sobre ella.
- Sentarse

Diga a los/as estudiantes que estarán usando los *condicionales* para resolver este problema en Code.org. Dé las definiciones de:

- **Condición:** Una afirmación que un programa revisa si es verdadera o falsa. Si es verdadera se toma una acción. De lo contrario, la acción es ignorada.
- **Condicionales:** Afirmaciones que sólo se ejecutan bajo ciertas condiciones.

Abra una discusión de cuándo deberían usar un condicional en sus códigos.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Los/as estudiantes se van a deleitar con esta lección. Es probable que la mayoría de sus estudiantes hayan oído hablar de Minecraft, pero dé una breve introducción para aquellos/as que no lo conozcan.

Minecraft es un juego de cubos. Puedes jugar como Alex o Steve mientras trabajas en los laberintos. Tendrás que evitar la lava, recoger objetos y explorar un mundo hecho de cubos con cosas.

Pregunte a los/as estudiantes si alguna vez han jugado a Minecraft. Si ninguno lo ha hecho, pase a la actividad principal. Si algunos lo han hecho, pídale a esos/as expertos/as que expliquen el juego a la clase. Si todos en la clase ya han jugado, siga adelante y pase a los desafíos en línea.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten. Puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy, y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué se trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué disfrutaste de los desafíos de hoy?
- ¿Cuándo utilizaste los condicionales en esta lección? ¿por qué los usaste?

## Aprendizaje Extendido

Más Minecraft

Si encuentra que su clase realmente goza del entorno de Minecraft, **aquí hay algunos enlaces a otros juegos de Minecraft que pueden jugar en línea**. Estos juegos también enseñarán habilidades básicas de codificación.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.



# Lección 15: Variables: Variables en Sobres

Sin conexión | Variable

## Descripción

Las variables se usan como parámetros de ubicación para valores como números y palabras. Las variables entregan mucha libertad en programación. En lugar de tener que escribir una frase muchas veces o recordar un número desconocido, los científicos de la computación pueden usar las variables para mencionarlos. Esta lección ayuda a explicar lo que son las variables y cómo podemos usarlas de diferentes formas. La idea de las variables no es un concepto fácil de entender, así que recomendamos dar suficiente tiempo para discutir al final de la lección.

## Propósito

Las variables son muy útiles en programación. Se le presentará este tema a los/as estudiantes usando sobres para representar a las variables a las que se les han dado nombres. El valor de la variable será escrito en una tarjeta dentro de un sobre. Esta lección ayuda a los/as estudiantes a comprender cómo los nombres pueden ser un parámetro de ubicación para valores en el mundo físico, así programar con variables parecerá menos confuso en el mundo virtual.

## Agenda

### Actividad previa (10 min)

Vocabulario  
Introducción

### Actividad Principal (20 min)

Variables de sobre - Ficha

### Actividad de Cierre (10 min)

Conversación Rápida: ¿Qué aprendimos?  
Escribir en el Diario

### Evaluación (10 min)

Variables de sobre - Evaluación

### Aprendizaje Extendido

### Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Identificar variables y determinar sus valores.
- Definir y llamar variables en el contexto de actividades de la vida real.
- Crear situaciones que requieren el uso de variables.

## Preparación

- Vea el [teacher-video-envelope-variables-2][0].
- Consiga 6 o más sobres vacíos para la actividad previa y algunos más para la actividad principal.
- Imprima una [csf-envelope-variables-worksheet][1] por estudiante.
- Imprima una [csf-variables-assessment-worksheet][2] para cada estudiante.
- Entregue a sus estudiantes sobres, papel, lápices y lápices mina.
- Asegúrese de que cada estudiante tenga un [think-spot-journal][3].

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Variables en Sobres** - Video sin conexión ([descargar](#))
- **Variables de sobre** - Guía de respuestas en fichas [Hacer una copia](#)
- **Variables de sobre** - Guía de respuestas de la evaluación [Hacer una copia](#)

Para los estudiantes

- **Variables de sobre** - Ficha [Hacer una copia](#)
- **Variables de sobre** - Evaluación [Hacer una copia](#)

- **Diario Think Spot** - Diario de reflexiones

Hacer una copia ▾

## Vocabulario

- **Variable** - un marcador de posición para una información que puede cambiar.

# Guía docente

## Actividad previa (10 min)

### Vocabulario

Esta lección tiene un término nuevo:

- **Variable** - Díganlo conmigo: Va-ria-ble

Un parámetro de ubicación para información que puede cambiar.

### Introducción

Pida cuatro voluntarios/as al frente de la sala y póngalos/as en fila. Infórmeles que van a escribir un poema para cada uno/a de ellos/as.

En la pizarra (o proyectado) escriba la oración para su primer estudiante (suponga que es Pedro):

"Mi estudiante Pedro, parado orgulloso  
, es un buen ejemplo para la gente"

Anime a los/as estudiantes a aplaudir por sus habilidades y agradézcale a Pedro por ser voluntario. Pídale a Pedro que vuelva a su puesto (o que vaya al final de la fila) mientras borra la pizarra, luego llame al siguiente voluntario/a (digamos que su nombre es Ana).

"Mi estudiante Ana, parada orgullosa  
, es un ejemplo para la gente"

Nuevamente, aceptando los aplausos, borre la pizarra e invite al/a siguiente voluntario/a.

"Mi estudiante Jenny, parada orgullosa  
, es un ejemplo para la gente"

Mientras llama al/a último voluntario/a, pregunte si a todos en la clase les gustaría un poema escrito sobre ellos. ¿Quizás uno para todos en la escuela? ¡Cielos, eso va a tomar tiempo! Plantee esta pregunta a sus estudiantes:

"¿Cómo podría hacer esto más rápido?"

Sus estudiantes posiblemente se fijen en el hecho de que una palabra cambia, y es palabra es el nombre de la persona. Ayúdelos a ver la ubicación haciendo un círculo en el nombre Jenny en la pizarra y escriba "primerNombre" al lado.

"Tomaría mucho tiempo escribir un poema para todos en la escuela si no pudiera comenzar hasta saber de quién estoy escribiendo, ¿cierto?"

- ¿Cuánto tiempo crees que llevaría hacer video juego si no pudieran comenzar hasta que supieran tu nombre de usuario?
- ¿Qué tan caros serían los videos juegos si tuvieran que ser creados por separados para cada persona?
- ¿Cómo crees que podríamos evitarlo?

Para entonces, es muy probable que a su clase se le ocurra la idea de tener parámetro de ubicación. Con eso, están muy bien direccionados para comprender a dónde va esta lección.

- ¿Cómo llamaríamos a ese parámetro de ubicación?
  - Necesitamos ponerle un nombre que tenga sentido. No le daríamos el nombre "edad" si fuera un parámetro de ubicación para su nombre, ¿cierto?

Ahora, vamos a agregar más voluntarios/as. Dele a cada uno/a un trozo de papel para escribir sus nombres y pídale que lo pongan dentro de un sobre etiquetado "primerNombre".

Esta vez, ponga el poema en la pizarra con un espacio en blanco etiquetado "primerNombre" donde irá el nombre del/la estudiante.

- Pida al/la primer/a estudiante en la fila (posiblemente el/a último/a del ejemplo previo) que saque su nombre del sobre y eso es lo que usted escribirá en el espacio.
- Cuando borre la pizarra, solo borre el nombre del/a último/a estudiante.
- Llame al/a siguiente estudiante para mostrar su variable.
- Repita esto muchas veces mientras sea entretenido.

Llegó la hora de hacer la actividad principal.

## Actividad Principal (20 min)

### Variables de sobre - Ficha

Una vez que los/as estudiantes comprendan cómo los sobres se relacionan a las oraciones, entregue la guía de la actividad y deles tiempo para que preparen algunas variables propias.

#### Instrucciones:

- Divida a los/as estudiantes en grupos de 2 a 4.
- Pida a los/as estudiantes diseñar (dibujar) un robot.
- Después de 10-15 minutos, pida a los/as estudiantes que llenen sus sobres con detalles importantes de su robot como el nombre, la estatura, y el propósito.
- Recoja los sobres de cada grupo, luego llévelos al frente de la sala para compartir.
- Escriba en la pizarra: “El nombre de mi Robot es nombreRobot, mide númUnidad, y su propósito es propósito”.
- Use los sobres para llenar la variable apropiada en la oración, luego pida a cada grupo pararse cuando escuchen la oración que describe su creación.

## Actividad de Cierre (10 min)

### Conversación Rápida: ¿Qué aprendimos?

- ¿Qué aprendimos hoy?
- ¿Puedes pensar lugares dónde hayas visto variables antes?
- ¿Hay al menos una variable en la parte superior de la guía? ¿Puedes pensar cuál es?
- ¿Por qué crees que los profesionales no ponen espacios en las variables para nombres?
  - ¿Qué sucedería si hubiera una variable “ojo”, una variable “color” y una variable “color ojo”?
- Las variables también se pueden usar para almacenar números.
  - Suponga que tengo sobres etiquetados núm1 y núm2, y luego escribo  $\text{núm1} + \text{núm2}$ .
  - ¿Qué sucede si el sobre “núm1” contiene el número 4 y el “núm2” contiene el número 5?

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

#### Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué es una variable?
- ¿Por qué crees que las variables son importantes en programación?

## Evaluación (10 min)

### Variables de sobre - Evaluación

Dé tiempo suficiente a sus estudiantes para terminar esta evaluación. Si está dispuesto a pasar más tiempo en esta actividad, revise las respuestas con la clase.

# Aprendizaje Extendido

Use estas actividades para mejorar el aprendizaje de sus estudiantes. Pueden usarse como actividades fuera del aula u otros enriquecimientos.

## ¿Qué hay en el cuadrado?

- Dibuje cuadrados en un papel con operaciones matemáticas simples entre ellos.
  - Por ejemplo  $[\ ] + [\ ] = [\ ]$
- Tenga cuadrados de tamaños similares con números entre 1 y 20.
  - Pida a uno/a de los/as estudiantes crear una ecuación real, usando los número proporcionados.
- Una vez que el/a estudiante ha finalizado (y la clase ha verificado la ecuación) cambie uno de los números por otro, luego quite un segundo número completamente.
  - Diga a los/as estudiantes que hay un número escondido en el cuadrado vacío que hace que la ecuación esté correcta nuevamente.
  - ¿Qué número hay en el cuadrado?
- Haga este juego varias veces hasta que pueda quitar el número de cualquier ubicación y los/as estudiantes puedan descubrir cuál se supone que es.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 16: Variables en Artista

Variable | Artista

## Descripción

En esta lección, los/as estudiantes explorarán la creación de diseños repetitivos utilizando variables del entorno del Artista. Los/as estudiantes aprenderán cómo se pueden usar las variables para hacer que el programa sea más fácil de escribir y más fácil de leer. Después de los desafíos guiados, los/as estudiantes terminarán en un nivel de juego libre para mostrar lo que han aprendido y crear sus propios diseños.

## Propósito

Las variables son esencialmente marcadores de posición para valores desconocidos al momento de ejecutar el programa o para valores que pueden cambiar durante la ejecución de este. Estas construcciones son vitales para crear código dinámico porque permiten que su programa cambie y crezca basado en cualquier número de modificaciones potenciales. Esta etapa enseña a los/as estudiantes qué son las variables, utilizando las capacidades más básicas de ajuste y utilización.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Puente - Variables (15 mín)

Actividad Sin Conexión usando Bloques de Papel  
Previsualización de los Desafíos Online con la clase

### Actividad Principal (30 mín)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Asignar valores a las variables existentes.
- Utilizar variables en lugar de valores repetitivos dentro de un programa.
- Usar variables para cambiar valores dentro de un bucle.

## Preparación

- ☐ Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Variables** - Video para el estudiante ([descargar](#))
- **Bloques Blockly Sin Conexión (Cursos 2-5)** - Manipuladores
- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#)

## Vocabulario

- **Variable** - un marcador de posición para una información que puede cambiar.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Esta es la primera lección online que trata sobre variables, por lo que podría ser un ejercicio útil revisar la actividad sin conexión "Variables en Sobre", así como el vocabulario que se introdujo en esa lección.

- ¿Qué es una variable? (Un marcador de posición para una información que puede cambiar).
- ¿Cuándo puede ser útil una variable? (Cuando usted no sabe qué información se va a utilizar en un lugar determinado hasta el tiempo de ejecución, o cuando tiene muchos lugares en los que se va a utilizar una pieza de información, pero esa información podría cambiar algún día).

Pregunte a la clase cuándo podrían ver que una variable es útil para la programación. ¿Cuándo NO querrían usar una variable?

Si la clase parece interesada, continúe la discusión. De lo contrario, pase a una de las Actividades Puente.

## Actividad Puente - Variables (15 mín)

Esta actividad ayudará a llevar los conceptos sin conexión de "Variables en Sobre" al mundo en línea al que se están moviendo los/as estudiantes. Elija *una* de las siguientes opciones para realizar con su clase:

### Actividad Sin Conexión usando Bloques de Papel

Comience con una oración o párrafo en la pizarra que contenga por lo menos uno (y hasta cinco) espacios en blanco con nombre. Entregue bloques de papel a cada grupo de estudiantes. Desafíe a cada grupo a "establecer" sus variables para cada espacio en blanco. Llame a los diferentes grupos de estudiantes para compartir su tarea de cada variable y ver qué pasa.

```
[[[0]{: style='width:500px'}]
```

Ahora, cambie la oración a una ecuación matemática. ¿Qué pasa a la oración " $X + Y =$ " cuando los/as estudiantes le asignan valores diferentes a las variables X & Y?

### Previsualización de los Desafíos Online con la clase

Muestre un desafío a la clase. Recomendamos el tercero. Construya el código de la manera larga primero (use números exactos para cada valor, en lugar de utilizar variables) y luego sugiera que debería intentar hacer los cuadrados de sólo 50 píxeles. ¡Qué terrible! ¿Qué han aprendido los/as estudiantes que les permita darle un nombre a algo y usarlo tantas veces como quieran más adelante en el programa? Retroceda y añada una variable al principio. Poner la variable en 80, y sustituir todas las ocurrencias de 80 en el programa. Luego, cámbielo a 50. ¡Eso fue fácil!

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Note que esta etapa cubre primero la idea de una variable como una constante (una variable que se usa en muchos lugares, pero que no cambia). Una vez que se ha presentado esa idea, se voltea para mostrar cómo se puede incluir una variable para la información que cambia después de ejecutar el programa.

Cuidado con el desafío número 5. Es la primera vez que se espera que los/as estudiantes establezcan una variable por sí mismos/as. Esto puede ser difícil si no tienen un verdadero conocimiento del concepto. Si tienen problemas, envíelos de vuelta al nivel de predicción (nº 4) y pídeles que expliquen a sus parejas por qué la respuesta terminó

siendo la misma. Una vez que ambos miembros de la pareja estén convencidos, permítales continuar con el desafío número 5.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué es una variable? ¿Por qué es útil en la programación?
- ¿Qué tan bien crees que entiendes las variables? (Responde en una escala del 1 al 5 o con un emoticón.) Si tienes problemas, ¿puedes poner en palabras lo que no entiendes?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.



# Lección 17: Variables en Play Lab

Variable | Laboratorio de juegos

## Descripción

Los/as estudiantes podrán seguir practicando con las variables de esta lección creando escenas en Play Lab. Los/as estudiantes trabajarán con la entrada del usuario para establecer los valores de sus variables, y luego tendrán espacio para crear su propio mini-proyecto con variables.

## Propósito

Esta lección permite a los/as estudiantes utilizar variables para mostrar frases o conversaciones basadas en lo indicado por el usuario. Esta lección sirve como un maravilloso ejercicio de práctica de variables en la programación, ¡con una dosis extra de creatividad! Al final de la secuencia del desafío, los/as estudiantes tendrán la oportunidad de compartir sus proyectos con familiares y amigos/as.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Principal (30 min)

[csf-course-f][2]

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Usar variables para contener palabras y frases.
- Usar variables junto con las sugerencias de caracteres.

## Preparación

- ☐ Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Diario Think Spot - Diario de reflexiones** [Hacer una copia](#)

## Vocabulario

- **Variable** - un marcador de posición para una información que puede cambiar.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Pida a los/as estudiantes que levanten la mano y digan cuál es su comida favorita. Elija un par de estudiantes para que respondan. En una pizarra o cartulina, escriba:

A (nombre estudiante) le gusta (comida favorita)

Ejemplo: A Kiki le gusta la pasta.

Después de un par de estudiantes, pregunte a la clase si usted podría usar variables para continuar con esto el resto de la clase. Pregúnteles donde podría ir las variables y como las podrían llamar. Una vez que la discusión haya terminado, escriba:

[name] le gusta [food]

en la misma muestra

Pregunte a los/as estudiantes cuál es variable y cuál no. ¿Cómo lo saben? ¿Qué más podrían usar en oraciones como esta? (ejemplos: color favorito, ciudad, número de hermanos, etc.)

## Actividad Principal (30 mín)

### Curso F Desafíos Online - 2018 - Sitio web

Recuerde a sus estudiantes compartir su trabajo sólo con sus amigos cercanos y familiares. Para mayor información, vea o muestre a la clase [common-sense-media-pause-think][1].

### Curso F Desafíos Online - 2018 - Sitio web

[csf-course-f][2]

Sus estudiantes ya han conocido las variables, pero si piensa que están complicados con el concepto, recuérdelos que le pregunten a sus compañeros/as antes de acercarse a el/la profesor/a para que le ayude. Esto estimula la discusión y motiva una comunidad de aprendizaje.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil, y cómo se sienten. Puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Cuáles son algunas formas en las que has usados las variables hasta ahora?
- ¿Qué más crees que puedes hacer con las variables?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 18: Loops Para: Diversión con Loops Para

Sin conexión | Bucles desde

## Descripción

Sabemos que los loops nos permiten hacer cosas de forma repetida, pero ahora vamos a aprender cómo usar loops que tienen estructuras extras construidas en su interior. Estas estructuras nuevas permitirán a los/as estudiantes crear códigos que son más poderosos y dinámicos.

## Propósito

En este punto, los/as estudiantes son expertos/as en loops. Hoy, aprenderán acerca de otro loop usado comúnmente en programación. El Loop Para repite los comandos un cierto número de veces, pero también llevan registro de los valores que están repitiendo. Por ejemplo, un Loop Para que comienza en 4, termina en 8 y tiene un valor de paso de 1, repetirá 4 veces, pero los valores 4, 5, 6 y 7 también serán guardados para usarlos en otros lugares. Usando esta estructura con variables, se pueden crear programas fantásticos. Hoy, los/as estudiantes aprenderán los elementos básicos de los Loops Para, ¡antes de sumergirse en la programación con ellos!

## Agenda

### Actividad previa (20 min)

Para Uno y Para Todos

### Actividad Principal (20 min)

Loops Para: Diversión con Loops Para - Ficha

### Actividad de Cierre (15 min)

Conversación Rápida: ¿Qué aprendimos?

Escribir en el Diario

### Evaluación (5 min)

Loops Para: Diversión con Loops Para - Evaluación

### Aprendizaje Extendido

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Determine el valor inicial, el valor final y el valor de paso para un bucle `desde`.
- Ilustrar los valores del contador que se alcanzan cada vez a través de un bucle desde durante el tiempo de ejecución.

## Preparación

- Vea [teacher-video-for-loop-fun-2][0].
- Vea [lesson-in-action-for-loop-fun][1].
- Imprima una [csf-for-loop-fun-worksheet][2] por grupo.
  - Imprima una [csf-for-loop-fun-assessment][3] para cada estudiante.
  - Asegúrese de que cada estudiante tenga un [think-spot-journal][4].

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Loops Para: Diversión con Loops Para - Video sin conexión (descargar)**
- **Diversión con Loop Para - Clave de Evaluación - Guía para el docente**
- **Loops Para: Diversión con Loops Para - Video de lección en acción**
- **Diversión con Loop Para - Guía de respuestas en fichas**

Para los estudiantes

- **Loops Para: Diversión con Loops Para - Ficha**
- **Loops Para: Diversión con Loops Para - Evaluación**
- **Diario Think Spot - Diario de reflexiones**

# Vocabulario

- **Bucles desde** - bucles que tienen un comienzo, un final y un incremento predeterminados (intervalo de paso).

# Guía docente

## Actividad previa (20 min)

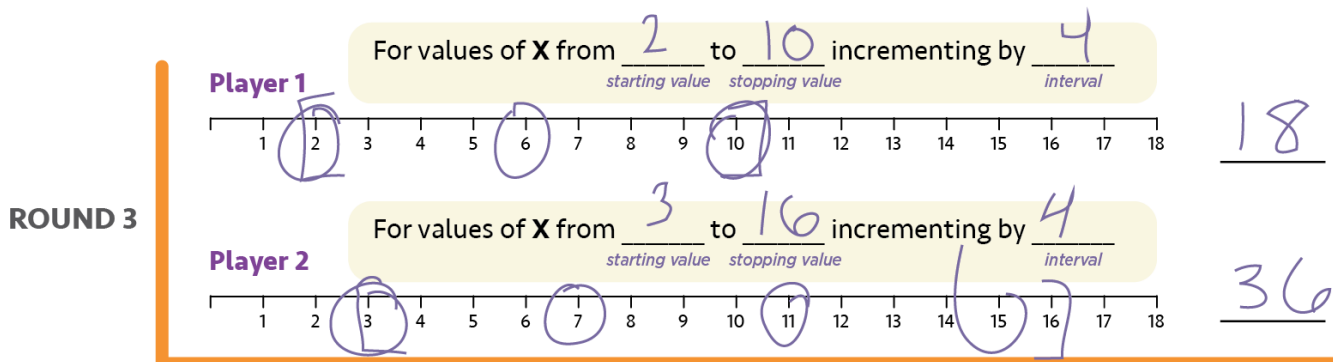
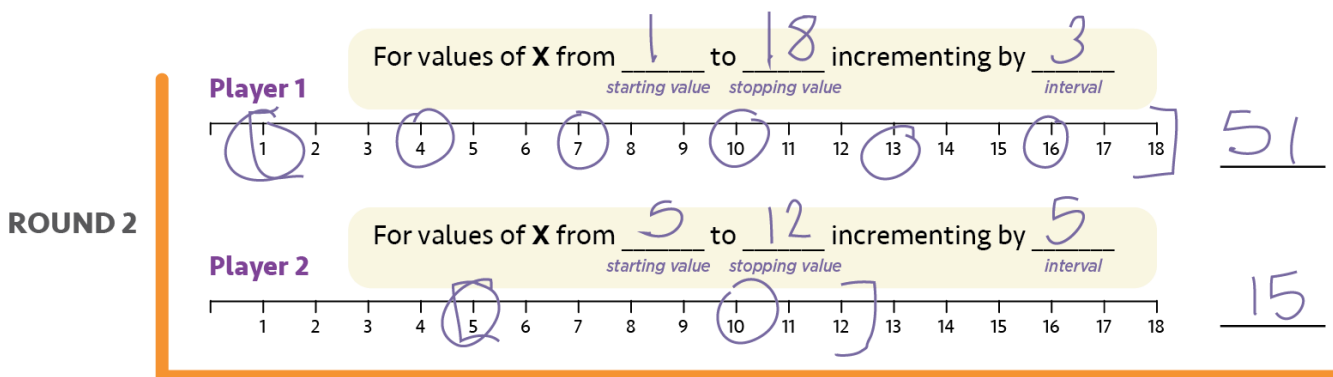
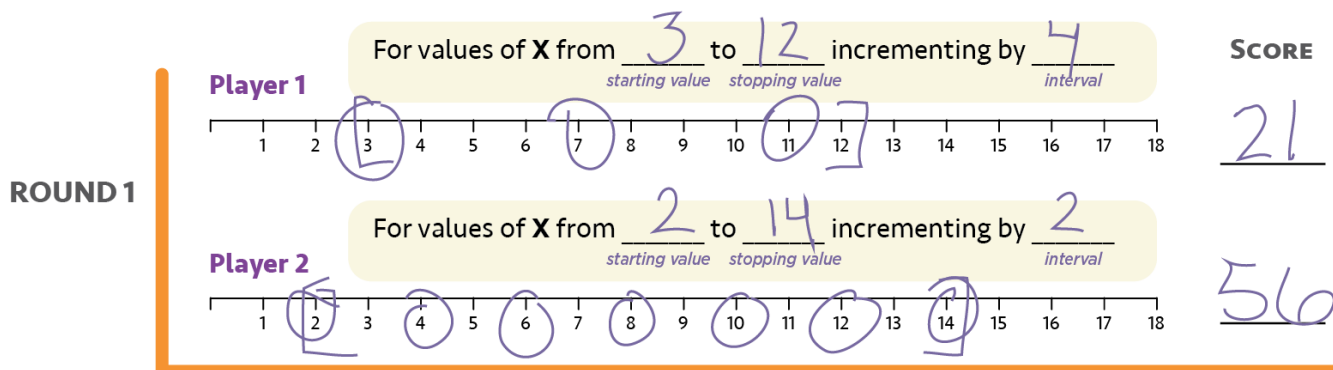
- **Loop Para** - Díganlo conmigo: Lup pa-ra

Son loops que tienen un comienzo, un final y un incremento (intervalo en pasos) predeterminado.

### Para Uno y Para Todos

- Señale que hay ciertos loops que son muy frecuentes, por ejemplo, los loops donde necesita llevar un registro de cuántas veces se ha usado
  - A veces, no quieres comenzar en uno.
  - A veces, no quieres contar de a uno.
  - Los loops "Para" te dan una forma poderosa de tener un contador que comienza cuando quieres, termina cuando quieres y se incrementa en cualquier tamaño que tú quieras.

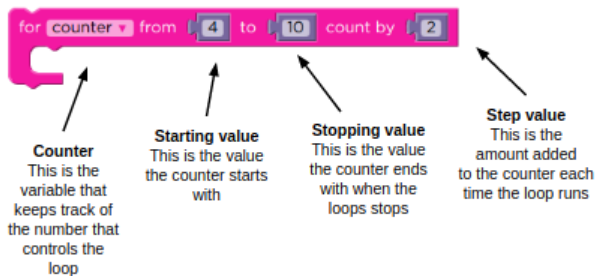
Aquí, puedes saltar a un ejemplo del juego



# Actividad Principal (20 min)

## Loops Para: Diversión con Loops Para - Ficha

A veces queremos repetir las cosas un cierto número de veces, pero queremos llevar un registro de los valores. Aquí es donde entran los Loops Para. Cuando usamos un "Loop Para", sabes desde el principio cuál es el valor de inicio, cuál es el valor de detención (de término) y cuánto cambia el valor cada vez que se avanza.



Bloque Loop "Para"

### Instrucciones:

Puede ser difícil para estudiantes pequeños comprender este pseudocódigo escrito, pero puede ser útil que lo explique en voz alta (y quizás con un diagrama) lo que estarán usando como contenido del Loop "Para".

- Divida a los/as estudiantes en parejas.
- Para comenzar una ronda, cada estudiante lanza el dado tres veces:
  - Un dado determina el valor de inicio para X
  - Tres dados determina el valor de detención para X
  - Un dado determina el intervalo de X cada vez que avanza
- Use una de las líneas numéricas previstas para trazar el Loop "Para" que han hecho
  - Comience en el valor de inicio X
  - Cuente el número de líneas, haciendo un círculo en el número del intervalo señalado por los dados
  - Deténgase cuando llegue al valor de detención predeterminado
- Agregue a su puntaje, todos los valores en círculos, luego el otro jugador tiene su turno.
- Gana el mejor de dos jugadas de tres.

### Sugerencia para la lección

Cuando haga este juego, es como si ejecutara un loop como este:

```
para (x=valorInicio; x <= valorDetención; x = x + intervalo){  
  círculo valorXactual;  
  sumar valorXactual al puntajeRedondeado  
}
```

### Sugerencia para la lección

Si cualquiera de los valores obtenidos está fuera de los márgenes del juego (como lanzar un valor inicial de 6 pero luego obtener 2, 1, 2 para el valor final), haga que el/la estudiante vuelva a lanzar todo de nuevo.

## Actividad de Cierre (15 mín)

Conversación Rápida: ¿Qué aprendimos?

- ¿Qué intervalo necesitarías para contar desde 4 hasta 13 de tres en tres?
- ¿Qué tipos de cosas se te ocurren que podrías hacer con un Loop Para?
- ¿Puedes reproducir un loop normal usando un Loop Para?
- ¿Qué necesitarías hacer?

### Sugerencia para la lección

Las preguntas de la conversación Rápida tienen el objetivo de ver la imagen completa y como se relaciona con el mundo real del/a estudiante y su futuro. Use el conocimiento que tiene de su curso para decidir si hace esto como clase, en grupos o con el/a compañero/a de puesto.

Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

**Sugerencias para el Diario:**

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué es un Loop "Para"?
- ¿Para qué usarías un Loop "Para" en lugar de un Loop "Repetir" o un Loop "Mientras"?

## Evaluación (5 min)

### Loops Para: Diversión con Loops Para - Evaluación

Distribuya la hoja de la evaluación y permita que los/as estudiantes completen la actividad en forma independiente luego que las instrucciones queden claras. Esto debería serles familiar debido a las actividades en lecciones previas.

## Aprendizaje Extendido

Use estas actividades para mejorar el aprendizaje de sus estudiantes. Pueden usarse como actividades fuera del aula u otros enriquecimientos.

### Ejecutarlo al Revés

- Intente esta actividad nuevamente, pero esta vez pida a los/as estudiantes comenzar con un número seleccionado usando 3 dados, y detenerse en un número seleccionado por un dado. ¡Asegúrese de tener un incremento negativo!

### El avión (o el luce)

- Usando tiza, dibuje un avión (o luce) en el patio
  - Enumere los cuadrados desde el inferior al superior.
  - Pida a los/as estudiantes darse un cuadrado de inicio mutuamente, un cuadrado de detención, y cuántos necesitan saltar cada vez.
  - Cuando el participante termine, pídale que escriba el loop que han hecho.
  - Comience a agregar actividades adicionales para hacer en cada cuadrado, esto también le agregará complejidad a la parte escrita

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.



# Lección 19: Loops Para en Abeja

Bucles desde | Abeja

## Descripción

Con Abeja, esta lección se centra en los bucles "para" y utiliza una variable gradual para resolver desafíos más complicados. Los/as estudiantes comenzarán repasando los loops de las lecciones anteriores. Luego, repasarán una introducción a los loops "para", para que puedan resolver problemas complicados de manera más efectiva.

## Propósito

El concepto actual, bucles para, es un tema muy importante en la informática. No sólo son ampliamente utilizados, el proceso de aprendizaje para los loops para mejora el aprendizaje de otros conceptos importantes (tales como las variables y parámetros). Los/as estudiantes tendrán mucha práctica en el pensamiento crítico de los problemas al determinar los valores de inicio, final y pasos para cada loop para. Este concepto también utiliza mucho las matemáticas, así que siéntase libre de combinarlo con una lección de matemáticas para una experiencia de aprendizaje aún más profunda.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Puente - Loops Para (15 mín)

Actividad Sin Conexión Usando Bloques de Papel  
Pre visualización de los Desafíos en Línea con la Clase

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Determine el valor inicial, el valor final y el valor de paso para un bucle `desde`.
- Reconocer cuándo usar un bucle `desde` y cuándo usar otros bucles, como `repetir` y `mientras`.

## Preparación

- ☐ Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores**
- **Diario Think Spot - Diario de reflexiones** [Hacer una copia](#)

## Vocabulario

- **Bucles desde** - bucles que tienen un comienzo, un final y un incremento predeterminados (intervalo de paso).

# Guía docente

## Actividad previa (15 mín)

### Introducción

Recuerde a los/as estudiantes el trabajo que hicieron en "Loop Para Divertido". Abra una discusión sobre lo que aprendieron, por qué creen que podría ser útil y si se divertieron. Aquí hay algunas ideas para empezar la discusión.

- ¿Qué aprendiste en " Loop Para Divertido "?
- ¿Cuáles son los tres componentes principales de un Loop?
  - Valor inicial, intervalo de paso, valor final.
- ¿Por qué crees que un loop podría ser útil en la programación?
  - Es posible que muchos/as estudiantes no sepan la respuesta a esto. Deje que formulen hipótesis, pero no se detenga en esta pregunta por mucho tiempo.
- ¿Te divertiste aprendiendo sobre los loops? ¿Por qué si, o por qué no?
- ¿Estás emocionado/a por utilizar loops en desafíos en línea?

## Actividad Puente - Loops Para (15 mín)

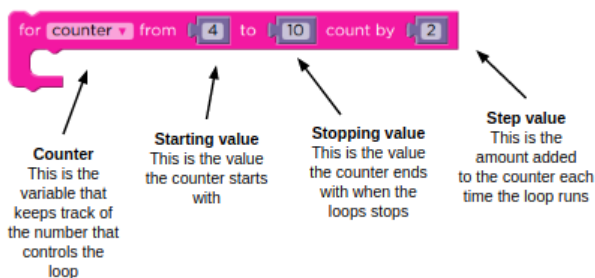
Esta actividad presentará los conceptos sin conexión de "Loop Para Divertido", al mundo en línea al que los/as estudiantes se están moviendo. Elija *una* de las siguientes actividades para realizar con su clase:

### Actividad Sin Conexión Usando Bloques de Papel

Divida a su clase en parejas. Idealmente forme las mismas parejas de la clase "Loop Para Divertido". Imprima un "Loop Para" de **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores** para cada pareja de estudiantes. Entregue uno a cada pareja. Haga que las parejas se alternen para tirar el dado para obtener los siguientes valores:

- Una tirada para el valor inicial.
- Cuatro tiradas para el valor final.
- Una tirada para el intervalo de paso.

Haga que cada pareja complete el "Loop Para" con el valor adecuado, en el espacio correcto.



Al usar una línea numérica básica, como la utilizada en "Loop Para Divertido", haga que los/as estudiantes marquen el inicio, final y valores intermedios que tocará el "Loop Para". Cuando todos hayan terminado, vea quién obtuvo la mayoría de los puntos sumando los números iniciales, medios y finales de cada pareja.

### Pre visualización de los Desafíos en Línea con la Clase

Muestre un desafío en línea de **Curso F Desafíos Online - 2018 - Sitio web** asociado a esta lección. Recomendamos el número nº 4, porque muestra la potencial solución y pide al/a estudiante que la evalúe.

Usando una línea numérica similar a las usadas en "Loop Para Divertido", marque los valores iniciales y finales del loop (si no está usando el desafío 4, necesitará encontrar una potencial solución primero). Con la ayuda de la clase, circule los valores entre el inicio y el final que tocará el "Loop Para". Si está trabajando en el desafío 4, pregunte a la clase cuál piensan que es la respuesta a la pregunta, dado lo que encontraron con la línea numérica.

# Actividad Principal (30 min)

## Curso F Desafíos Online - 2018 - Sitio web

Es posible que algunos/as estudiantes tengan dificultad diferenciando entre los “Loops Repetir” y los “Loops Para”. Le recomendamos tener un papel a mano para que los/as estudiantes hagan sus predicciones en relación a los valores de inicio, final y paso. También puede ser beneficioso para su clase implementar la programación en parejas.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿En qué se diferencia un bucle para de un bucle repetir?
- ¿Por qué crees que los bucle para podrían ser útiles?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 20: Loops Para en Artista

Bucles desde | Artista

## Descripción

En esta lección, los/as estudiantes continuarán practicando con los loop "para", pero esta vez con Artista. Los/as estudiantes completarán desafíos combinando las ideas de variables, loops y loops "para", para crear diseños complejos. Al final, tendrán la oportunidad de crear su propio arte en un nivel de juego libre.

## Propósito

La creatividad y el pensamiento crítico se conjugan a la perfección en esta lección. Los/as estudiantes continuarán su práctica con loops y variables mientras crean imágenes que los dejarán con la boca abierta. Esta lección inspira una mente creativa a la vez que enseña conceptos básicos de informática.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Usar bucles `desde` para cambiar el bucle varias veces con diferentes valores.
- Reconocer cuándo usar un bucle `desde` y cuándo usar otros bucles, como `repetir` y `mientras`.

## Preparación

- ▣ Recorra **Curso F Desafíos Online - 2018 - Sitio web** asociados a esta lección para encontrar potenciales áreas problemáticas para su clase.
- ▣ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ▣ Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018** - Sitio web
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación** - Recomendaciones de lecciones [Hacer una copia ▾](#)

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones [Hacer una copia ▾](#)

## Vocabulario

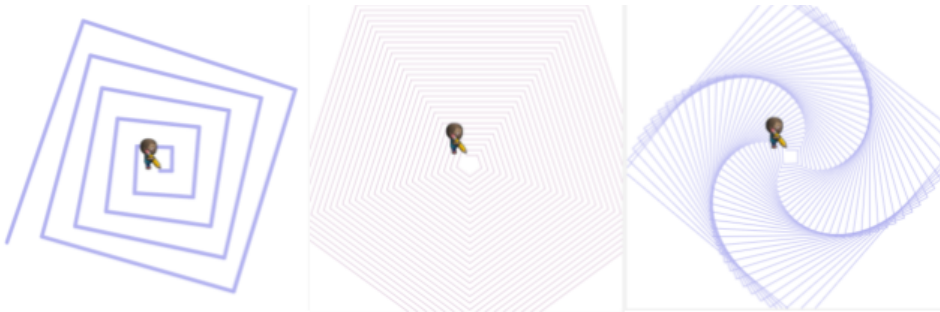
- **Bucles desde** - bucles que tienen un comienzo, un final y un incremento predeterminados (intervalo de paso).

# Guía docente

## Actividad previa (15 mín)

### Introducción

Dibuje en la pizarra para que toda la clase pueda ver (o utilizando un proyector) uno de los proyectos finales de **Curso F Desafíos Online - 2018 - Sitio web** asociado a esta lección. Recomendamos uno de los siguientes:



Pregunte a la clase como podría un computador hacer el dibujo que usted acaba de mostrar.

Luego de algunas cuantas respuestas, diga: ¡con los loops "Para", por supuesto!

Dígale a sus estudiantes que pronto aprenderán como crear estos geniales dibujos usando variables y loops "para".

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Estos desafíos son muy divertidos, pero puede ser útil para los/as estudiantes tener transportadores y papeles en blanco para ver estos diseños hechos en un formato físico. Si esto no es una opción en su clase, intente que los/as estudiantes tracen la pantalla del computador con sus dedos.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten, puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- Dibuja uno de los diseños que hiciste hoy. ¿Cuál era el código necesario para crearlo?
- ¿Cuáles son algunos de los diseños que te gustaría crear? ¿Cómo crees que los loops "para" o las variables podrían ayudar a crearlos?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a  
Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 21: Funciones: Componer Canciones con Parámetros

Sin conexión | Función | Parámetro

## Descripción

Una de las estructuras más importantes en el mundo de la informática es la función. Las funciones (a veces llamadas procedimientos) son pequeños programas que usted puede usar una y otra vez dentro de un programa más grande. Esta lección ayudará a los estudiantes a comprender intuitivamente por qué combinar bloques de código en funciones, es una práctica tan útil, y cómo pueden usar esas estructuras incluso cuando los bloques de código son ligeramente diferentes.

## Propósito

El uso de las funciones ayuda a simplificar el código y desarrolla la capacidad del estudiante para organizar su programa. Los parámetros ayudarán a los/as estudiantes a personalizar sus funciones para que puedan ser utilizados para patrones que son similares, pero no idénticos. Los/las estudiantes reconocerán rápidamente que las funciones de escritura harán que sus programas largos sean más fáciles de leer y más fáciles de depurar, si algo sale mal.

## Agenda

### Actividad previa (15 mín)

Vocabulario

Cante una canción

### Actividad Principal (20 min)

Funciones Sin conexión: Componer Canciones con Parámetros - Ficha

### Actividad de Cierre (15 mín)

Conversación Rápida: ¿Qué aprendimos?

Escribir en el Diario

### Evaluación (5 min)

Funciones Sin conexión: Componer Canciones con Parámetros - Evaluación

### Aprendizaje Extendido

## Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Modificar funciones para aceptar parámetros.
- Describir cómo las funciones y los parámetros pueden hacer que los programas sean más fáciles de escribir.

## Preparación

- ▢ Vea el [teacher-video-songwriting-functions][0].
- ▢ Vea el [lesson-in-action-songwriting-parameters][1].
- ▢ Imprima varios [csf-songwriting-parameter-worksheet][2] para cada grupo.
- ▢ Imprima una [csf-songwriting-parameter-assessment][3] para cada estudiante.
- ▢ Entre a internet, o descargue previamente canciones y letras para esta actividad.
- ▢ Asegúrese de que cada estudiante tenga un [think-spot-journal][4].

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Componer Canciones con Parámetros** - Video sin conexión (**descargar**)
- **Funciones Sin conexión: Componer Canciones con Parámetros** - Video de lección en acción
- **Funciones Sin conexión: Componer Canciones con Parámetros** - Ficha
- **Funciones Sin Conexión: Componer Canciones con Parámetros** - Guía de respuestas en fichas
- **Funciones Sin conexión: Componer Canciones con Parámetros** - Evaluación

- **Funciones Sin Conexión: Componer Canciones con Parámetros** - Guía de respuestas de la evaluación

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones

Hacer una copia ▾

## Vocabulario

- **Función** - un grupo de instrucciones de programación. Las funciones son abstracciones que se pueden volver a utilizar para reducir la complejidad a la hora de escribir y mantener programas.
- **Parámetro** - información adicional que se transfiere a una función para personalizarla para una necesidad específica.



# Guía docente

## Actividad previa (15 mín)

### Vocabulario

Esta lección tiene dos nuevas e importantes palabras:

- **Función** - Díganlo conmigo: Fun - ción.

Un bloque de código que se puede llamar fácilmente una y otra vez.

- **Parámetro** - Dilo conmigo: Pa - rá - me - tro.

Un bloque extra de información pasado a una función con el objeto de personalizarlo para una necesidad específica

### Cante una canción

- ¡Cuénteles a la clase que hoy es día de canción! Vamos a aprender una canción juntos/as.
  - Empiece con una canción fácil, escrita o proyectada en la pantalla.
  - Señale el estribillo y asegúrese de que la clase lo conozca antes de continuar con el resto de la canción.
  - Recorra rápidamente la canción, cantando con ellos en el comienzo y luego vea qué sucede cuando llegue a la parte que llama al estribillo.

Nota: La canción a continuación es una sugerencia. Usted puede elegir la canción que considere pertinente siempre y cuando esté alineada con los objetivos de la lección.



#### \*Chorus:

Little bunny Foo Foo  
Hopping through the forest  
Scooping up the field mice  
And bopping 'em on the head  
Down came the Fairy  
And she said  
"Little bunny Foo Foo  
I don't wanna see you  
Scooping up the field mice  
And bopping 'em on the head"\*

#### \*Song:

Chorus\*

*I'll give you 3 chances.  
Then I'll turn you into a goon!  
The next day. . .*

*Chorus*

*I'll give you 2 chances.  
Then I'll turn you into a goon!  
The next day. . .*

*Chorus*

*I'll give you 1 chance.  
Then I'll turn you into a goon!  
The next day. . .*

*Chorus*

#### 💡 Teaching Tip

Little Bunny Foo Foo is being used here as an example only. If your students know this song, feel free to use it. Otherwise, choose an appropriate song that they might be more familiar with (either from music class or the radio.)

*"I gave you two chances.  
Now I'll turn you into a goon!"  
(POOF!)  
And the moral of the story is:  
Hare today, goon tomorrow!*

- Es muy probable que la mayoría de la clase cante la letra del estribillo cuando señale esa parte.
  - Detenga la canción una vez que eso suceda, y destaque explícitamente lo que acaba de suceder.
    - Usted definió el estribillo.
    - Usted llamó el estribillo.
    - Ellos cantaron el estribillo.
- Pregúntele a la clase por qué suponen que usted sólo escribió el estribillo una vez en la parte superior del papel en lugar de escribirlo una y otra vez en cada lugar donde se supone que debe ser cantado.
  - ¿Cuáles son los otros beneficios de escribir el estribillo una sola vez cuando lo cantas muchas veces?

Ahora, imagine que esta canción es un programa de computador. Definir un título (como "estribillo") para un pequeño bloque de código que se usa una y otra vez, se llama crear una función. Esto es útil para los informáticos, por algunas de las mismas razones por las que es útil para los compositores.

- Ahorra tiempo al no tener que escribir todo el código una y otra vez en el programa.
- Si se equivoca, sólo debe cambiarlo en un lugar.
- El programa se siente menos complicado con las piezas que se repiten sólo una vez en la parte superior.

**¿Qué pasa con las canciones cuyo estribillo no es siempre exactamente igual? Puede igualmente usar un estribillo, pero debe tener una forma de decirle al cantante cuales son las palabras especiales que usará para cada verso.**

- Estas palabras especiales se llaman parámetros.
- En programación, los parámetros son instrucciones especiales para funciones como esta:

```
estribillo(parámetro1, parámetro2)
```

¿Siente que se está poniendo complicado? No se preocupe. Vamos a jugar un poco más con canciones para comprender de verdad como esta técnica es utilizada.

## Actividad Principal (20 min)

### Funciones Sin conexión: Componer Canciones con Parámetros - Ficha

Una manera fantástica de comparar las funciones con algo que vemos en nuestra vida diaria es mirar las canciones. Las canciones a menudo tienen ciertos grupos de letras que se repiten una y otra vez. A eso lo llamamos estribillo.

#### Instrucciones:

1. Dividir en grupos de 4, 5 o 6.
2. Entregue a cada grupo varias copias de la Guía para Componer Canciones.
3. Toque una breve canción para la clase que contenga un estribillo claro que no cambie de verso en verso.
4. Desafíe a la clase a identificar (y escribir) el estribillo.
5. Compare los resultados de cada grupo. ¿Todos escribieron lo mismo?
6. Intente la actividad nuevamente, pero esta vez con una canción que cambie durante cada repetición del estribillo. Algunos ejemplos buenos son: Old Mcdonald, Baby Bublebee o The Hokey Pokey.

Discuta con la clase:

- ¿Pueden los estudiantes identificar un estribillo cuando algunas palabras cambian?

#### Recomendación para la lección

Para aumentar el interés, busque una canción popular en internet. Muéstrela a los/las estudiantes que el estándar para repetir la letra es lo que define el coro en la parte de arriba y lo llama desde el cuerpo de la canción.

- ¿Cómo podrían usar la misma idea de cantar un estribillo cuando los versos del estribillo son diferentes? éste es diferente de las estrofas
- Estas palabras y frases que cambian se llaman "parámetros" y se pueden pasar al estribillo de la siguiente manera: estribillo (cow, moo)
- Juegue este juego una y otra vez hasta que la clase tenga pocos problemas para identificar los estribillos.

A menudo es más fácil hacer que la clase escuche (o vea) la canción, y luego votar acerca de cuál es el estribillo cantando juntos, en lugar de escribirlo todo. Si elige este método, considere la posibilidad de que la clase haga un estribillo escrito para la selección final de canciones, para asegurarse de que los/as estudiantes visuales reciban el refuerzo adecuado.

#### 💡 Lesson Tip

Es más emocionante para los/as estudiantes hacer esta clase con música popular de la radio, pero si le está costando encontrar canciones apropiadas donde las letras se repitan de forma exacta, aquí hay algunas alternativas.

Nota: Las canciones Old McDonald, Baby Bumblebee o The Hokey Pokey son una sugerencia. Usted puede elegir la canción que considere pertinente, siempre y cuando esté alineada con los objetivos de la lección.

- **5 Little Monkeys**
- **Old MacDonald**
- **Hokey Pokey**
- **BINGO**
- **Baby Bumble Bee**

## Actividad de Cierre (15 mín)

### Conversación Rápida: ¿Qué aprendimos?

- ¿Preferirías escribir la letra de una canción muchas veces o definir un estribillo?
- ¿Crees que sea posible hacer varios estribillos para la misma canción?
- ¿Tiene sentido hacer un estribillo nuevo, cada vez que se necesita en una canción?

#### 💡 Lesson Tip

Las preguntas de la Charla Exprés buscan ver la foto completa para ayudar a pensar cómo la lección se relaciona con el mundo y el futuro de los/as estudiantes. Use su conocimiento de los/as estudiantes para decidir si quiere discutir esto como clase, en grupos o con el/a compañero/ de puesto.

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten. Puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy, y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Cómo ves que las funciones son útiles en la ciencia de la computación?
- Describe por qué los parámetros son útiles cuando se escribe la letra de una canción en la que el estribillo cambia ligeramente.

## Evaluación (5 min)

### Funciones Sin conexión: Componer Canciones con Parámetros - Evaluación

Distribuya la hoja de la evaluación y permita que los/as estudiantes completen la actividad en forma independiente luego que las instrucciones queden claras. Esto debería serles familiar debido a las actividades en lecciones previas.

## Aprendizaje Extendido

Utilice las siguientes actividades para mejorar el aprendizaje de los/las estudiantes. Pueden ser utilizadas como actividades fuera de la clase u otro fortalecimiento.

**Atrapa sueños Funcionales** Visite la **Tabla Fundamentos de Ciencias de la Computación Sin Conexión** o haga clic en el link para **Atrapa Sueños Funcionales**. Esta actividad usa algunos artículos extra pero ayuda a los/as estudiantes a ver el valor de usar múltiples funciones.

### Crear su propia canción

- Comenzar creando un estribillo juntos/as, luego repetirlo entre los versos de la canción que desarrolla.
- Haga un cambio en el estribillo, y piense en lo fácil que es realizar un cambio en un solo lugar.
- Cambie el estribillo de nuevo, haciéndolo mucho más largo de lo que era originalmente.
- Añada un segundo estribillo y alterne entre los versos.
- Agregue parámetros a uno de sus estribillos y vea cuántas opciones más tiene.

\*Componer un programa\*\*

- ¿Y si actuáramos las canciones en vez de cantarlas? De repente, nuestro estribillo sería una función de acciones repetidas, más que de palabras.
- Utilice los conceptos de las flechas de la lección de Programación de Papel Gráfico y cree un programa con muchas instrucciones repetidas.
  - Marque con un círculo las acciones que se repiten para que la clase pueda ver dónde están.
  - Defina una función que se llame “estribillo” en la parte de arriba de programa.
  - Tache en todos lados donde aparezcan las acciones repetidas en el programa y escriba “estribillo” en su lugar.
- Repita hasta que la clase pueda realizar este proceso sin ayuda.
- ¿Puede descifrar cómo pasar los parámetros en este ejercicio?

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 22: Funciones en Abeja

Función | Abeja

## Descripción

Esta lección enseña a los/as estudiantes cómo crear funciones simples usando nuestro sofisticado editor de funciones "modales", preparándoles el camino para que puedan incorporar parámetros en futuras lecciones.

## Propósito

Los/as estudiantes descubrirán la versatilidad de programar practicando las funciones en diferentes ambientes. Aquí, reconocerán patrones en el laberinto de la abeja. La abeja necesitará navegar el área de juego, recoger néctar y hacer miel. Aprenderán a organizar sus programas y a crear funciones para códigos repetidos.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Puente - Funciones (15 mín)

Actividades Sin Conexión Usando Bloques de Papel  
Previsualización de los Desafíos Online con la Clase

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el diario

### Aprendizaje Extendido

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Categorizar y generalizar código en funciones útiles.
- Reconocer cuándo una función podría ayudar a simplificar un programa.

## Preparación

- ☐ Recorra los **Curso F Desafíos Online - 2018 - Sitio web** para encontrar áreas de problemas potenciales para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia](#)

Para los estudiantes

- **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores**
- **Diario Think Spot - Diario de reflexiones** [Hacer una copia](#)

## Vocabulario

- **Función** - un grupo de instrucciones de programación. Las funciones son abstracciones que se pueden volver a utilizar para reducir la complejidad a la hora de escribir y mantener programas.

- **Parámetro** - información adicional que se transfiere a una función para personalizarla para una necesidad específica.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Los/as estudiantes que han hecho **Curso E Desafíos Online - Sitio web**, ya tienen experiencia en el uso de funciones para resolver desafíos en línea, pero el editor de funciones en este curso es ligeramente diferente. Informe a los/as estudiantes que realizarán una revisión rápida de las funciones simples antes de pasar a desafíos más difíciles con el nuevo editor "modal".

Para los/as estudiantes que están menos familiarizados con el uso de funciones en línea, empiece repasando las palabras del vocabulario de las "Funciones Sin Conexión": Composición de canciones con parámetros".

- **Función** - Díganlo conmigo: Fun-ción.

Un trozo de código al que puedes llamar una y otra vez.

- **Parámetro** - Díganlo conmigo: Pa-rá-me-tro.

Un pedazo extra de información que usted pasa a la función para personalizarla para una necesidad específica.

Dígale a la clase que el uso de funciones con parámetros tiene dos componentes principales.

1. **La Declaración:** Las declaraciones de función, son las que crean una función. En una declaración de función, se rellena la función con el código y se le da un nombre. Dentro de la declaración de función, se debe indicar dónde se utiliza el parámetro dentro del código de función. Debe declarar una función antes de poder utilizarla.
2. **El Llamado:** Las llamadas de función, son las que hacen que el programa ejecute el código de la función. Para llamar una función, coloque el nombre de la función en su programa con un valor para el parámetro. Asegúrese de que su función está correctamente definida (con un parámetro) antes de llamarla en su programa.

La clase puede usar la composición de canciones como ejemplo para entender estos dos componentes. En la actividad sin conexión, la función que contiene la letra del estribillo se llamaba "estribillo". Cuando hicimos esta función por primera vez, marcamos con un círculo la letra que iría en la función. Una vez que nombramos la función, pudimos leer las letras y reemplazar las letras repetidas del coro por una llamada de función a "estribillo".

Continúe la conversación hasta que los/as estudiantes tengan una comprensión básica de las funciones que se están explicando y llamando. Si los/as estudiantes no llegan a este punto, asegúrese de hacer una de las actividades de enlace antes de pasar a los desafíos de Code.org.

Nota: Los/as estudiantes no usarán parámetros en sus funciones hoy. Sin embargo, es bueno revisar para la próxima vez, qué son los parámetros y por qué se usan.

## Actividad Puente - Funciones (15 mín)

Esta actividad ayudará a ver los conceptos de "Funciones Sin Conexión: Composición de canciones con Parámetros" en el mundo en línea al que los/as estudiantes se están moviendo. Elija una de las siguientes opciones para su clase:

### Actividades Sin Conexión Usando Bloques de Papel

Escoja una canción que los/as estudiantes disfruten e imprima la letra. Puede utilizar la misma canción de "Funciones sin Conexión: Composición con Parámetros". Divida su clase en grupos o parejas. Reparta las letras impresas (incluyendo el estribillo repetido) y la función con bloques de parámetros de **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores** para cada grupo o pareja de estudiantes. Vea las recomendaciones para la lección, para más detalles.

Pida a los/as estudiantes que tachen cualquier parte de la canción que se pueda convertir en una función, incluso si tiene un par de palabras diferentes (el estribillo es un buen ejemplo) y que la pongan en los bloques de función proporcionados. Los/as estudiantes deben rellenar la declaración de función con un nombre de función y las palabras de las letras repetidas. Una vez hecha la declaración de función, pida que rellenen las llamadas de función y las coloquen encima de las letras tachadas.

Una vez que todos los grupos o parejas han terminado, pregunte a la clase donde ubicaron sus funciones y por qué. ¿Hicieron todos/as la misma función? ¿Qué tan a menudo se repite la función?

### 🔗 Sugerencia para la lección

Bloques de función:



El bloque a la izquierda es una función de declaración, un bloque que los/as estudiantes nombrarán y completarán en la función. El bloque a la derecha es la función llamada, un bloque que hace el código de función correr. Los/as estudiantes necesitarán múltiples bloques de la función llamada.

## Previsualización de los Desafíos Online con la Clase

Muestre un desafío de **Curso F Desafíos Online - 2018 - Sitio web**. Recomendamos el nº 12 para esta actividad. Con la clase, resuelva el desafío sin usar funciones. Una vez que haya obtenido la solución, muéstreala en una pizarra o en un proyector. Pida a la clase que señale el código repetido. Pregunte cómo simplificarían el programa. En la pizarra o en el proyector, reescriba el programa sin el código repetido, pero dejando una línea de espacio. En ese/esos espacio(s), llame una función. Declare la función como el bloque de ejemplo a la izquierda en la recomendación para la lección. Pregunte a la clase qué creen que hará el código ahora.

Inicie una discusión con la clase sobre por qué las funciones pueden ser útiles en la programación. Invite a los/as estudiantes a discutir la diferencia entre las funciones y los loops.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Los/as estudiantes podrían beneficiarse de escribir códigos sin funciones y luego crear funciones a partir de los códigos repetidos. Si los/as estudiantes no disfrutaban haciendo esto en el espacio de trabajo de Code.org, recomendamos proporcionar papel y lápices para que escriban (o dibujen) sus ideas.

## Actividad de Cierre (15 mín)

### Escribir en el diario

Hacer que los/as estudiantes escriban acerca de lo que han aprendido, por qué es útil, y cómo se sintieron puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy y a construir una hoja de revisión que pueden mirar en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Cómo ayudaron las funciones a la abeja a recolectar néctar y hacer miel de forma más eficiente?
- ¿Puedes imaginar usar parámetros en estos desafíos? Si es así, explica cómo. Si es no, ¿Por qué no?

## Aprendizaje Extendido

Dibujar con Funciones



Divida a la clase en grupos de 2-3 estudiantes. Haga que cada grupo escriba una función que dibuje algún tipo de forma y un programa que use esa función. Dependiendo de la creatividad o del enfoque de los grupos, los/as estudiantes pueden necesitar que se les asignen las formas a crear. Una vez que todos/as hayan terminado, pida a los grupos que intercambien programas. En una hoja distinta, cada grupo debe dibujar lo que crea el programa. Los grupos deben regresar los programas y los dibujos al grupo original.

¿Todos los grupos obtuvieron el dibujo que esperaban? Si no, ¿Qué salió mal? Haga que la clase realice el proceso de debugging y lo intenten de nuevo.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 23: Funciones con Parámetros en Artista

Función | Parámetro | Artista

## Descripción

En esta lección, los/as estudiantes continúan trabajando con funciones con y sin parámetros. Tendrán la oportunidad de crear sus propios dibujos, antes de modificar las funciones en un nivel de juego libre.

## Propósito

Esta lección ofrece un espacio para crear algo de lo que se sientan orgullosos/as.

Estos desafíos permiten a los/as estudiantes crear imágenes complejas a partir de proyectos anteriores más simples. Al final de esta lección, los/as estudiantes tendrán imágenes de las que sentirse orgullosos.

## Agenda

### Actividad previa (10 min)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 min)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Practicar la abstracción con el uso de parámetros.
- Reconocer cuándo una función podría ayudar a simplificar un programa.

## Preparación

- Juegue a través de **Curso D Desafíos Online - Sitio web** para encontrar áreas de potenciales problemas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018** - Sitio web
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación** - Recomendaciones de lecciones [Hacer una copia](#)

Para los estudiantes

- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#)

## Vocabulario

- **Función** - un grupo de instrucciones de programación. Las funciones son abstracciones que se pueden volver a utilizar para reducir la complejidad a la hora de escribir y mantener programas.

- **Parámetro** - información adicional que se transfiere a una función para personalizarla para una necesidad específica.

# Guía docente

## Actividad previa (10 min)

### Introducción

Pregunte a la clase si recuerdan haber usado los parámetros de "Funciones Sin Conexión": Composición de canciones con parámetros".

- ¿Qué significa "utilizar un parámetro" con funciones?
  - El uso de un parámetro significa que la función toma un valor de variable que puede especializar la función. En "Composición con Parámetros" esto significa que podemos cambiar las letras repetidas para que sean un poco diferentes cada vez.
- ¿Por qué usaríamos un parámetro con una función?
  - Utilizamos parámetros con funciones para no tener que escribir múltiples funciones que son muy similares. Si quisiéramos dibujar tres cuadrados con tres longitudes de lados diferentes, sólo tendríamos que escribir una función con un parámetro en lugar de tres funciones diferentes sin usar parámetros.
- ¿Por qué no usamos siempre parámetros con funciones?
  - No necesitamos SIEMPRE una función personalizable. A veces las funciones son sólo una forma práctica de reutilizar un código idéntico en múltiples lugares.

¡Dígale a la clase que harán dibujos increíbles en Artista, usando funciones con parámetros!

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

Pida a los/as estudiantes que cierren los ojos y levanten la mano. Si se sienten muy bien con el uso de los parámetros, pídale que pongan todos los dedos abiertos (como en "choca esos cinco"). Si no se sienten muy bien al usar los parámetros, pídale que levanten el puño. Si se sienten en algún punto intermedio, pídale que levanten uno, dos, tres o cuatro dedos de la mano.

Con eso, determine si su clase necesitará más práctica con las funciones antes de pasar a los desafíos en línea. Si sólo una pequeña parte de su clase no se siente bien con el uso de parámetros, asegúrese de implementar la programación en parejas en esta lección.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten. Puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy, y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- Haz un dibujo que hiciste hoy. ¿Puedes escribir el código necesario para crear esto?
- Dibuja lo que te gustaría crear con código. Intenta escribir o redactar el código que haría ese dibujo.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a  
Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 24: Funciones con Parámetros en Abeja

Función | Parámetro | Abeja

## Descripción

Esta lección presenta el entorno de Abeja, y continúa con el concepto de funciones con parámetros de la anterior etapa Artista. Los/as estudiantes practicarán la escritura y el uso de funciones para seguir caminos complejos y recolectar patrones de néctar y miel.

## Propósito

Las funciones son increíblemente importantes en las ciencias de la computación por muchas razones. La capacidad de desglosar y categorizar el código se volverá inmensamente importante a medida que los programas que sus estudiantes escriben se vuelven más y más complejos. Las funciones con parámetros requieren un nivel extra de habilidad. El uso de funciones con parámetros enseña a sus estudiantes a reconocer cuándo se necesita una función y si esa función puede generalizarse lo suficiente para ser utilizada en múltiples casos. Esta lección, junto con las anteriores sobre funciones con parámetros, construye un sólido conjunto de habilidades de pensamiento crítico y resolución de problemas.

## Agenda

### Actividad previa (15 mín)

Introducción

### Actividad Principal (30 min)

Curso F Desafíos Online - 2018 - Sitio web

### Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Reconocer las tareas repetidas que deben especializarse caso por caso y crear funciones para ejecutar estas tareas de manera eficiente.
- Usar funciones predefinidas con parámetros para completar tareas que se repiten con frecuencia.

## Preparación

- Recorra los **Curso F Desafíos Online - 2018 - Sitio web** asociados a este nivel para encontrar cualquier potencial área problemática para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Curso F Desafíos Online - 2018 - Sitio web**
- **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones** [Hacer una copia ▾](#)

Para los estudiantes

- **Diario Think Spot - Diario de reflexiones**

[Hacer una copia ▾](#)

## Vocabulario

- **Función** - un grupo de instrucciones de programación. Las funciones son abstracciones que se pueden volver a utilizar para reducir la complejidad a la hora de escribir y mantener programas.
- **Parámetro** - información adicional que se transfiere a una función para personalizarla para una necesidad específica.

# Guía docente

## Actividad previa (15 mín)

### Introducción

Antes de la clase, prepare un par de caminos en la sala de clases para que los/as estudiantes puedan caminar. Asegúrese de que el número de pasos sea obvio, ya sea usando cinta adhesiva o usando huellas de pies. La longitud de estos recorridos debe variar.

Reúna a la clase e indique que hay diferentes senderos para caminar, pero no quiere tener que escribir funciones separadas, para caminar por cada uno de ellos.

En cambio, en un lugar donde toda la clase pueda ver, escriba o muestre lo siguiente

**FUNCIÓN** - "trayectoria", **PARÁMETRO** - "paso"

- Repita la cantidad de "pasos":
  - Camine hacia adelante

Pregunte a la clase si saben lo que significa el código que usted escribió. Dígale a la clase que en lugar de escribir una función única para cada sendero, usted escribió una función que puede ser personalizada a la longitud de este.

Esto se hizo declarando una función, "trayectoria", y luego dándole un parámetro, "paso". La variable "paso" permite mantener el número de pasos de cada trayectoria.

Juegue con la función para cada sendero, teniendo a un/a voluntario/a que diga el número de pasos en un sendero y otro/a voluntario/a caminando por el sendero de acuerdo al código.

## Actividad Principal (30 min)

### Curso F Desafíos Online - 2018 - Sitio web

A medida que la clase trabaja con estos desafíos, camine alrededor preguntando lo siguiente a cada estudiante.

- ¿Estás usando una función? ¿Por qué? ¿Por qué no?
- Si no estás usando una función, ¿crees que una sería útil aquí?
- Si estás usando una función, ¿estás usando un parámetro? ¿Por qué? ¿Por qué no?
- Si no estás usando un parámetro, ¿crees que uno sería útil aquí? ¿Por qué? ¿Por qué no?

A veces los/as estudiantes no necesitarán una función o una función con parámetros, pero siempre deben saber por qué están haciendo lo que están haciendo.

## Actividad de Cierre (15 mín)

### Escribir en el Diario

Hacer que los/as estudiantes escriban acerca de lo que aprendieron, por qué es útil y cómo se sienten. Puede ayudar a solidificar cualquier conocimiento que hayan obtenido hoy, y a construir una hoja de revisión para que la revisen en el futuro.

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Crees que los parámetros son útiles en el código?
- ¿Cuándo usó un parámetro y cómo cambió la forma en que escribió el resto de tu programa?

## Normas de aprendizaje





This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 25: Explorar Ideas de Proyectos

Proyecto | Definir | Preparar | Probar | Revisión | Reflexionar

## Descripción

Las próximas cinco lecciones ofrecen a los/as estudiantes la oportunidad de poner en práctica sus habilidades de codificación en un proyecto final. Este proyecto ayudará a las personas a adquirir experiencia con la codificación y a producir un ejemplo para compartir con sus compañeros/as y seres queridos. Se trata de un proyecto para varias lecciones o varias semanas en las que los/as estudiantes dedicarán tiempo a la lluvia de ideas, al aprendizaje del proceso de diseño, a la construcción y a la presentación de su trabajo final.

En la etapa "Explorar", los/as estudiantes jugarán con programas de Artista y Play Lab ya construidos para inspirarse. A continuación, los/as estudiantes aprenderán sobre el proceso de diseño y cómo implementarlo en sus propios proyectos. Luego se les dará el espacio para crear su propio proyecto en Artista, Play Lab, o cualquier otra interfaz que usted se sienta cómodo proporcionando. (Esta es probablemente la etapa más larga del proyecto.) Los/as estudiantes revisarán su código después de las pruebas y la revisión de los compañeros. Finalmente, los/as estudiantes podrán presentar sus trabajos terminados a sus compañeros/as de clase.

## Propósito

Explorar ideas de proyecto está hecho para inspirar a los/as estudiantes con ideas realistas y entretenidas para sus proyectos finales.

## Agenda

### Día 1 - Explora Ideas Para Proyectos (45 min)

Ejemplos de Proyectos

### Día 2 - El Proceso de Diseño (45 min)

Definir y Preparar

### Día 3. Construye Tu Proyecto (45 min)

Intentar

### Día 4 (Recomendado para 5º Grado) - Revisa Tu Proyecto (45 min)

Reflexionar e Intentar nuevamente

### Día 5 & 6 - Presenta Tu Proyecto (45 min cada uno)

Presentaciones

### Actividad de Extensión

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Aprender a planificar con anticipación para una asignación continua.
- Poder explicar cómo las limitaciones del sistema pueden afectar el diseño del proyecto.
- Describir cómo el compromiso puede ayudar a mantener en marcha un proyecto e inspirar creatividad.

## Preparación

- Recorra **Ejemplos de Proyectos Curso F** para hacerse una idea de las fortalezas, debilidades y limitaciones de la herramienta.
- Decida si hará que sus estudiantes realicen o no la sección **Revisions** (recomendada para el Curso F).
- Imprima una copia de **Proceso de Diseño - Guía de preparación para el docente** para cada estudiante.
- Modifique **Proyecto Final de Fundamentos de Ciencias de la Computación - Rúbrica** para que sea acorde a los objetivos de su clase e imprima una copia para cada estudiante.
- Modifique **Diseño del Proyecto Final - Ficha** para que sea acorde a su clase e imprima un paquete para cada estudiante.

## Enlaces

**¡Importante!** Haz una copia de los documentos que piensas compartir con los estudiantes.

Para los docentes

- **Ejemplos de Proyectos Curso F**
- **Proceso de Diseño - Guía de preparación para el docente** [Hacer una copia](#)
- **Diseño del Proyecto Final - Ficha** [Hacer una copia](#)

- **Proyecto Final de Fundamentos de Ciencias de la Computación** - Rúbrica
- **72 formas creativas de que tus estudiantes muestren lo que saben** - Sitio web
- **Google Slides**

## Vocabulario

- **Definir** - descubrir los detalles de los problemas que estás tratando de resolver.
- **Preparar** - investigar, planificar y adquirir materiales para la actividad que estás a punto de realizar.
- **Reflexionar** - pensar cuidadosamente en algo con la intención de mejorar el resultado en el futuro.
- **Probar** - intentar hacer algo.

# Guía docente

## Día 1 - Explora Ideas Para Proyectos (45 min)

### Ejemplos de Proyectos

Objetivo: Esta parte del proceso es una investigación. Los/as estudiantes verán un escenario lleno de proyectos de ejemplo para reutilizar y aprender. Esto no sólo les dará una idea de lo que es posible, sino que también les ayudará a ver las limitaciones de la herramienta.

Dé a los/as estudiantes un día para jugar y reutilizar los proyectos que se encuentran en **Ejemplos de Proyectos Curso F**. Pídales que usen sus diarios (o su cuaderno) para llevar un registro de sus pensamientos e ideas a medida que avanzan.

Esta actividad debe realizarse en las mismas parejas/grupos que trabajarán juntos en los proyectos durante las próximas lecciones.

Asegúrese de que su clase entienda que pasarán las próximas semanas trabajando con sus propios proyectos, así que deben prestar mucha atención a cómo se escribieron estos programas, así como a los conceptos que utilizan.

## Día 2 - El Proceso de Diseño (45 min)

### Definir y Preparar

Meta: Los/as estudiantes presentarán un proyecto y planificarán su estrategia para programar ese proyecto en un solo día. Deben tener un bosquejo del proyecto y una descripción para cuando termine el día.

Preparación de los/as estudiantes para el proceso:

La responsabilidad más importante que usted tiene al iniciar este segmento, es ayudar a su clase a entender el alcance de este proyecto. Los/as estudiantes deben tener claro cuáles son las diversas expectativas para las próximas semanas, de modo que puedan prepararse para su revisión y presentaciones de manera apropiada.

Para ayudar a su clase a manejar este proyecto de múltiples etapas, se les debe entregar el primer día de la planificación tanto la **Diseño del Proyecto Final - Ficha** como la **Proyecto Final de Fundamentos de Ciencias de la Computación - Rúbrica**. Los/as estudiantes podrán entonces seguir la rúbrica de cada paso del camino para predecir cuál será la calificación de su proyecto al final.

La Guía de Diseño del Proyecto Final proporcionará un lugar para que los/as estudiantes capturen pensamientos y procesos relevantes a medida que avanzan, para que estén más preparados/as para sus revisiones y presentaciones al final.

Como profesor/a, debe descargar una copia de los documentos y decidir qué elementos son importantes para usted. Asegúrese de editar o eliminar cualquier cosa que no tenga la intención de atraer la atención del/la estudiante.

Definir y Preparar:

Ahora que la clase tiene su Guía de Diseño del Proyecto Final en la mano, deben comenzar a llenar las preguntas del **Día 1**.

Es probable que los/as estudiantes tengan que acudir a sus notas al revisar los proyectos de ejemplo, especialmente si no tienen acceso a los niveles de proyectos de Artista o Play Lab en línea mientras planifican.

Los/as estudiantes deben concentrarse en definir y planificar su proyecto durante el Día 1, y no pasar a la construcción hasta que sus ideas hayan sido escritas y/o dibujadas.

#### 💡 Sugerencia para la lección:

Deje 5 minutos al final del día para que los/as estudiantes intercambien sus Guías de Diseño Proyecto Final para ver el trabajo de los otros/as. Esto ayudará a asegurarse de que no se ha omitido o pasado por alto algo.

Si los/as estudiantes se quedan estancados/as, ayúdelos/as con sus ideas haciendo preguntas y recordando ejemplos, en lugar de ofrecer soluciones.

## Día 3. Construye Tu Proyecto (45 min)

### Intentar

Objetivo: Los/as estudiantes usarán este día para construir una versión inicial de su proyecto.

Equipados con su Guía de Diseño de Proyecto Final, los/as estudiantes deben dirigirse a los computadores para comenzar a dar vida a sus proyectos.

Este proceso se completará con muchas pruebas y errores. Es probable que los proyectos se conviertan en versiones truncadas del alcance original (si no se modifican por completo). Recuerde a los/as estudiantes que este tipo de compromiso es común en el diseño de software, pero deben asegurarse de documentar las razones de los cambios en su producto.

No deje que la clase se olvide de llenar sus Guías de Diseño del Proyecto Final a medida que avanzan. Podría ser útil sugerir que las parejas/grupos tomen un descanso y trabajen con la guía para comenzar a discutir estas preguntas a mitad de su tiempo en el laboratorio. Alternativamente, el navegador puede mantener los ojos abiertos para obtener respuestas pertinentes mientras el conductor programa.

Asegúrese de que cada miembro del equipo tenga su propia Guía de Diseño del Proyecto Final, ya que hay preguntas sobre los pensamientos y comportamientos individuales de cada estudiante que necesitan ser capturados a lo largo del camino.

## Día 4 (Recomendado para 5º Grado) - Revisa Tu Proyecto (45 min)

### Reflexionar e Intentar nuevamente

Objetivo: Los/as estudiantes trabajarán con otro grupo para dar y recibir retroalimentación en función de hacer el proyecto del otro más fuerte.

#### Reflexionar

Para las reflexiones, haga que cada grupo se junte con otro para probar su proyecto. Después de unos 10 minutos, pida a los grupos que discutan las preguntas en la Guía de Diseño del Proyecto Final.

Anime a los/as estudiantes a hacer las preguntas en la guía y a escribir los comentarios de los equipos que lo han revisado para que puedan volver a consultarlo más tarde. Esta debe tomar aproximadamente 15 minutos más.

#### Intentar nuevamente

Con sus nuevas reflexiones en la mano, los/as estudiantes pueden volver a sus computadores para hacer ediciones. Con sólo 10 minutos para terminar, es probable que tengan que seleccionar sólo la retroalimentación más importante para incorporar.

#### 💡 Sugerencia para la lección:

Los/as profesores/as deben evitar asignar la parte final del trabajo del proyecto como tarea a menos que estén seguros de que los/as estudiantes viven muy cerca uno del otro y tienen acceso a Internet en casa.

## Día 5 & 6 - Presenta Tu Proyecto (45 min cada uno)

### Presentaciones

Objetivo: Los/as estudiantes crearán y presentarán sus proyectos de una manera aprobada (escrita, oral o usando multimedia).

Crear:

Idealmente, tendrá tiempo de clase disponible para que sus estudiantes trabajen en sus presentaciones. Esto les permitirá incorporar ricos componentes multimedia, como **Google Slides**. Para otras ideas de presentación, visite **72 formas creativas de que tus estudiantes muestren lo que saben - Sitio web**.

#### 💡 Lesson Tip:

Si está buscando una sección de esta serie para asignar como tarea, ¡ésta es! Los proyectos no tienen que ser presentados en forma electrónica, así que esta es una gran opción fuera de línea.

Anime a los/as estudiantes a incluir toda la información de la Sección J de la Guía de Diseño del Proyecto Final en su presentación, así como dos o más preguntas de la Sección K.

Presente:

Los/as estudiantes deben mostrar sus aplicaciones primero, luego pueden discutir las preguntas que cubrieron en sus presentaciones.

Puede ser muy útil que los/as estudiantes se anoten en un orden específico para dar sus presentaciones, de modo que puedan disfrutar de las demostraciones de sus compañeros/as de clase, sin preocuparse de si serán llamados/as a continuación.

## Actividad de Extensión

Si sus estudiantes ya se sienten cómodos con los conceptos de codificación, motíuelos a que intenten crear sus proyectos en otra plataforma, como **Scratch** o **Alice**.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 26: El Proceso de Diseño

Proyecto

## Descripción

En el transcurso de cinco lecciones, los/as estudiantes irán construyendo un proyecto de diseño propio utilizando Play Lab o Artista como entorno de programación. En esta parte del proyecto, los/as estudiantes aprenderán sobre el proceso de diseño y cómo implementarlo en sus propios proyectos, [here](#).

## Propósito

Los/as estudiantes pueden estar listos/as para ir directamente a la construcción de sus proyectos, pero esta lección ayudará a darle forma a sus ideas y transformarlas en planes. Esta estructura mantendrá a los/as soñadores/as con los pies en la tierra e iluminará el camino para aquellos/as que se sienten en la oscuridad.

## Agenda

**Día 2 - El Proceso de Diseño (45 min)**

Definir y Preparar

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Convertir ideas en objetivos y planes razonables.
- Reconocer los posibles obstáculos, como limitaciones de tiempo o errores.

# Guía docente

## Día 2 - El Proceso de Diseño (45 min)

### Definir y Preparar

Los/as estudiantes inventarán un proyecto y planificarán su estrategia para programar ese proyecto en un día. Ellos/as deberían tener un bosquejo y una descripción al final del día.

Preparando a los/as estudiantes para el Proceso:

La responsabilidad más importante que usted tiene en dar el inicio a este segmento es ayudar a su clase a comprender el alcance de este proyecto. Ellos/as deberían tener claro las expectativas de las semanas que siguen para que así puedan prepararse para sus presentaciones de manera adecuada.

Para ayudar a su clase a manejar esta labor de varios pasos, se les debería entregar tanto la **Diseño del Proyecto Final - Ficha** como la **Proyecto Final de Fundamentos de Ciencias de la Computación - Rúbrica** el primer día de planificación. Los/as estudiantes podrán entonces seguir cada paso de la rúbrica para predecir la nota que su proyecto tendrá al final.

La **Diseño del Proyecto Final - Ficha** proporcionará un lugar para que los/as estudiantes capturen los pensamientos y los procesos relevantes mientras trabajan, de esa manera estarán mejor preparados/as para sus presentaciones al final.

Como docente, usted debería decidir qué elementos de estos documentos son importantes para usted y asegurarse de editar o remover cualquier cosa que no necesite del foco de los/as estudiantes.

[Recomendaciones para la lección][3]

Reserve 5 minutos al final del día para que los/as estudiantes intercambien su **Diseño del Proyecto Final - Ficha** para que vean los trabajos de otros. Esto ayudará a asegurarse que nada ha sido omitido o pasado por alto.

[/][3]

Definir y Preparar:

Ahora que la clase tiene la **Diseño del Proyecto Final - Ficha** en sus manos, deberían comenzar a completar las preguntas del **Día 1**.

Los/as estudiantes posiblemente necesitarán revisar sus notas de cuando jugaron con los proyectos de ejemplo, especialmente si no tienen acceso a los niveles de proyectos en Artista o Play Lab mientras planifican.

Ellos/as deberán enfocarse en definir y planificar sus proyectos durante el Día 1 y no pasar a la construcción hasta que sus ideas estén escritas y/o las hayan dibujado.

Si los/as estudiantes se estancan, ayúdelos/as haciendo preguntas y recordando ejemplos, más que ofrecerles la solución.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.



# Lección 27: Construye Tu Proyecto

Proyecto

## Descripción

En el transcurso de cinco lecciones, los/as estudiantes irán construyendo un proyecto de diseño propio utilizando Play Lab o Artista como entorno de programación. Ahora los/as estudiantes tendrán su propio espacio para crear su proyecto con Artista o Play Lab. Esta será la parte más larga del proyecto. La guía de la lección que entrega una reseña de las cinco etapas del proceso, se puede encontrar al principio del proceso del proyecto, [here](#).

## Propósito

Esta lección proporciona a los/as estudiantes suficiente tiempo para construir y revisar sus proyectos. El ensayo y error inevitablemente considerado en esta lección, enseñará sobre la solución de problemas y la perseverancia.

## Agenda

**Día 3. Construye Tu Proyecto (45 min)**

Intentar

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Usar el diseño planificado como modelo para la creación.
- Superar los obstáculos, como limitaciones de tiempo o errores.

# Guía docente

## Día 3. Construye Tu Proyecto (45 min)

### Intentar

Los/as estudiantes usarán este día para construir una versión inicial de su proyecto.

Equipados/as con su **Diseño del Proyecto Final - Ficha**, deberían dirigirse a los computadores para comenzar a dar vida a sus proyectos.

Este proceso se completará con suficiente uso del ensayo y error. Es posible que los proyectos sean versiones truncadas del original (si es que no totalmente cambiado). Recuérdeles que este tipo de cambios es común en el diseño de software, pero deben asegurarse de documentar las razones para los cambios en su producto.

No deje que la clase se olvide de llenar sus **Diseño del Proyecto Final - Ficha** a medida que avanzan. Podría ser útil sugerir que las parejas/grupos tomen un descanso en la hoja de trabajo para comenzar a discutir estas preguntas a mitad de su tiempo de laboratorio. Alternativamente, el navegante puede mantener los ojos abiertos para obtener respuestas pertinentes mientras el conductor codifica.

Asegúrese de que cada miembro del equipo tenga su propia Hoja de Trabajo de Diseño del Proyecto Final, ya que hay preguntas sobre los pensamientos y comportamientos individuales de cada estudiante que necesitan ser capturados a lo largo del camino.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ **AP** - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 28: Revisa Tu Proyecto

Proyecto

## Descripción

En el transcurso de cinco lecciones, los/as estudiantes estarán construyendo un proyecto de su propio diseño usando ya sea Play Lab o Artista como su entorno de programación. Ahora que los proyectos están contruidos, los/as estudiantes tienen la oportunidad de obtener retroalimentación de sus compañeros/as y revisar sus proyectos. La reseña de las cinco etapas del proceso se puede encontrar al principio del proceso del proyecto, [here](#).

## Propósito

Esta lección ayuda a los/as estudiantes a dar un paso atrás y ver su proyecto desde una nueva perspectiva. Aquí, los/as estudiantes podrán decidir si han alcanzado sus objetivos. Si no lo han hecho, esta lección les da tiempo y espacio para completar el proyecto.

## Agenda

**Día 4 - Revisa Tu Proyecto (45 min)**

Reflexionar e Intentar nuevamente

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Determinar si los criterios establecidos en una rúbrica se han cumplido en el proyecto actual.
- Redactar e implementar planes para resolver cualquier problema en el código.

# Guía docente

## Día 4 - Revisa Tu Proyecto (45 min)

### Reflexionar e Intentar nuevamente

Objetivo: Los/as estudiantes trabajarán con otro grupo para dar y recibir retroalimentación en función de hacer el proyecto del otro más fuerte.

#### Reflexionar

Para las reflexiones, haga que cada grupo se junte con otro para probar su proyecto. Después de unos 10 minutos, pida a los grupos que discutan las preguntas en la Guía de Diseño del Proyecto Final.

Anime a los/as estudiantes a hacer las preguntas en la guía y a escribir los comentarios de los equipos que lo han revisado para que puedan volver a consultarlo más tarde. Esta debe tomar aproximadamente 15 minutos más.

#### Intentar nuevamente

Con sus nuevas reflexiones en la mano, los/as estudiantes pueden volver a sus computadores para hacer ediciones. Con sólo 10 minutos para terminar, es probable que tengan que seleccionar sólo la retroalimentación más importante para incorporar.

#### Sugerencia para la lección:

Los/as profesores/as deben evitar asignar la parte final del trabajo del proyecto como tarea a menos que estén seguros de que los/as estudiantes viven muy cerca uno del otro y tienen acceso a Internet en casa.

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.

# Lección 29: Presenta Tu Proyecto

Proyecto

## Descripción

En el transcurso de cinco lecciones, los/as estudiantes irán construyendo un proyecto de diseño propio utilizando Play Lab o Artista como entorno de programación. Finalmente, los/as estudiantes podrán presentar su trabajo terminado a sus compañeros/as o compartirlo con sus seres queridos, con un link especial. La guía de lecciones que resume las cinco etapas del proceso se encuentra al principio del proceso del proyecto, [here](#).

## Propósito

En este punto, los/as estudiantes han trabajado muy duro en sus proyectos, por lo que esta lección pretende ofrecer un espacio para que compartan sus proyectos. Esta lección construirá una comunidad de apoyo donde los/as estudiantes construirán su propia confianza y se sentirán conectados con sus esforzados/as compañeros/as.

## Agenda

**Día 5 & 6 - Presenta Tu Proyecto (45 min cada uno)**

Presentaciones

Ver en Code Studio

## Objetivos

Los estudiantes podrán:

- Indicar claramente dónde se cumple cada punto de criterio de la rúbrica en el código del proyecto finalizado.
- Articular el proceso de diseño y cómo ayudó a dar forma al proyecto finalizado.

# Guía docente

## Día 5 & 6 - Presenta Tu Proyecto (45 min cada uno)

### Presentaciones

Los/as estudiantes crearán y presentarán sus proyectos de una manera aprobada (escrita, oral o usando multimedia).

#### Crear:

Idealmente, tendrá tiempo de clase disponible para que sus estudiantes trabajen en sus presentaciones. Esto les permitirá incorporar ricos componentes multimedia, como **Google Slides**. Para otras ideas de presentación, visite **72 formas creativas de que tus estudiantes muestren lo que saben - Sitio web**.

Anime a los/as estudiantes a incluir toda la información de la Sección J de la Guía de Diseño del Proyecto Final en su presentación, así como dos o más preguntas de la Sección K.

#### Presente:

Los/as estudiantes deben mostrar sus aplicaciones primero, luego pueden discutir las preguntas que cubrieron en sus presentaciones.

Puede ser muy útil que los/as estudiantes se anoten en un orden específico para dar sus presentaciones, de modo que puedan disfrutar de las demostraciones de sus compañeros/as de clase, sin preocuparse de si serán llamados/as a continuación.

#### 💡 Lesson Tip:

Si está buscando una sección de esta serie para asignar como tarea, ¡ésta es! Los proyectos no tienen que ser presentados en forma electrónica, así que esta es una gran opción fuera de línea. Otras formas de presentar sus proyectos en línea o sin conexión, pueden ser:

- Reporte
- Publicación en un blog
- Online
- Frente a la clase con un afiche

## Normas de aprendizaje

Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ AP - Algoritmos y programación



This curriculum is available under a Creative Commons License (CC BY-NC-SA 4.0).

Español (Latinoamérica) ▼

**Contáctanos** si estás interesado en contar con la licencia de los materiales de Code.org para uso comercial.