

# Lección 1: Detener la maldad en Internet

35 minutos

## Resumen



Esta lección fue creada originalmente por Common Sense Education. [Más información.](#)

Internet está repleto de toda clase de personas interesantes, pero a veces, algunas personas pueden ser algo crueles con los demás. A través de un juego de roles, ayuda a tus estudiantes a entender por qué a veces es más fácil ser “malo” en internet que en persona, y a lidiar con el maltrato en internet, cuando se les presente.

## Propósito

Common Sense Education creó esta lección para enseñar a los estudiantes sobre la maldad y lo que deben hacer si la encuentran en línea.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **IC** - Impacts of Computing

## Agenda

**Aprender: ¿qué es el maltrato? (10 minutos)**

**Vocabulario Clave**

**Representación: en línea versus cara a cara (5 minutos)**

**Explorar: ALTO, basta de maltrato (15 minutos)**

**Actividad de cierre: momento para reflexionar (5 minutos)**

**Ampliación del aprendizaje**

**Niveles**

## Objetivos

Los estudiantes podrán:

- Identificar formas de reaccionar ante palabras crueles en línea, con el método A-L-T-O.
- Entender cómo es el maltrato en línea y cómo puede hacer sentir a las personas.

## Preparación

- Revisar el material instructivo.
- Imprimir guías de trabajo para cada estudiante.
- Traer peluches o juguetes a la clase.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los profesores

- **Basta de maltrato en línea: diapositivas** - Conjunto de Diapositivas

▼ Hacer una copia

Para los estudiantes

- **Guía del estudiante: ALTO, basta de maltrato en línea** - Guía para el estudiante

▼ Hacer una copia

# Guía Didáctica

## Aprender: ¿qué es el maltrato? (10 minutos)

### Vocabulario Clave

- **en línea:** usar un computador, teléfono o tablet para visitar un sitio web o una app

**Pregunta:** ¿Qué se considera un mal comportamiento? (**Diapositiva 3**)

Las respuestas de los estudiantes serán variadas, pero recalca las siguientes: burlarse de cómo se ve alguien más, decir mentiras sobre alguien, decir cosas feas a los demás.

**Pregunta:** ¿Cómo hace sentir a los demás este tipo de comportamiento?

Invita a los estudiantes a responder. Las respuestas serán variadas, pero las posibles respuestas podrían incluir: herido, enojado, molesto y asustado.

**Pregunta:** ¿Qué debes hacer si alguien te trata mal o si ves a alguien trata mal a alguien más?

Invita a los estudiantes a responder y enfatiza que ellos siempre deben decirle a un adulto de confianza si sufren o ven a alguien más sufrir cualquier forma de maltrato.

**Comenta:** *Internet está lleno de cosas divertidas e interesantes, pero a veces, la gente puede ser desagradable con otras personas en línea.*

Aclara que **en línea** significa *significa usar un computador, teléfono o tablet para visitar un sitio web o una app.* (**Diapositiva 4**)

Explica que algunos ejemplos de maltrato en línea podrían ser:

- Enviar un mensaje cruel a alguien
- Publicar cosas crueles sobre alguien en un sitio web
- Burlarse de alguien más en internet
- Hacer cosas malas al personaje de alguien en un juego en línea

Si los estudiantes están familiarizados con el término **ciberbullying** (usar dispositivos digitales, sitios y apps para intimidar, lastimar y molestar a alguien), puedes usar dicho término.

**Pregunta:** ¿Alguna vez han visto a una persona hacer sentir mal a otra persona en internet?

Invita a los estudiantes a conversar en parejas, pero enfatiza que pueden contar sus anécdotas sin usar los nombres reales.

Explica que van a aprender más sobre cómo ocurre el maltrato en línea y qué deben hacer cuando les ocurra a ellos o a alguien que conozcan.

## Representación: en línea versus cara a cara (5 minutos)

**Pregunta:** ¿Por qué creen que alguien sería malo con otra persona en internet? (**Diapositiva 5**)

Las respuestas serán variadas, entre ellas podrían incluir que “era una broma” o que “se puede sentir mal consigo mismo e intenta desquitarse con los demás”. Los estudiantes podrían incluso notar que la gente podría decir cosas por internet que no diría en persona.

**Comenta:** *Para las personas, a veces, es más fácil ser malo en línea que en persona, ya que no tienen que ver a los demás a la cara ni pueden notar sus sentimientos.*

**Explica** que van a fingir que el peluche es, en realidad, una persona que conoces. Mientras encaras al peluche, di (con un tono burlesco): *¡No puedo creer que te gusten los superhéroes! ¡Eso es algo que vería mi hermano menor!*

Luego, proyecta la **Diapositiva 6** y pide a los estudiantes que lean, en silencio, la conversación de chat.

**Pregunta:** *¿Cuál fue la diferencia entre escribir algo malo y decirlo en persona?* Proyecta la **Diapositiva Z** y recopila las respuestas de los estudiantes en el cuadro.

Enfatiza que ambas fueron malas actitudes (tanto cara a cara como en línea), pero que probablemente es más fácil escribirlo que decirlo en persona, ya que no puedes ver la reacción de la otra persona y ella no puede oír el tono de tu voz.

**Comenta:** *Recuerden, si hay algo que no le dirían a otra persona a la cara, entonces definitivamente no está bien decirlo en internet.*

## Explorar: ALTO, basta de maltrato (15 minutos)

**Entrega** las **Guías del estudiante - ALTO, basta de maltrato en línea**. Pide a un estudiante que lea el escenario de la **Diapositiva 8** en voz alta. Dale a las parejas cinco minutos para que contesten las primeras dos preguntas.

**Invita** a los estudiantes a compartir sus respuestas con la clase.

**Comenta:** *Si alguna vez los maltratan en línea, ¡recuerden decir ALTO!*

**Proyecta** la **Diapositiva 9** y revisa las cinco reglas para lidiar con el maltrato en línea. Mientras revisan cada regla, haz las siguientes preguntas de seguimiento para guiar la discusión.

**A:** aléjate del dispositivo

- *¿Por qué crees que deberías dejar de usar el dispositivo y alejarte?*

**L:** llama a un adulto de confianza

- *Si alguien te hace sentir enojado, triste o asustado en línea, ¿a qué adultos puedes recurrir para pedir ayuda?*

**T:** ten permiso para visitar sitios

- *¿Por qué es importante estar en línea sólo con un adulto o cuando un adulto dice que está bien?*

Haz énfasis en que algunos sitios no están hechos para niños, pero que permiten que los niños puedan comunicarse con otras personas. Esto abre la posibilidad de que ocurran cosas malas en línea.

**O:** oye el consejo de pausar y pensar en línea

- *¿Ayudaría hablar con la persona que te está maltratando?*

**Pregunta:** *¿Qué consejo le darías a Jada para afrontar esta situación? ¡Recuerda, ALTO!*

Pide a los estudiantes que respondan la pregunta 3 de la **Guía del estudiante - ALTO, basta de maltrato en línea** con un compañero y luego invítalos a compartir sus respuestas con la clase.

- **A:** Jada debería alejarse del computador.
- **L:** Jada debería buscar a un adulto de confianza.
- **T:** Jada no debería entrar a internet ni al sitio web del poni otra vez, hasta que un adulto diga que está bien.
- **O:** Jada debería tomarse su tiempo y esperar a estar más calmada. Si Jada y Miguel son buenos amigos, quizás Jada quiera decirle a Miguel cómo la hizo sentir, después de hablar con un adulto. Si Miguel sigue siendo malo con ella, Jada debe inmediatamente decirle a un adulto.

**Pregunta:** ¿Cuál de las cuatro reglas de A-L-T-O crees que es la más importante?, ¿por qué?

Dales a los estudiantes un momento para reflexionar y luego haz una encuesta rápida sobre cuál es la regla más importante. Asegúrate de enfatizar que “llama a un adulto de confianza” es lo más importante que deben tener en cuenta si alguna vez sufren maltrato en línea.

## Actividad de cierre: momento para reflexionar (5 minutos)

**Comenta:** Entramos a internet para ver videos, enviar mensajes a los que conocemos, jugar videojuegos y hacer tareas. A veces, las personas dicen cosas malas o que asustan. Si alguna vez alguien es malo contigo en internet, ¡recuerda decir A-L-T-O!

**Dirige** a los estudiantes al ejercicio de Momento para reflexionar, en la página 2 de la **Guía del estudiante - ALTO, basta de maltrato en línea**. Lee las instrucciones en voz alta y permíteles completar la reflexión de forma individual. (**Diapositiva 10**)

**Invita** a los estudiantes a compartir sus reflexiones con la clase. Recoge las guías de trabajo para evaluar el aprendizaje de los estudiantes.

## Ampliación del aprendizaje

Aquí hay algunos recursos adicionales que puedes compartir con los estudiantes para ampliar el aprendizaje:

- **Actividad familiar**
- **Consejos familiares**

## Niveles

Actividad sin cone...



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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

# Lección 2: Mis amigos robóticos Jr.

45 minutos

## Resumen

En estas lecciones de **context-setting**, los estudiantes utilizarán un conjunto de símbolos para instruir a un “robot” para apilar tazas en diferentes patrones. Los alumnos se turnarán para participar como el robot, respondiendo únicamente al algoritmo definido por sus compañeros.

## Propósito

Esta lección sin conexión encomienda a toda la clase, juntos como un gran equipo, una tarea simple: que un “robot” apile vasos formando un diseño específico. Esta actividad sienta la base de la programación con la que los estudiantes trabajarán a lo largo de este curso, ya que aquí aprenderán la importancia de definir un algoritmo comunicado claramente.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (5 minutos)

Hablar con robots

### Actividad (30 minutos)

Mis amigos robóticos

### Actividad de cierre (10 minutos)

Reflexión

## Objetivos

Los estudiantes podrán:

- Prestar atención a la precisión al dar instrucciones.
- Identificar y solucionar errores en secuencias de instrucciones

## Preparación

- Prepara una pila de 20 vasos desechables (o trapezoides de papel) para cada grupo (2-3 estudiantes).
- Muestra los símbolos de **\*Simbología**, de manera que los estudiantes puedan recurrir a ellos durante la lección.
- (Opcional) Imprime un **\*Ideas para apilar vasos** para cada grupo (2-3 estudiantes).
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los estudiantes

- **Mis amigos robóticos** - Espacio entre vasos
- **Mis amigos robóticos** - Ideas para apilar vasos

- **Mis amigos robóticos** - Plantilla de trapecios en papel
- **Mis amigos robóticos** - Clave de símbolos
- **Mis amigos robóticos** - Video sin conexión ([Descarga](#))

## Vocabulario

- **Algoritmo** - una lista de pasos para terminar una tarea.
- **Bug (error)** - parte de un programa que no funciona correctamente.
- **Depuración** - (v) 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 Didáctica

### Actividad previa (5 minutos)

#### Hablar con robots

**Mostrar:** miren uno de los videos a continuación para contextualizar a los estudiantes sobre la clase de cosas que puede hacer un robot:

- **[Robot Asimo de Honda](#)** (3:58)
- **[Robot que dibuja huevos](#)** (3:15)
- **[Robot bailarín de Lego](#)** (1:35)

**Análisis:** refiérete al video que hayas elegido y pregunta cómo creen que el robot sabía lo que debía hacer. *¿Un robot “entiende” realmente lo que le dices? ¿Le preocupa meterse en problemas si no hace lo que le dicen?*

**Objetivos del análisis:** este breve análisis tiene como objetivo hacer énfasis en que, a pesar de que los robots pareciesen comportarse como humanos, realmente sólo responden a su programa. Es probable que los estudiantes hagan referencia a algunos robots de las películas y la televisión con comportamientos más humanos. Guíalos a considerar robots que hayan visto u oído en la vida real, como el *Roombas*, o incluso asistentes digitales como *Amazon Alexa*.

**Comenta:** *Los robots sólo pueden hacer lo que les dicen que deben hacer. Sin embargo, esas instrucciones no se dan usando palabras. Con el propósito de hacer algo, un robot necesita una lista de pasos que pueda leer. Hoy vamos a aprender qué se necesita para que eso ocurra.*

## Actividad (30 minutos)

Mis amigos robóticos

Actividad sin cone...

Mis Amigos Robóticos - Actividad Desconectada



Pick Up Cup



Put Down Cup



Step Forward



Step Backward

### Introducción y ejemplo

**Preparación:** ten disponibles pilas de vasos o recorta papeles con forma de trapezoide para todos los grupos.

## Reglas útiles:

- **Arriba** significa que el vaso se debe levantar automáticamente tan alto como sea necesario.
- **Abajo** significa que el vaso se debe bajar automáticamente hasta que se pose sobre algo.
  - La mano debe volver automáticamente a la pila de vasos después de colocar un vaso.
- **Adelante** significa que el robot se mueve un paso hacia adelante (medio vaso de distancia).
- **Atrás** significa que el robot se mueve un paso hacia atrás (medio vaso de distancia).
  - Nota: los estudiantes quizás no usen atrás en esta etapa, a menos que quieran construir la torre de vasos en reversa (lo que no está mal).
- Los programadores no están autorizados a hablar mientras el robot está en movimiento. Esto incluye gritar la respuesta o indicar cuando el robot se ha equivocado.
- Los programadores deben levantar su mano si ven un error.

## Opciones de diferenciación:

**Simplificación:** ¿todo esto parece un poco complicado para tus estudiantes?

No olvides mostrarlo, a modo de ejemplo, enfrente de la clase hasta que los estudiantes entiendan todas las reglas. Si aún es algo confuso, intenta realizar esta actividad completa en conjunto con el curso, usando voluntarios como robots, en lugar de dividirlos en grupos.

**Avance:** ¿tus estudiantes están más avanzados? ¿Quieres que esta lección se relacione más de cerca con los desafíos en línea? Aquí hay algunas modificaciones para conseguirlo:

- Una flecha corresponde a un movimiento:
  - Cuando un vaso es removido de la pila, debe volver al nivel de la mesa antes de ser movido.
  - Los estudiantes deben usar varias flechas hacia "arriba" para levantar el vaso varios niveles.
  - Los estudiantes deben usar varias flechas hacia "abajo" para bajar el vaso varios niveles.
  - Los estudiantes deben usar las flechas hacia "atrás" para retroceder a la pila de vasos.

**Mostrar:** muestra la **\*Simbología** o escribe las acciones permitidas en la pizarra. Asegúrate de que esta simbología esté en un lugar visible para toda la clase, durante toda la actividad. Explícales que éstas serán las únicas cuatro acciones que podrán usar durante el ejercicio. Para esta tarea, los estudiantes deberán indicarles a sus amigos "robots" cómo construir una torre de vasos específica usando sólo los comandos listados.

**Ilustrar:** con el propósito de explicar cómo se supone que deba funcionar el ejercicio, enseña a la clase la forma de crear y seguir un algoritmo para replicar un patrón simple. Coloca una pila de vasos frente a ti para comenzar.

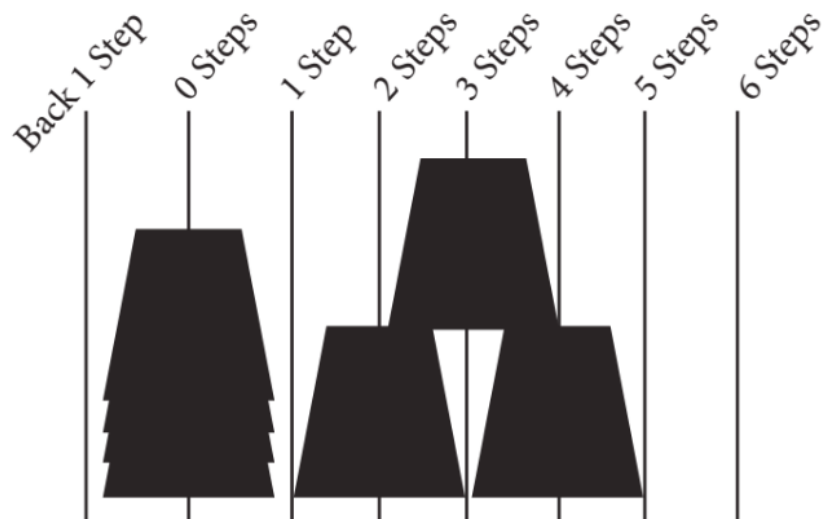
**Mostrar:** sostén el modelo que planeas realizar a modo de ejemplo. Un patrón simple de tres vasos es perfecto para comenzar.





**Pregunta:** pregunta cuál de las *cuatro instrucciones permitidas* debería ser la primera instrucción. El primer movimiento debería ser “tomar vaso”. Si los estudiantes sugieren algún otro paso de la lista, realiza esa acción y permíteles ver el error. Si sugieren algo que no esté en la lista, reacciona con un gesto evidente de malfuncionamiento y hazles saber que el comando es desconocido.

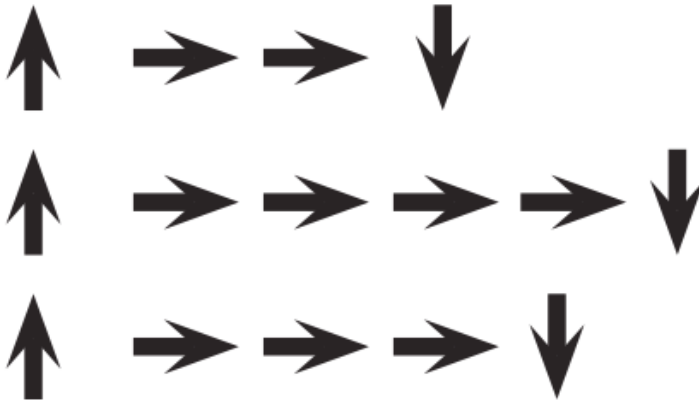
Con el vaso en la mano, pídeles que continúen dando instrucciones hasta colocar el primer vaso. Esta es la ocasión perfecta para aclarar que “un paso adelante” y “un paso atrás” implican un movimiento de medio vaso de distancia. Usa la imagen a continuación como referencia.



Continúa pidiendo instrucciones hasta que hayas realizado el diseño completo.

Una vez que su pila está completa, indícales que acaban de darte una lista de pasos para completar una tarea. Eso es un algoritmo. Los algoritmos son ideales para compartir ideas, pero comunicarlos palabra por palabra puede tardar mucho. ¡Para eso están los símbolos! Cuando escribes un algoritmo con símbolos que pueda entender un robot (o una computadora), ¡estás programando!

Pide a la clase que te ayude a escribir el “programa” para ese primer movimiento, cambiando el texto por flechas. Luego, trabajen juntos para escribir el resto de los movimientos necesarios para completar el modelo. Dependiendo de la confianza de tus estudiantes, es posible alternar frecuentemente entre escribir el código y actuar como el “robot” para ejecutarlo, o puedes alentarlos a escribir el código completo antes de ejecutarlo. Una posible solución se vería algo así:



**Voluntario:** una vez que la clase haya completado el programa de ejemplo, pide a un estudiante que actúe como el “robot” para asegurarte de que el programa realmente funciona. Recomiéndales que el robot diga las instrucciones en voz alta mientras “ejecuta” el código.

## Programación de tus robots

**Grupo:** separa a los estudiantes en grupos de cuatro. Luego, cada grupo se volverá a dividir en dos parejas. Cada pareja desarrollará su propio programa para que sea “ejecutado” por la otra pareja.

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

**Mostrar:** muestra las **\*Ideas para apilar vasos** a la clase o entrega copias individuales a cada grupo. Haz que cada pareja (no grupo) elija qué patrón quieren que hagan sus robots. Propón patrones simples para comenzar y que luego elijan otros más complejos. Alienta a las parejas a que mantengan su elección en secreto, de manera que la otra pareja del grupo no sepa cuál eligieron.

**Análisis:** dale tiempo a cada pareja para analizar cómo se debería construir la torre de vasos, usando sólo los símbolos provistos. Asegúrate de que cada pareja escriba el “programa” en donde pueda leerlo el “robot”.

**Acción:** una vez que ambas parejas del grupo hayan terminado sus programas, pueden turnarse para ser los “robots” y seguir las instrucciones que la otra pareja haya escrito. Alienta a los estudiantes a que vigilen cuidadosamente a sus “robots”, a fin de asegurarse de que sigan las instrucciones. Si un estudiante ve un error y levanta su mano, permite que el robot termine de seguir las instrucciones antes de intervenir. A continuación, pídeles que analicen y conversen sobre el posible error y lleguen a una solución. Repite hasta construir la torre como corresponde.

**Observación:** vigila que los grupos no intenten tomar atajos, al agregar cosas extras (como números) a sus códigos. Si algún grupo lo hace, felicítalos por su ingenio; sin embargo, recuérdales que, en este ejercicio, los “robots” no entienden nada que no sean los símbolos provistos. Si gustas, puedes sugerirles que guarden esas soluciones brillantes para la próxima vez que jueguen este juego, ¡ya que podrían tener que usarlas pronto!

**Repetición:** dependiendo del tiempo disponible, vuelve a armar los grupos y dales la oportunidad de elegir un modelo diferente. Cada vez que los grupos repitan el proceso, alientalos a elegir un modelo más y más complicado.

**Análisis:** luego de que cada estudiante haya tenido la oportunidad de ser el “robot”, reagrupa a la clase para conversar sobre sus experiencias. El objetivo de este análisis es dar a los estudiantes el espacio para darle sentido a sus experiencias como robot y como programador. Las preguntas son intencionalmente amplias, pero están diseñadas para que los estudiantes reflexionen sobre los desafíos de escribir un programa claro y las limitaciones de un robot o computador para interpretar instrucciones.

- *¿Cuál fue la parte más difícil de elaborar las instrucciones?*
- *¿Alguien encontró un error en una de sus instrucciones justo antes de que el robot la ejecutara?*
  - *¿Cuál fue el error?*
  - *¿Por qué crees que no lo notaste cuando estabas escribiendo el programa?*
- *Cuando fuiste el robot, ¿cuál fue la parte más difícil de tener que seguir las instrucciones?*

## Actividad de cierre (10 minutos)

### Reflexión

#### Sugerencias:

- Dibuja tu propia pila de vasos que te gustaría que construyera un robot.
- ¿Puedes crear un programa para esa pila de vasos?

💡 Consejo didáctico ▲

Las actividades de reflexión como estas se pueden completar de diversas formas. ¡Escribir un diario o debatir (con grupos o compañeros) son excelentes opciones!



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 3: Programación con Angry Birds

52 minutos

## Resumen

A partir de los personajes del juego Angry Birds, los estudiantes desarrollarán algoritmos secuenciales que les permitirán mover al pájaro desde un lado del laberinto hacia el cerdo que se encuentra al otro lado. Para ello, apilarán bloques de código en una secuencia lineal, lo que hará que el pájaro se mueva derecho, y luego gire a la izquierda o a la derecha.

## Propósito

En esta lección, los estudiantes desarrollarán habilidades de programación y depuración en una plataforma informática. El formato de bloques de estos desafíos les permite a los estudiantes aprender sobre conceptos y secuencias, sin la preocupación de perfeccionar la sintaxis.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (4 minutos)

#### Introducción

### Actividad puente: programación (10 minutos)

#### Transición de “sin conexión” a “en línea”

### Revisión de desafíos en línea en conjunto (3 minutos)

### Actividad Principal (30 minutos)

#### Programación con Angry Birds

### Actividad de cierre (5 minutos)

#### Reflexión

### Ampliación del aprendizaje

### Oportunidad multidisciplinaria

## Objetivos

Los estudiantes podrán:

- Identificar y encontrar bugs en un programa.
- Traducir movimientos a una serie de comandos.

## Preparación

- Realiza los desafíos para encontrar cualquier área potencialmente problemática para los estudiantes.
- (Opcional) Elige un par de desafíos para hacer en conjunto con la clase.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los estudiantes

- **Laberinto de la actividad puente**
- **Bloques del laberinto sin conexión** - Manipulativos

## Vocabulario

- **Algoritmo** - una lista de pasos para terminar una

tarea.

- **Bug (error)** - parte de un programa que no funciona correctamente.
- **Depuración** - (v) encontrar y solucionar problemas en un algoritmo o programa.
- **Secuencias** - Poner los comandos en el orden correcto para que los computadores puedan leerlos.

## Guía Didáctica

### Actividad previa (4 minutos)

#### Introducción

Esta lección se basa en muchos de los conceptos sin conexión que los estudiantes han aprendido en las semanas previas a esta primera actividad en línea. Es importante que retomes estos conceptos (como perseverancia, depuración, algoritmos y programas) de manera que tus estudiantes puedan aprovecharlos en sus actividades en línea.

**Mostrar:** muestra una torre de vasos del ejercicio previo “Mis amigos robóticos”.

**Análisis:** pide a los estudiantes recordar los símbolos usados en “Mis amigos robóticos”.

- *¿Qué pasa cuando el robot lee las flechas?*

Alienta a los estudiantes a pensar en los consejos de depuración:

- *¿Todo iba bien en el primer paso?*
- *¿Qué tal en el segundo?*
- *¿Cuándo comenzó a fallar?*

**Transición:** una vez que estés seguro de que tus estudiantes recuerdan “Mis amigos robóticos”, puedes continuar con la actividad puente.

### Actividad puente: programación (10 minutos)

#### Transición de “sin conexión” a “en línea”

Esta breve actividad ayudará a los estudiantes a conectar los conceptos de perseverancia y depuración con los desafíos en línea que están a punto de realizar.

**Mostrar:** proyecta una copia del **\*Laberinto de la actividad puente**, para que la clase pueda verla. Asegúrate de haber colocado con anticipación los bloques de movimiento en el espacio de trabajo. Para esto, utiliza los **\*Bloques del laberinto** y ubícalos de la siguiente forma:



**Ilustrar:** cuéntales a tus estudiantes que el espacio de trabajo que tienes aquí se ve exactamente igual al que ellos verán cuando comiencen a realizar los desafíos en línea en Code.org. Como docente, hazles saber que eres TAN LISTO que ya pusiste todos los bloques de código que necesitas para resolver el desafío. Luego, pídeles que vean cómo “ejecutas” el código, moviendo tu dedo (o una moneda, o cualquier clase de indicador) por el área.

No pasará mucho tiempo hasta que choques con el bloque de TNT. Finge frustración.

**Comenta:**

- *¿Qué piensan que estoy sintiendo en este momento?*
- *¿Debería rendirme?*
- *¿Debería deshacerme de todo el código y empezar desde el principio?*

**Reflexión:** *¿Cómo puedo reparar este programa, de manera que no choque con el bloque de TNT?*

**Interacción:** pide a los estudiantes que trabajen en soluciones para que el ave rodee el bloque de TNT. Dependiendo de tus estudiantes, podrías pedirles que corrijan cada error, uno por uno (haciendo demostraciones) o que trabajen cómodamente todos juntos para corregir el programa completo.

**Exposición:** pide voluntarios para que reacomoden los bloques en la posición correcta. En conjunto como clase, “ejecuta” el programa una y otra vez, reparando los errores, hasta que el ave haga lo que se supone que deba hacer. Continúa resaltando experiencias que se relacionen con la perseverancia, la frustración y la depuración.

Cuando la clase logre llegar al cerdo, ifelicítalos no sólo por su logro, sino también por su perseverancia!

## Revisión de desafíos en línea en conjunto (3 minutos)

¡Los estudiantes deberían estar listos para ver un desafío real en acción!

💡 Consejo didáctico ▲

Algunos estudiantes podrían tener problemas para hacer que el ave doble en la dirección correcta, particularmente cuando el ave no esté de frente. Recuérdales que cuando decimos “izquierda” o “derecha”, estamos dando instrucciones desde el punto de vista del ave.

**Ilustrar:** muestra el desafío 5 para realizarlo frente a la clase. Este es el mismo desafío que vieron en la actividad puente. Mientras completas este desafío con tu clase, recuérdales a tus estudiantes que no hay ningún problema con cometer errores, y que no olviden que la única forma de tener éxito es ser perseverante.

**Análisis:** *¿Alguien recuerda cómo resolver este desafío?*

Como docente, debes decidir si los estudiantes te recordarán cómo resolverlo desde sus puestos o pedirás voluntarios para que ellos mismos pasen a resolverlo en el computador, bloque a bloque.

**Transición:** ahora que los estudiantes han visto un desafío en línea real, deberían estar preparados para resolverlos por ellos mismos. Continúa en el laboratorio o saca los computadores en tu clase.

## Actividad Principal (30 minutos)

### Programación con Angry Birds

💡 Consejo didáctico ▲

Enseña a los estudiantes la manera correcta de ayudar a sus compañeros:

- No sentarse en la silla del compañero
- No usar el teclado del compañero
- No tocar el mouse del compañero
- Asegurarte de que tu compañero pueda describirte la solución en voz alta antes de que te vayas

🎥 1

Vídeo: Introducción al Laberinto - Programación con Bloques

💻 2-7

Desarrollo de Habilidades

2

3

4

5

6

7

💻 8

Desafío

💻 9

Práctica

💻 10

Predicción

💻 11

Práctica

🚩 12-13

Lecciones adicionales



**Observación:** los docentes son de vital importancia en la educación de las ciencias de la computación y juegan un rol fundamental para producir un ambiente vibrante y colaborativo en la sala de clases. Durante las actividades en línea, el rol del docente es alentar y apoyar. Los desafíos en línea están

estructurados para estar centrados en el estudiante, así que los docentes deben evitar involucrarse cuando los estudiantes tengan problemas para resolverlos. Algunas ideas de cómo hacerlo son:

- Usa la programación en parejas cada vez que sea posible durante la actividad.
- A través de desafíos o preguntas, anima a los estudiantes para que busquen respuestas con sus respectivas parejas.
- Las preguntas sin responder pueden ser delegadas a un grupo cercano, que podría ya tener la respuesta.
- Recuérdales usar el proceso de depuración antes de que te acerques a ayudar.
- Pide a los estudiantes que describan el problema que estén viendo. ¿Qué se supone que debe hacer?, ¿qué hace?, ¿qué te dice eso?
- Recuérdales que la frustración es un paso en el camino del aprendizaje y que la perseverancia dará sus frutos.
- Si un estudiante sigue con problemas para avanzar después de las acciones anteriores, haz preguntas clave para que los estudiantes identifiquen el error por ellos mismos.

**Análisis:** luego de advertir que se acerca el fin de la clase, llama la atención de todos los estudiantes para que reflexionen sobre las experiencias que acaban de vivir.

## Actividad de cierre (5 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- Dibuja una actividad que te haya gustado pero que te haya costado realizar la primera vez. Dibuja o escribe cómo lograste realizarla.

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### Créalo tú mismo

En pequeños grupos, permite que los estudiantes creen sus propios laberintos y se desafíen, entre ellos, a escribir programas para resolverlos. Para más diversión, hagan laberintos en tamaño real, con estudiantes haciendo el papel del ave y del cerdo.

## Oportunidad multidisciplinaria

### Errores, errores y más errores (45-60 minutos)

#### Ciencias de la computación + Lengua y Literatura + Matemáticas

**Errores, errores y más errores** es una actividad opcional alineada con los estándares de Lengua y Literatura, y Matemáticas de Common Core, escrita por nuestra comunidad de docentes. Los estudiantes practicarán la depuración mientras arreglan el código, resuelven un problema matemático y corrigen muestras de escritura.



*Estándares abordados:*

- **CSCS.ELA-LITERACY.L.2.1.B:** formar y usar sustantivos plurales irregulares que ocurren con frecuencia (por ejemplo, pies, niños, dientes, ratones, peces).
- **CSCS.ELA-LITERACY.L.2.2.A:** poner en mayúsculas los días festivos, los nombres de productos y los nombres geográficos.
- **CCSS.ELA-LITERACY.L.2.2.C:** usar un apóstrofe para formar contracciones y posesivos frecuentes.
- **CCSS.MATH.CONTENT.2.OA.B.2:** sumar y restar con fluidez dentro de 20 usando estrategias mentales.
- **CSCS.CONTENIDO.2.NBT.A.4:** comparar dos números de tres dígitos basándose en los significados de los dígitos de las centenas, las decenas y las unidades, utilizando los símbolos  $>$ ,  $=$  y  $<$  para registrar los resultados de las comparaciones.



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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

# Lección 4: Depuración en el laberinto

50 minutos

## Resumen

En esta lección de **construcción de estudios**, los estudiantes encontrarán código preescrito que contiene errores. Deberán pasar por el código existente para identificar errores.

## Propósito

Tus estudiantes podrían llegar a sentirse frustrados con esta lección, dada la naturaleza de la depuración. Depurar es un concepto muy importante en la programación. Los programadores deben trabajar para ser muy buenos enfrentándose a los errores en sus propios programas. La depuración fuerza a los estudiantes a reconocer problemas y superarlos, mientras desarrollan un pensamiento crítico y habilidades de resolución de problemas.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (15 minutos)

#### Introducción

### Actividad Principal (30 minutos)

#### Depuración en el laberinto

### Actividad de cierre (5 minutos)

#### Reflexión

### Ampliación del aprendizaje

## Objetivos

Los estudiantes podrán:

- Modificar un programa existente para resolver errores.
- Predecir dónde fallará un programa.
- Reflexionar sobre el proceso de depuración de una forma apropiada para la edad.

## Preparación

- Realiza los desafíos para encontrar cualquier área potencialmente problemática para tu clase.
- (Opcional) Elige un par de desafíos para hacer en conjunto con tu clase.
- Revisa las **\*Recomendaciones de la lección – FCC: consejos para la actividad principal.**
- Revisa la **\*Guía de depuración**, en la reseña del plan de estudio de este curso.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**iAviso!** Por favor, haga una copia de cualquier documento que planee compartir con los estudiantes.

Para los profesores

- **FCC: consejos para la actividad principal** - Recomendaciones de la lección

- **Guía de Depuración** -

Recurso

Para los estudiantes

- **Programación en parejas** - Vídeo

## Vocabulario

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

## Guía Didáctica

### Actividad previa (15 minutos)

#### Introducción

Pide a los estudiantes que piensen en problemas que tengan que resolver en la vida diaria.

- *¿Cómo reparas algo que no está funcionando?*
- *¿Sigues alguna serie de pasos?*
- *Los desafíos en esta lección ya fueron resueltos para ti (iyupi!), pero parece que no están funcionando bien (ibuu!).*
- *A los problemas en estos programas se les llama "bugs", y será tu misión "depurarlos".*

#### Vocabulario

Esta lección tiene tres nuevos conceptos muy importantes:

- **Bug:** *díganlo conmigo, BAA - G.*

Algo que anda mal, un error.

- **Depurar:** *díganlo conmigo, DEE - PUU - RAR.*

Encontrar y reparar errores.

- **Perseverancia:** *díganlo conmigo PERR - SE - VE - RAN - CIA.*

No rendirse. La perseverancia funciona mejor cuando lo intentas muchas veces, de varias formas.

**Comenta:** *Depurar* es un proceso. Primero debes darte cuenta de que hay un error en tu programa. Entonces, debes revisarlo, paso a paso, hasta encontrar el error. Intenta realizar el primer paso ¿todo funciona?, ¿qué tal en el segundo paso? Si vas paso a paso asegurándote de que tu programa esté funcionando como corresponde, entonces encontrarás el bug cuando llegues al punto en donde deje de funcionar. Una vez que hayas encontrado el bug, ¡podrás comenzar a repararlo (o "depurarlo")!

Si crees que emocionará y motivará a los estudiantes, podrías presentar al personaje que participará en los desafíos de hoy, *Scrat*, de "La era de hielo". Si los estudiantes no conocen a *Scrat*, muéstrales algunos videos de la extravagante ardilla metiéndose en problemas.

## Actividad Principal (30 minutos)

### Depuración en el laberinto

Antes de que los estudiantes comiencen a trabajar en los computadores, recuérdales las ventajas de la programación en parejas y de pedir ayuda a sus compañeros. Ubica a los estudiantes en parejas y recomiéndales que, frente a alguna duda o problema, consulten con al menos dos compañeros antes de ir con el docente.



Vídeo: Depuración con el Botón Step



Desarrollo de Habilidades

2

3

4

5

6

7



Desafío



Predicción



Práctica



Lecciones adicionales



Como se mencionó en el propósito de esta lección, asegúrate de que los estudiantes estén conscientes de que se enfrentarán a desafíos muy frustrantes. Coméntales que está bien sentirse frustrados, pero que es muy importante buscar cómo solucionar el problema y pedir ayuda. Mientras los estudiantes realizan los desafíos, recorre el salón para asegurarte de que ningún estudiante se sienta tan frustrado como para no querer continuar con la actividad.

## Actividad de cierre (5 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué clase de bugs encontraste hoy?
- Dibuja un bug que hayas encontrado en un desafío hoy. ¿Qué hiciste para “depurar” el programa?

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### Plantando bugs

Pide a los estudiantes que vuelvan a desafíos de lecciones anteriores y, a propósito, añadan bugs a sus soluciones. Luego, haz que pidan a un compañero que depure el programa. Esto también puede realizarse con desafíos en papel.

Mientras los estudiantes estén depurando, asegúrate de que las críticas sean constructivas. Si esto pudiese ser un problema para tu clase, repasa cómo depurar de forma respetuosa antes de esta actividad, a través de un juego de roles con otro estudiante.



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 5: Recoger tesoros con Laurel

50 minutos

## Resumen

En esta serie de desafíos, los estudiantes continuarán desarrollando su comprensión de los algoritmos y la depuración. Crearán algoritmos secuenciales para que el nuevo personaje Laurel la Aventurera pueda recoger el tesoro mientras recorre el camino.

## Propósito

En esta lección, los estudiantes pondrán en práctica sus habilidades de programación con un nuevo personaje, Laurel, la aventurera. Cuando se comienza a programar, se juntan instrucciones en un orden específico, a través de un medio que una máquina pueda leer. Por medio de la programación, los estudiantes comenzarán a entender cómo un computador navega a través de instrucciones y órdenes. Al usar un nuevo personaje y al resolver desafíos con nuevos y diferentes objetivos, los estudiantes ampliarán su abanico de experiencias con secuenciación y algoritmos en la programación.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (5 minutos)

#### Introducción

### Actividad puente (10 minutos)

#### Revisión de desafíos en línea en conjunto

### Actividad Principal (30 minutos)

#### Recoger tesoros con Laurel

### Actividad de cierre (5 minutos)

#### Reflexión

## Objetivos

Los estudiantes podrán:

- Desarrollar habilidades de resolución de problemas y pensamiento crítico al revisar prácticas de depuración.
- Ordenar comandos de movimiento en pasos secuenciales dentro de un programa.
- Explicar un algoritmo como un programa de computador.

## Preparación

- Realiza los desafíos para encontrar cualquier área potencialmente problemática para tu clase.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**iAviso!** Por favor, haga una copia de cualquier documento que planee compartir con los estudiantes.

Para los estudiantes

- **Bloques sin conexión (cursos C-F)** - Manipulativos

▼ Hacer una copia

## Vocabulario

- **Algoritmo** - una lista de pasos para terminar una tarea.

- **Programa** - un algoritmo que se ha programado como algo que puede ser ejecutado por una máquina.
- **Programación** - El arte de crear un programa

## Guía Didáctica

### Actividad previa (5 minutos)

#### Introducción

Esta lección usa la mayoría de los bloques usados en la lección anterior, además de añadir la habilidad de `recolectar`. Comenta que este bloque le permitirá a *Laurel, la aventurera* recoger el tesoro en donde ella esté parada. Se conversará más acerca de este nuevo bloque en la actividad puente.

### Actividad puente (10 minutos)

#### Revisión de desafíos en línea en conjunto

Elige un desafío de la lección en línea. Recomendamos el desafío 7. Pide a los estudiantes que conversen sobre un patrón que pueda ayudar a *Laurel, la aventurera*, a recolectar todo el tesoro. Pídeles que compartan sus soluciones. ¡Mira cuántos estudiantes tienen las mismas respuestas!

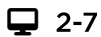
### Actividad Principal (30 minutos)

#### Recoger tesoros con Laurel

*Laurel, la aventurera*, busca recolectar tantos tesoros como sea posible. Indica a los estudiantes que recorran el desafío para recolectar todo lo que puedan. Algunos niveles requerirán que sólo se recoja un tesoro, mientras que otros requerirán recolectarlos todos. ¡Atención a las instrucciones!



Vídeo: El Coleccionista



Desarrollo de Habilidades

2

3

4


5

6

7



Desafío

 9-11

Práctica

9

10

11

 12

Predicción

 13

Práctica

 14-15

Lecciones adicionales



## Actividad de cierre (5 minutos)

Reflexión

### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- Dibuja un laberinto que podrías resolver con los bloques que usaste hoy.



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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



# Lección 6: Crear arte con código

50 minutos

## Resumen

En esta lección, los estudiantes tomarán control del Artista para completar los dibujos en pantalla. En esta etapa del Artista, los estudiantes podrán crear imágenes de mayor complejidad mediante nuevos bloques, tales como como `avanzar 100 píxeles` y `girar 90 grados a la derecha`.

## Propósito

Basándose en la experiencia previa de los estudiantes con secuencias, esta lección buscará inspirar más creatividad al programar. El propósito de esta lección es reforzar el conocimiento en secuenciación al introducir nuevos bloques y objetivos. En este caso, los estudiantes aprenderán más sobre píxeles y ángulos, usando los nuevos bloques, mientras siguen desarrollando sus habilidades de secuenciación. Además, los estudiantes serán capaces de percibir nuevos objetivos, tales como programar al Artista para que dibuje un cuadrado.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (10 minutos)

#### Introducción

### Actividad Principal (30 minutos)

#### Crear arte con código

### Actividad de cierre (10 minutos)

#### Reflexión

### Ampliación del aprendizaje

### Oportunidad multidisciplinaria

## Objetivos

Los estudiantes podrán:

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

## Preparación

- Realiza los desafíos para encontrar cualquier área potencialmente problemática para tu clase.
- (Opcional) Consigue transportadores para tu clase, para que puedan ver los ángulos que deben usar para completar los desafíos.
- Imprime una **\*Guía de trabajo** para cada estudiante.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planee compartir con los estudiantes.

Para los estudiantes

- **Introducción a Artista** - Video del estudiante
- **Giros & Ángulos** - Guía para el estudiante
- **Giros & Ángulos** - Video del estudiante

# Guía Didáctica

## Actividad previa (10 minutos)

### Introducción

Muestra uno o ambos videos a continuación, a modo de introducción a los ángulos:

**\*Video - Introducción al Artista** (1:5 minutos de duración)

**\*Video - Giros y ángulos** (2 minutos de duración)

Usa la **\*Guía de trabajo** para enseñar las diferencias entre los ángulos interiores y exteriores de diferentes figuras. Puedes usar este documento como una guía de trabajo o puedes imprimirlo a modo de póster para que los estudiantes puedan recurrir a él en cualquier momento.

**Analiza** las formas cuadrada y triangular en el documento.

- *¿Cómo programarían a un computador para que dibujase esa forma?*
- *¿En qué orden deben estar las instrucciones?*

Comenta que, en estos desafíos, moverán a un personaje que deja una línea por donde pasa. Los estudiantes escribirán códigos para que el personaje dibuje varias formas, incluso un cuadrado.

## Actividad Principal (30 minutos)

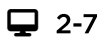
### Crear arte con código

En esta serie de desafíos, el artista ya no estará limitado a ángulos de 90 grados. Tener transportadores a la mano puede ser de gran ayuda para visualizar de mejor manera los ángulos que necesiten. Sin embargo, la actividad les entrega imágenes de los ángulos a los estudiantes, cada vez que seleccionen qué ángulo usar (importante: la selección de ángulos está limitada a dos dentro del menú desplegable, lo que reduce la cantidad de opciones que el estudiante tiene para utilizar).

Antes de pasar a los computadores, podría ser de gran ayuda hacer una breve presentación sobre cómo usar las herramientas en este nivel. Recomendamos el desafío 5, ideal para demostrar cómo funciona el transportador en línea.



Presentación del Artista con JR Hildebrand



Desarrollo de Habilidades

2

3

4

5

6

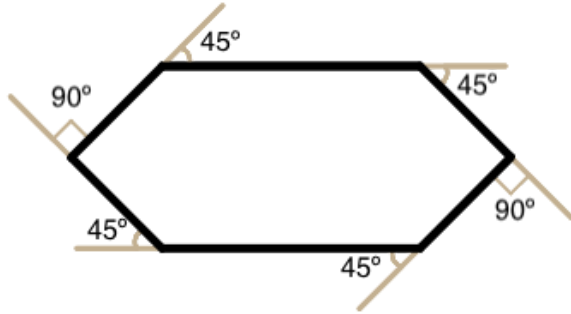
7



Desafío

### Consejo didáctico

El octavo desafío les pide dibujar un polígono de 6 lados. Esto podría ser desafiante para algunos estudiantes. Recomendamos que los estudiantes lo intenten unas cuantas veces, luego que pidan ayuda a un compañero y finalmente al docente. A continuación, una imagen que podría ser de gran ayuda para los estudiantes:



9

Práctica

10

Predicción

11-12

Lecciones adicionales



## Actividad de cierre (10 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Cuáles son los ángulos interiores para construir un cuadrado?, ¿y para un triángulo?
- Dibuja una forma simple e imagina el código necesario para dibujarla. ¿Puedes escribir ese código al lado de la forma?

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

### La fotocopiadora

- Entrega a los estudiantes dos hojas de papel.
- En una hoja, dibujen una imagen simple, usando sólo líneas rectas.
- En la segunda hoja, escriban instrucciones para recrear los comandos de dicha imagen, para moverse en línea recta y para girar en diferentes ángulos.
- Intercambien las hojas de instrucciones e intenten recrear la imagen usando sólo las instrucciones provistas.

## Oportunidad multidisciplinaria

Formas y paisajes (45-60 minutos)

 **Ciencias de la Computación + Lengua y Literatura + Matemáticas + Ciencias**

**Formas y paisajes** es una actividad opcional alineada con los estándares de Lengua y Literatura, Matemáticas y Ciencias de la Próxima Generación de Common Core, escrita por nuestra comunidad de docentes. Se pide a los estudiantes que diseñen una presa para evitar futuras inundaciones. Utilizando el código, crearán un plano para mostrar al ayuntamiento local cómo será su presa.

*Estándares abordados:*

- **CCSS.ELA-LITERACY.W.2.8:** recordar información de experiencias o reunir información de fuentes proporcionadas para responder a una pregunta.
- **CCSS.MATH.CONTENT.2.MD.A.3:** estimar longitudes usando unidades de pulgadas, pies, centímetros y metros.
- **NGSS.2-ESS2-1:** comparar diversas soluciones diseñadas para frenar o evitar que el viento o el agua cambien la forma de la tierra.



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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

# Lección 7: Mis amigos robóticos y repetitivos Jr.

45 minutos

## Resumen

Esta lección **context-setting** se basa en la actividad inicial “Mis amigos robóticos”, con diseños más grandes y complicados.

## Propósito

Esta lección sirve como una reintroducción a los bucles, usando el ya conocido conjunto de instrucciones para programar “robots”. Los estudiantes desarrollarán sus habilidades de pensamiento crítico al buscar patrones de repetición en los movimientos de sus compañeros y determinar cómo simplificar esos patrones repetidos a través de bucles.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (10 minutos)

#### Repaso de Mis amigos robóticos

### Actividad (30 minutos)

#### traducción pendiente: en-

US.data.unplugged.courseB\_unplugged\_loopyMRF\ 2018.title

### Actividad de cierre (5 minutos)

#### Reflexión

### Ampliación del aprendizaje

## Objetivos

Los estudiantes podrán:

- Identificar patrones repetidos en un código que pueden ser reemplazados por un bucle.
- Escribir instrucciones que usen bucles para repetir patrones.

## Preparación

- Asegúrate de que cada estudiante tenga su Diario de apuntes.
- Prepara una pila de 20 vasos desechables (o **\*trapezoides de papel**) para cada grupo de 4 estudiantes.
- Coloca los **\*símbolos** en donde los estudiantes puedan verlos y usarlos de referencia durante la lección.
- (Opcional) Imprime una serie de **\*Ideas para apilar vasos** para cada grupo de 4 estudiantes.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planee compartir con los estudiantes.

Para los estudiantes

- **Caras de sensaciones - Imagen de emociones** -

Recurso

- **Mis amigos robóticos** - Espacio entre vasos
- **Mis amigos robóticos** - Ideas para apilar vasos
- **Mis amigos robóticos** - Plantilla de trapecios en papel
- **Mis amigos robóticos** - Clave de símbolos

## Vocabulario

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

## Guía Didáctica

### Actividad previa (10 minutos)

#### Repaso de Mis amigos robóticos

**Objetivo:** este repaso refrescará la mente de los estudiantes y les recordará que los programas de la actividad “Mis amigos robóticos” pueden ponerse intensos muy rápidamente.

**Mostrar:** muestra la **\*Simbología** usada en la actividad “Mis amigos robóticos”. Pide a los estudiantes que te muestren cómo se vería un robot siguiendo las instrucciones, con cada uno de los cuatro símbolos.



Unplugged

# My Robotic Friends

Symbol Key (Course B)



Pick Up Cup



Put Down Cup



Step Forward



Step Backward

**Ilustrar:** en conjunto como clase, elijan uno de los modelos más sencillos de las **\*Ideas para apilar vasos** y programen todos juntos, con el propósito de recordar las reglas y la terminología.

A continuación, elige un modelo un poco más complejo, pero que requiera varios pasos para ser completado. Por ejemplo:



**Voluntario:** pide a un voluntario (o un grupo de voluntarios) que pase al frente y te ayude a programar este patrón en la pizarra. Si se apeg a la regla de “usar sólo símbolos que estén en la guía”, es probable que tarde un poco!

**Mostrar:** ahora, muestra esta otra imagen:



¿Cuál es la reacción de la clase?

**Sugerencia:** dales a los estudiantes la oportunidad de pensar formas más cortas de transmitir el código que van a crear (esta parte puede ser omitida si tus estudiantes comienzan a decir cosas como “avanzar 6 veces”, ya que esto llevaría a la discusión sobre cómo expresar “seis veces” con símbolos).

Una vez que los estudiantes hayan llegado a la idea de “repetir” códigos, dales el vocabulario correspondiente. Asegúrate de comentarles que, a menudo, el término “repetir algo” y “poner algo en bucle” se usan indistintamente.



## Actividad (30 minutos)

traducción pendiente: en-

US.data.unplugged.courseB\_unplugged\_loopyMRF\\_2018.title

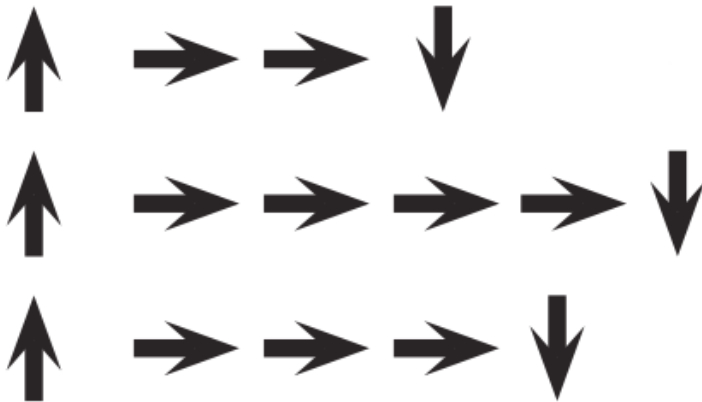
Actividad sin cone...

Mis Amigos Robóticos Loopy - Actividad Desconectada

### Introducción y ejemplo

**Preparación:** prepara pilas de vasos desechables o de trapezoides de papel para los grupos.

**Mostrar:** toma el programa de alguna de las torres de papel previamente realizadas y muéstralo a la clase. En su defecto, usa este:

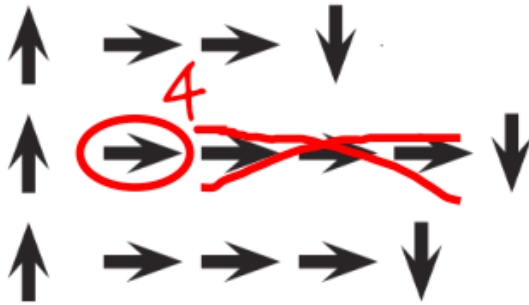


**Reflexión:** pide a los estudiantes que piensen en silencio y analicen en qué parte de este programa pueden encontrar un patrón de instrucciones que se repita sin interrupciones (una repetición tras otra).

**Interacción:** con el compañero del lado, compartan los patrones repetidos que encontraron.

**Exposición:** pide a algunos estudiantes que compartan con la clase los patrones que hayan encontrado. Intenta mostrar diferentes perspectivas para agrupar patrones. Pídeles identificar cuántas veces se repite cada patrón.

**Ilustrar:** en uno de los patrones identificados por la clase, demuestra cómo encerrar en un círculo la instrucción o el patrón encontrado, escribir el número de veces que se repite cerca del círculo y tachar con una línea el resto de las flechas.



Repita esto hasta que todo el programa haya sido simplificado. Luego, reescribe el programa de manera que los estudiantes puedan notar cuánto más simple son las instrucciones de esta manera.

## Robots en bucles

**Grupo:** divide a los estudiantes en grupos de a cuatro. Cada grupo debe ser, además, separado en dos parejas. Cada pareja desarrollará su propio programa y lo ejecutará en la otra pareja del mismo grupo.

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

**Mostrar:** muestra las **\*Ideas para apilar vasos** a la clase o imprime copias para cada grupo. Haz que cada pareja (no grupo) elija qué torre les gustaría que su robot construyese. Esta vez, alienta a las parejas a elegir modelos más complicados.

**Análisis:** permite que los grupos discutan sobre cómo deberían ser construidas las torres. Luego, indica a cada grupo que deben convertir ese algoritmo en símbolos. Asegúrate de que cada grupo escriba los símbolos del algoritmo en un lugar en donde, luego, el “robot” pueda verlos. Mientras los estudiantes trabajan en sus programas, recuérdales estar alertas a la oportunidad de reemplazar con un bucle un patrón repetitivo.

### Consejo didáctico

**Buscando bucles:** asegúrate de estar atento a que los estudiantes utilicen bucles. Intenta evitar corregir sus algoritmos en general o recomendar una solución, pero siéntete libre de indicarles patrones que podrían acortarse gracias al uso de bucles.

Observa a los estudiantes mientras desarrollan sus códigos. ¿Hay algún error? Usa las preguntas de depuración para ayudarlos a encontrar una solución.

- ¿Qué hace?
- ¿Qué debería hacer?
- ¿Qué te dice eso?
- ¿Funciona en el primer paso?
- ¿Funciona en el segundo paso?
- ¿Dónde deja de funcionar?

**Acción:** cuando las parejas hayan terminado de escribir sus instrucciones, deben intercambiarlas dentro del grupo para ejecutar los códigos de los demás. Recuerda a los estudiantes estar atentos a los errores, pero que no interrumpan al robot hasta que termine de ejecutar el programa completo.

**Análisis:** una vez que todas las parejas hayan tenido la oportunidad de ejecutar los programas, pídeles a algunos que compartan sus soluciones con la clase. Aprovecha esta oportunidad para conversar sobre cómo los grupos llegaron a diferentes soluciones para el mismo desafío. Por cada programa, podrías preguntar en particular:

- *¿Cómo identificaron los bucles?*
- *¿Hay alguna otra forma en que podrían haber escrito esos bucles?*
- *¿Cuánto más corto es el programa con bucles que sin ellos?*
- *¿Es más fácil entender un programa con instrucciones escritas en bucle o escritas una a una? ¿Por qué?*

## Actividad de cierre (5 minutos)

### Reflexión

#### Sugerencias:

- Haz que los estudiantes escriban o dibujen en sus diarios algo que les recuerde qué son los bucles. Para esto, podrías preguntar:
  - ¿Qué significa “repetir” para ti?
  - Haz un dibujo de ti repitiendo algo.

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

- Pide a los estudiantes que dibujen sus propias torres de vasos para que alguien más las programe.
- Entrégales algoritmos que usen repeticiones; luego, haz que expandan el programa en una versión paso a paso.



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 8: Bucles con Rey y BB-8

50 minutos

## Resumen

En base al concepto de repetición de instrucciones de "Getting Loopy", en esta etapa los estudiantes usarán bucles para ayudar a BB-8 a atravesar un laberinto de manera más eficiente que antes.

## Propósito

En esta lección, los estudiantes aprenderán más sobre los bucles y cómo implementarlos con un código Blockly. El uso de bucles en programación es una habilidad importante, ya que repetir comandos manualmente es tedioso e ineficiente. Con los desafíos de Code.org, los estudiantes aprenderán a añadir instrucciones a un bucle existente, agrupar en bucles los códigos repetidos y reconocer patrones que necesitan estar en bucle. Es importante destacar que los estudiantes se enfrentarán a desafíos que pueden tener varias soluciones diferentes. Esto abrirá el debate sobre las diferentes formas de resolver un desafío y sobre las ventajas y desventajas de cada método.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (5 minutos)

#### Introducción

### Actividad puente: elige una (10 minutos)

#### Actividad sin conexión: bloques de papel

#### Revisión de desafíos en línea en conjunto

### Actividad Principal (30 minutos)

#### Bucles con Rey y BB-8

### Actividad de cierre (5 minutos)

#### Reflexión

### Ampliación del aprendizaje

## Objetivos

Los estudiantes podrán:

- Descomponer una secuencia larga de instrucciones en una secuencia de repetición más corta.
- Utilizar una combinación de comandos secuenciales y en bucle para llegar al final de un laberinto.
- Identificar los beneficios de usar una estructura de bucle en lugar de la repetición manual.

## Preparación

- Realiza los desafíos para encontrar cualquier área potencialmente problemática para tu clase.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los estudiantes

- **Bloques sin conexión (cursos C-F)** - Manipulativos
  - ▼ Hacer una copia
- **Programación en parejas** - Vídeo

## Vocabulario

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

## Guía Didáctica

### Actividad previa (5 minutos)

#### Introducción

En conjunto con los estudiantes, repasa la actividad “Mis amigos robóticos y repetitivos”:

- *¿Qué son los bucles?*
- *¿Para qué los usamos?*

Brevemente, muéstrales un programa con instrucciones repetidas. Pídeles que encuentren los bucles en el programa y los señalen.

Si te parece una buena idea, presenta a *BB-8* de la saga *La guerra de la galaxias*. Muchos niños podrían estar familiarizados con el adorable droide; sin embargo, una presentación de seguro será emocionante.

### Actividad puente: elige una (10 minutos)

Esta actividad ayudará a los estudiantes a conectar los conceptos sin conexión de “Mis amigos robóticos y repetitivos” con el mundo digital al cual se están sumergiendo. Elige *una* de las siguientes actividades para realizar con tu clase:

#### Actividad sin conexión: bloques de papel

Retoma “Mis amigos robóticos y repetitivos”. Esta vez, trabaja con la clase para “programar” usando los **\*manipulativos\*** en vez de escribir las instrucciones en una hoja. Asegúrate de que los estudiantes sepan que los bloques deben estar ordenados desde arriba hacia abajo, y que deben estar en contacto!

-O-

#### Revisión de desafíos en línea en conjunto

Elige uno de los desafíos en línea para realizar frente a la clase. Recomendamos el desafío 10, dado su patrón de escalones. Pide a los estudiantes que escriban en una hoja un programa para resolver el desafío. Luego, pídeles que encierren en un círculo las partes repetidas y las marquen con el número de repeticiones, de la misma forma en que lo hicieron en “Mis amigos robóticos y repetitivos”.

### Actividad Principal (30 minutos)

Bucles con Rey y BB-8

 1 Vídeo: Programación con Rey y BB-8

 2 Desarrollo de Habilidades

 3 Predicción

 4 Vídeo: Bloques Repetir con BB-8

 5-9 Desarrollo de Habilidades

5 6 7 8 9

 10 Desafío

 11-12 Práctica

11 12

 13 Predicción

 14 Práctica

 15-16 Lecciones adicionales

Mientras los estudiantes trabajan en los desafíos, mira si logran notar la diferencia entre la cantidad de bloques que necesitan cuando usan bucles y la cantidad de bloques que necesitan cuando no los usan. La **\*programación en parejas** funciona muy bien con esta clase de desafíos, dado que hay pocas formas de llenar los bucles. Frente a un desacuerdo sobre cómo resolver un desafío, propone una discusión amistosa entre las partes. Aliéntalos a preguntarse entre ellos:

- *¿Cómo llegaste a esa solución?*
- *¿Cuáles son los beneficios de resolver el desafío de esa forma?*

También recomendamos tener una hoja a mano, para que los estudiantes puedan escribir sus códigos y encontrar cualquier repetición para poner en bucle.

## Actividad de cierre (5 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Cómo fue que los bucles hicieron que tu programa fuese más fácil de escribir?
- Piensa en algo que se repita una y otra vez. ¿Cómo se vería un programa para eso?

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### ¡A moverse!

- Dale a los estudiantes imágenes de acciones o pasos de baile que puedan realizar.
- Pídeles que realicen los pasos de baile y añadan bucles para crear sus propias coreografías.
- Comparte los bailes con el resto de la clase.

#### Vuelve a conectarlo

- Busca algunos videos en YouTube de bailes populares que sean repetitivos.
- ¿Los estudiantes pueden descifrar el bucle?
- ¡Intenta lo mismo con canciones!



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 9: Cosechar cultivos con bucles

45 minutos

## Resumen

Los alumnos realizan nuevas acciones para ayudar a la recolectora a recoger múltiples verduras que crecen en grandes racimos.

## Propósito

En esta lección, los estudiantes usarán bucles para repetir otras acciones, como recolectar calabazas. Aparecerán nuevos patrones y los estudiantes usarán su creatividad y lógica para determinar cuál código necesita ser repetido y cuántas veces.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

► **AP** - Algorithms & Programming

## Agenda

### Actividad previa (5 minutos)

#### Introducción

### Actividad Principal (30 minutos)

#### Cosechar cultivos con bucles

### Actividad de cierre (10 minutos)

#### Reflexión

### Ampliación del aprendizaje

### Oportunidad multidisciplinaria

## Objetivos

Los estudiantes podrán:

- Emplear una combinación de comandos secuenciales y en bucle para moverse y realizar acciones.
- Identificar cuándo puede usarse un bucle para simplificar una acción repetitiva.
- Escribir un programa que repita en bucle un comando simple para una tarea en particular.

## Preparación

- Juega con los rompecabezas para encontrar cualquier área potencialmente problemática para tu clase.
- Asegúrate de que cada estudiante tenga un diario de reflexión.

## Vocabulario

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



# Guía Didáctica

## Actividad previa (5 minutos)

### Introducción

Hasta ahora, los estudiantes han usado bucles sólo para resolver desafíos en laberintos, enfocados en repetir instrucciones de movimiento. En esta etapa, los estudiantes deberán usar los bucles para recoger varios objetos en un mismo punto.

#### Pregunta:

- *¿Qué otras partes de nuestro programa podrían beneficiarse del uso de bucles?*
- *En los desafíos de cosecha podría haber muchos objetos en un mismo punto. ¿Cómo podríamos usar los bucles para recoger una cantidad determinada de objetos, usando sólo dos bloques?*

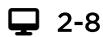
## Actividad Principal (30 minutos)

### Cosechar cultivos con bucles



Vídeo: La Cosechadora

Cuando los estudiantes usen bucles para repetir una acción (como cosechar calabazas), alientos a pensar en los movimientos anteriores y posteriores a dicha acción. Estas acciones ¿podrían ser incluidas en el bucle?

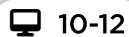


Desarrollo de Habilidades

2 3 4 5 6 7 8



Desafío



Práctica

10 11 12



Predicción



## Actividad de cierre (10 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- Da dos ejemplos de bucles usados en tu código.
- ¿Qué otra cosa podría cosechar una cosechadora? Dibuja el bloque necesario para cosechar tu respuesta.

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### Cosechando útiles

- Permite que los estudiantes apilen útiles escolares en sus puestos (lápices, gomas, etc.).
- Haz que sus compañeros descifren cómo debería moverse una cosechadora para avanzar de pila en pila recolectando los objetos.
- Compartan los programas con el resto de la clase.

## Oportunidad multidisciplinaria

### La granja de la cosechadora (60-75 minutos)

#### Ciencias de la Computación + Lengua y Literatura + Matemáticas + Ciencias

**\*\*La granja de la cosechadora\*\*** es una actividad opcional alineada con los estándares de Lengua y Literatura, Matemáticas y Ciencias de la Próxima Generación de Common Core, escrita por nuestra comunidad de docentes. Los estudiantes utilizarán los organizadores gráficos proporcionados para clasificar y describir los elementos de la granja de la cosechadora. Los organizadores gráficos servirán como componente de pre-escritura mientras los estudiantes escriben un texto informativo/explicativo para informar a los lectores sobre lo que hace la cosechadora en su granja.

#### *Estándares abordados:*

- CSCS.ELA-LITERACY.W.2.2:\*\* escribir textos informativos/explicativos en los que introduzcan un tema, utilicen hechos y definiciones para desarrollar puntos, y proporcionen una declaración o sección de conclusión.
- CCSS.ELA-LITERACY.W.2.5:\*\* con la orientación y el apoyo de los adultos y los compañeros, se centran en un tema y fortalecen la escritura según sea necesario mediante la revisión y la edición.

- CSS.MATH.CONTENT.2.OA.B.2:\*\* sumar y restar con fluidez dentro de 20 usando estrategias mentales.
- **NGSS.2-PS1-1:** planear y conducir una investigación para describir y clasificar diferentes tipos de materiales por sus propiedades observables.



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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

# Lección 10: Mini-Proyecto: Arte con Stickers

60 minutos

## Resumen

Esta lección de **miniproyecto** se basa en la comprensión de los bucles de las lecciones anteriores. Los estudiantes crearán una obra de arte única con el Artista.

## Propósito

Esta lección resalta el poder de los bucles para crear diseños personales y creativos.

Propuesta como una actividad secuencial respaldada por proyectos, esta etapa les permitirá a los estudiantes construir sobre su propio trabajo y crear materiales sorprendentes.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (15 minutos)

#### Introducción

### Actividad Principal (30 minutos)

#### Arte con stickers y bucles

### Actividad de cierre (15 minutos)

#### Reflexión

### Oportunidad multidisciplinaria

## Objetivos

Los estudiantes podrán:

- Diferenciar entre comandos que deben ser repetidos en bucle y comandos que deben ser usados de forma casual.
- Identificar los beneficios de usar una estructura de bucle en lugar de la repetición manual.

## Preparación

- Practica creando tu propio diseño. Presta atención a cómo estos niveles son diferentes de todo lo anterior hecho.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Vocabulario

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

# Guía Didáctica

## Actividad previa (15 minutos)

### Introducción

A este punto, los estudiantes ya deben haber tenido varias introducciones a los bucles. Basado en lo que crees que podría ayudar más a tus estudiantes, recomendamos:

- Crear un nuevo baile con bucles, similar a “girar y girar”.
- Como clase, realizar uno de los desafíos de la lección “bucles con Rey y BB-8”.
- Repasar cómo jugar con el Artista, realizando algún desafío de la lección “Programación en el Artista”.
- Revisar un desafío de esta lección.

Con estas opciones repasarás bucles o niveles artísticos. ¡Ambas opciones prepararán a tus estudiantes para divertirse con estos desafíos en línea!

## Actividad Principal (30 minutos)

### Arte con stickers y bucles

¡Fomenta la creatividad durante esta lección! Las instrucciones dan ejemplos; sin embargo, no hay problema con que los estudiantes experimenten con diferentes stickers o tipos de diseños. De hecho, ¡los ejemplos que damos no son las *únicas* soluciones!

#### 💡 Consejo didáctico

Esta lección podría sentirse muy diferente a lo anteriormente visto en el curso. Estos niveles tienen características nuevas que quizás quieras explorar antes de empezar. El código que escriban los estudiantes en un nivel será transferido automáticamente a los otros. Esto les permite construir gradualmente y replantearse sus ideas mientras aprenden. Nota que en estos niveles no hay respuestas correctas o incorrectas; de esta manera se fomenta una creatividad sin límites. Anima a tus estudiantes a determinar, por ellos mismos, cuándo han completado cada tarea.

### Miniproyecto: arte con stickers

🖥️ 1-8

Miniproyecto: Arte con Pegatinas

1

2

3

4

5

6

7

8

🖥️ 9

Juego libre

Para algunos estudiantes, identificar en dónde agregar bucles de repetición podría ser más fácil si escriben primero el programa sin bucles y luego encierran en un círculo las secciones que se repitan. De ser así, podría ser de gran ayuda para ellos permitirles tener hojas y lápices junto a los dispositivos. Los

estudiantes también podrían disfrutar dibujar algunas formas y figuras en una hoja antes de programarlas en internet (para dibujar estampas, podría ser más sencillo representarlas con formas simples, como círculos y cuadrados).

## Actividad de cierre (15 minutos)

### Reflexión

#### Sugerencias:

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

## Oportunidad multidisciplinaria

### Formas en bucle y sus funciones (45-60 minutos)

#### Ciencias de la Computación + Lengua y Literatura + Matemáticas + Ciencias

**\*\*Formas en bucle y sus funciones\*\*** es una actividad opcional alineada con los estándares de Lengua y Literatura, Matemáticas y Ciencias de Próxima Generación de Common Core, escrita por nuestra comunidad de docentes. El artista tiene que ayudar a sus amigos a diseñar su casa dibujando los planos. Es importante que los estudiantes comprendan las formas básicas y cómo cada forma cumple diferentes funciones en los planos de la casa. A continuación, escribirán artículos de opinión, aportarán razones y utilizarán palabras de enlace para justificar sus elecciones de diseño.

#### *Estándares abordados:*

- **CSCS.ELA-LITERACY.W.2.1:** escribir artículos de opinión en los que presenten el tema o el libro sobre el que están escribiendo, expongan una opinión, proporcionen razones que apoyen la opinión, utilicen palabras de enlace (por ejemplo, porque, y, también) para conectar la opinión y las razones, y proporcionen una declaración o sección de conclusión.
- **CCSS.MATH.CONTENT.2.G.A.1:** reconocer y dibujar formas que tengan atributos específicos, como un número determinado de ángulos o un número determinado de caras iguales.1 Identificar triángulos, cuadriláteros, pentágonos, hexágonos y cubos.
- **NGSS.K-2-ETS1-2:** desarrollar un boceto simple, un dibujo o un modelo físico para ilustrar cómo la forma de un objeto ayuda a que funcione como se necesita para resolver un problema dado.



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 11: El gran evento

45 minutos

## Resumen

En esta lección **context-setting**, los estudiantes aprenderán que los eventos son una excelente manera de agregar flexibilidad a un algoritmo preescrito.

## Propósito

Hoy, los estudiantes aprenderán a distinguir los eventos de las acciones. Los estudiantes verán cómo una serie de actividades son interrumpidas al apretar un “botón” en un control remoto de papel. Al ver este evento, la clase reaccionará con una acción en especial. Los eventos son muy usados en la programación y deberían ser fácilmente reconocidos luego de esta lección.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (15 minutos)

Una serie de eventos

Vocabulario

### Actividad Principal (15 minutos)

El gran evento

### Actividad de cierre (15 minutos)

Reflexión

Evaluación

### Ampliación del aprendizaje

## Objetivos

Los estudiantes podrán:

- Practicar el diferenciar acciones preestablecidas y acciones disparadas por eventos.
- Reconocer acciones del docente como una señal para iniciar comandos.
- Repetir comandos dados por un instructor.

## Preparación

- Imprime o proyecta el **\*control de eventos**.
- Imprime una **\*evaluación** para cada estudiante.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planee compartir con los estudiantes.

Para los profesores

- **El gran evento** - Guía de respuestas de la evaluación

Para los estudiantes

- **El gran evento** - Evaluación
- **El gran evento** - Video sin conexión (**Descarga**)
- **El gran evento (Curso C)** - Controlador de eventos

## Vocabulario

- **Evento** - una acción que hace que algo suceda.

## Guía Didáctica

### Actividad previa (15 minutos)

Una serie de eventos

**Comenta:** *Veamos si podemos encontrar un patrón aquí:*

- *Cuando pulsas un interruptor, las luces se encienden.*
- *Cuando tocas un dispositivo, se inicia una app.*
- *Cuando suena la alarma, te levantas de la cama.*
- *Cuando (evento), (acción).*

*En ciencias de la computación, los eventos provocan otras acciones. Nuestra nueva palabra de vocabulario de hoy es evento. Normalmente, cuando oímos "evento", pensamos en:*

- Excursión
- Competición
- Fiesta de cumpleaños

*Hoy vamos a aprender un nuevo significado para la palabra "evento". Vamos a centrarnos en los "eventos" que provocan otras acciones, como cuando al pulsar un interruptor se encienden las luces o al pulsar un botón se mueve un personaje en un juego.*

## Vocabulario

Esta lección tiene una nueva e importante palabra:

**Evento:** *díganlo conmigo, E - ven - to.*

Una acción que causa que algo pase.

### Actividad Principal (15 minutos)

El gran evento

Actividad sin cone...

El Gran Evento - Actividad Desconectada

- *¿Recuerdan haber ayudado a Red, de "Angry Bird", a encontrar al cerdo en los desafíos de laberinto?*
  - *En ese ejercicio, ustedes sabían de antemano exactamente a dónde querían que llegase Red, así que pudieron crear un programa que llevase al ave desde el inicio hasta el final sin ninguna interrupción.*



- En la mayoría de los programas reales no podemos hacer eso, porque queremos tener opciones, dependiendo de las necesidades del usuario.
  - Quiero que mi personaje se mueva solamente cuando toque la pantalla con mi dedo. Entonces, tendría que programar al personaje para que se mueva sólo cuando toque la pantalla con mi dedo.
  - Tocar la pantalla con mi dedo sería un "evento" que le dice a mi personaje que debe moverse.

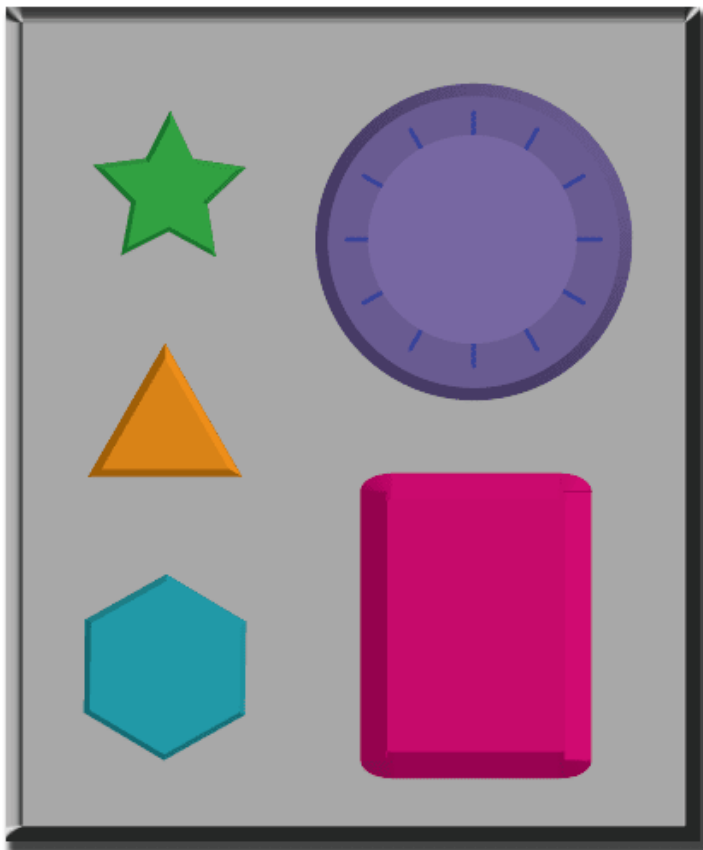
En lecciones anteriores, desarrollamos algoritmos que nos permitían controlar a un amigo o a un ave, varios pasos a la vez. Fue divertido y muy útil, pero, ¿qué pasa si no sabes de antemano todo lo que quieres que haga tu amigo? ¡Ahí entran los eventos!

#### 💡 Consejo didáctico

Si los estudiantes parecen confundidos, háldeles de sus juegos favoritos y de todas las formas en que le hacen saber al personaje del juego lo que deben hacer. Haz énfasis en lo aburrido que sería si el personaje sólo fuese desde el inicio hasta el final sin necesidad de ningún evento.

#### Instrucciones:

- Proyecta el **\*control de eventos** en la pantalla de tu salón.



- Decide con tu clase qué hará cada botón. Sugerimos:
  - Estrella verde -> Aplauso
  - Triángulo naranja -> Manos en el aire
  - Hexágono azul -> Manos en las caderas

- Círculo púrpura -> Giro
- Rectángulo rosa -> Pisotón
- Practica presionando los botones y que tu clase reaccione.
- Crea algunas secuencias con los botones y que los estudiantes intenten seguirlas con las acciones.
- Haz saber a tu clase que cada vez que presionas un botón, se trata de un evento que les permite saber lo que deben hacer.
- Para hacer las cosas más difíciles, pide a los estudiantes que comiencen a realizar una actividad en particular e interrúmpelos ocasionalmente presionando los botones. Sugerimos:
  - Contar hasta 10
  - Cantar alguna canción conocida para ellos
- Practica hasta que los estudiantes entiendan que deben realizar una acción por cada botón que se presiona.

## Actividad de cierre (15 minutos)

### Reflexión (10 minutos)

#### Sugerencias:

- ¿Qué aprendimos?
- ¿Cuáles son algunos ejemplos de eventos?

💡 Consejo didáctico ▲

Los estudiantes deben demostrar que comprenden que los eventos son algo que pueden provocar otras acciones. Recuerda los ejemplos anteriores de la lección o anima a los estudiantes a pensar sus propios ejemplos.

### Evaluación (5 minutos)

- Dale a cada estudiante una **\*Evaluación** y permíteles completarla individualmente luego de haber explicado claramente las instrucciones.
- Gracias a las actividades anteriores, esta actividad debería ser familiar para ellos.

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### El evento de uno es la reacción de otro

- Asigna a cada estudiante un evento al que deba estar atento y una reacción apropiada para dicho evento. Encadena las acciones de manera que la reacción de un estudiante sea el evento de otro. Asigna tantos eventos como sea necesario, hasta que todos tengan algo que hacer y todos hagan reaccionar a alguien más.

#### Eventopalooza.

- Esta actividad usa el control de eventos. Divide la clase en grupos y asigna a cada grupo una reacción diferente para el mismo botón. Haz esto para los tres botones y luego mira el caos!



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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

# Lección 12: Construye un juego Flappy

60 minutos

## Resumen

Esta lección combina **construcción de estudios** en torno a eventos con un **mini-proyecto** donde los estudiantes pueden construir su propio juego Flappy Bird. Al final de la lección, los alumnos podrán personalizar su juego cambiando las visuales o reglas.

## Propósito

Los eventos son muy comunes en los programas de computador. En esta lección, los estudiantes desarrollarán aún más su comprensión de los eventos, construyendo un juego *Flappy Bird*. Los estudiantes aprenderán a hacer que su personaje se mueva a través de la pantalla, hacer ruidos y reaccionar a los obstáculos en base a eventos iniciados por el usuario.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (10 minutos)

#### Introducción

### Actividad puente: elige una (10 minutos)

#### Actividad sin conexión: bloques de papel

#### Revisión de los desafíos en línea

### Actividad Principal (30 minutos)

#### Construye un juego Flappy

### Actividad de cierre (10 minutos)

#### Reflexión

### Ampliación del aprendizaje

### Oportunidad multidisciplinaria

## Objetivos

Los estudiantes podrán:

- Crear un juego usando el control de eventos.
- Relacionar bloques con el control de eventos apropiado.
- Compartir un artefacto creativo con otros estudiantes.

## Preparación

- Juega con los rompecabezas para encontrar cualquier área potencialmente problemática para tu clase.
- Asegúrate de que cada estudiante tenga un diario de reflexión.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los profesores

- **Pauta con ejemplos resueltos** - Guía de respuestas

▼ Hacer una copia

Para los estudiantes

- **Bloques sin conexión (cursos C-F)** - Manipulativos

▼ Hacer una copia

- **Momento de reflexión en línea** - Vídeo
- **El gran evento (Curso C)** - Controlador de eventos

## Vocabulario

- **Evento** - una acción que hace que algo suceda.

## Guía Didáctica

### Actividad previa (10 minutos)

💡 Consejo didáctico ▲

Los estudiantes tendrán la oportunidad de compartir el resultado final con un enlace. Esta es una gran oportunidad para mostrarle a la comunidad escolar las grandes cosas que tus estudiantes están logrando. Reúne todos los enlaces y guárdalos en la página web del curso, de manera que todos los puedan ver!

### Introducción

- Repasen la actividad “el gran evento”:
  - *¿Para qué “programamos” los botones de eventos?*
- *Ahora vamos a agregar eventos a nuestro código, específicamente, vamos a crear dos eventos: uno para cuando se haga clic con el mouse y otro para cuando el ave choque con algún objeto como obstáculos o el suelo. ¿Has visto que un personaje tocando algún objeto sea un evento en un juego?*

### Actividad puente: elige una (10 minutos)

Esta actividad los ayudará a traer los conceptos sin conexión de “El gran evento” hasta el mundo digital al cual los estudiantes se están sumergiendo. Elige una de las siguientes actividades para hacer con tu clase:

#### Actividad sin conexión: bloques de papel

Con el **\*Control de eventos** y los **\*Manipulativos**, reúne a tu clase para repetir la actividad de la lección anterior. Pregúntales: “Cuando presionamos el botón verde agua, ¿qué hacemos?”. Rellena uno de los bloques de evento `cuando` y uno de los bloques de acción azules con la acción establecida por la pregunta anterior. Asegúrate de que los estudiantes entiendan que los bloques `cuando` necesitan estar sobre el bloque azul y que necesitan estar en contacto, para que el programa funcione.

-O-

## Revisión de los desafíos en línea

Elige uno de los desafíos en línea de la lección. Recomendamos el desafío número 2. Pregunta qué debería pasar cuando el *Flappy Bird* vaya hacia algo como el suelo o un obstáculo. Explique que *Flappy*, en este juego, se moverá hacia adelante cuando hagan clic con el mouse y que el juego terminará si *Flappy* choca con cualquier objeto.

Completa el desafío con la clase y debatan brevemente sobre qué era y qué no era un evento. Pregunta cuál es la acción correspondiente a cada evento visto.

## Actividad Principal (30 minutos)

### Construye un juego Flappy

#### Desafíos en línea y juego libre

Toda la lección está enfocada a que los estudiantes creen sus propios juegos Flappy Birds. Bajo esta premisa, en la segunda mitad de la lección los niveles están todos vinculados. Esto significa que los estudiantes verán sus propios códigos representados en los desafíos 6 a 10, lo que les permitirá personalizarlos cada vez más, hasta llegar al resultado final.

En la etapa final de esta lección, los estudiantes podrán modificar sus juegos para hacerlos únicos. Motívalos a ver qué tan diferentes pueden hacer sus juegos dadas las limitaciones. Si la clase no programa en parejas, pídeles que miren los juegos de sus compañeros. De lo contrario, haz que los estudiantes analicen y prueben diferentes formas de armar sus juegos con sus parejas.

1

Vídeo: Codifica Tu Propio Juego Flappy

2-5

Desarrollo de Habilidades

2

3

4

5

6-11

Miniproyecto: Juego de Flappy

6

7

8

9

10

11

Consejo didáctico

Recuerda a los estudiantes que sólo deben compartir su trabajo con sus amigos o familiares cercanos. Para más información, visualiza o muestra a la clase el video **\*Momento para reflexionar en línea.**

## Actividad de cierre (10 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué hiciste para que tu juego fuese único?
- Dibuja un juego que te gustaría hacer en el futuro.

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### Mira detrás de la cortina

Cuando compartes un enlace de tu juego, también compartes todo el código que está detrás y que lo compone. Esta es una gran oportunidad para que los estudiantes aprendan uno del otro.

- Publica enlaces a juegos completos en línea o en la pizarra.
  - ¡Haz tú también un juego para compartir!
- Cuando los estudiantes carguen un enlace, haz que presionen el botón “Cómo funciona” para que vean el código detrás del juego.
- En grupos, conversen sobre las diferentes formas en que sus compañeros programaron sus juegos.
  - ¿Qué te llamó la atención?
  - ¿Qué te gustaría probar?
- Elige el juego de alguien más y juega con él (no te preocupes, el juego original está guardado).

## Oportunidad multidisciplinaria

### Conviértete en diseñador de juegos (45-135 minutos)

#### **Ciencias de la computación + Matemáticas + Ciencias**

**\*\*Conviértete en diseñador de juegos\*\*** es una actividad opcional alineada con los estándares de Matemáticas y Ciencias de la Próxima Generación de Common Core, escrita por nuestra comunidad de docentes. Los estudiantes se convertirán en diseñadores de juegos mientras crean y prueban un juego. Los estudiantes también registrarán las puntuaciones, harán observaciones, crearán un gráfico de barras, analizarán los resultados y justificarán los cambios que harían en su juego.

*Estándares abordados:*

- **2.MD.D.10:** dibujar un gráfico de imagen y un gráfico de barras (con escala de una unidad) para representar un conjunto de datos con hasta cuatro categorías.
- **K-2 ETS1-1:** hacer preguntas, hacer observaciones y reunir información sobre una situación que la gente quiere cambiar para definir un problema simple que puede ser resuelto a través del desarrollo de un objeto o herramienta nueva o mejorada.
- **K-2 ETS1-3:** analizar los datos de las pruebas de un objeto o herramienta para determinar si funciona según lo previsto y comparar los puntos fuertes y débiles de su rendimiento.



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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



# Lección 13: Mini-Proyecto: Juego de Persecución

55 minutos

## Resumen

Esta lección combina **construcción de estudios** en torno a eventos con un **mini-proyecto** donde los estudiantes pueden construir su propio juego animado.

## Propósito

En esta lección, los estudiantes desarrollarán aún más su comprensión de los eventos, a través de Play Lab. Los estudiantes usarán eventos para hacer que los personajes se muevan por la pantalla, hacer ruidos y cambiar el fondo en base a la configuración del usuario. Al final de los desafíos, los estudiantes tendrán la oportunidad de compartir sus proyectos.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (10 minutos)

#### Introducción

### Actividad Principal (30 minutos)

#### Chase Game con Eventos

### Actividad de cierre (15 minutos)

#### Reflexión

### Ampliación del aprendizaje

## Objetivos

Los estudiantes podrán:

- Crear un juego animado e interactivo usando secuencias y controladores de eventos.
- Identificar acciones que se correlacionan con eventos de entrada.

## Preparación

- Juega con los rompecabezas para encontrar cualquier área potencialmente problemática para tu clase.
- Asegúrate de que cada estudiante tenga un diario de reflexión.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los profesores

- **Niveles de programación abiertos** - Guía de respuestas

▼ Hacer una copia

Para los estudiantes

- **Bloques sin conexión (cursos C-F)** - Manipulativos

▼ Hacer una copia

- **Momento de reflexión en línea** - Vídeo

## Vocabulario

- **Evento** - una acción que hace que algo suceda.

## Guía Didáctica

### Actividad previa (10 minutos)

#### Introducción

Analicen brevemente el juego *Flappy Bird* de la lección anterior. Pide a los estudiantes que mencionen los diferentes eventos en el juego, entre los cuales se incluyen:

- *Flappy* golpeándose contra el suelo
- *Flappy* chocando con un obstáculo
- Un jugador haciendo clic en la pantalla
- *Flappy* esquivando un obstáculo

Ahora analicen las acciones que corresponden con esos eventos. Si *Flappy* choca con cualquier objeto se termina el juego, pero si esquivo un obstáculo, gana un punto. Un jugador haciendo clic en la pantalla provoca que *Flappy* mueva sus alas.

Pregunta qué otros eventos y acciones les gustaría ver. ¿Qué otra clase de juegos podríamos crear con estos eventos y acciones?

### Actividad Principal (30 minutos)

#### Chase Game con Eventos

💡 Consejo didáctico ▲

Los estudiantes tendrán la oportunidad de compartir el resultado final con un enlace. Esta es una gran oportunidad para mostrarle a la comunidad escolar las grandes cosas que tus estudiantes están logrando. Reúne todos los enlaces y guárdalos en la página web del curso, de manera que todos los puedan ver.

Recuérdales que sólo pueden compartir sus trabajos con sus amigos cercanos o familiares. Para más información, mira o muestra a la clase el video **\*Momento de reflexión en línea**.

Esta es la actividad en línea que ofrece más libertad de todo el curso. En la etapa final, los estudiantes tendrán la libertad de crear un juego por ellos mismos. Quizás quiera proporcionarles pautas estructuradas sobre qué clase de juegos crear, particularmente para aquellos estudiantes que podrían sentirse agobiados por tantas opciones.



Vídeo: Introducción al Laboratorio de Juegos



2-6

Desarrollo de Habilidades

2

3

4

5

6



7-12

Miniproyecto: Juego de persecución

7

8

9

10

11

12



13

Juego libre



14

Lecciones adicionales

## Actividad de cierre (15 minutos)

### Reflexión

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué evento usaste hoy en tu programa?
- ¿Hay algún evento que te hubiese gustado usar en tu juego y que no hayas podido usar en Play Lab?

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

#### Mira detrás de la cortina

Cuando compartes un enlace de tu historia, también compartes todo el código que está detrás y que la compone. Esta es una gran oportunidad para que los estudiantes aprendan uno del otro.

- Publica enlaces a historias ya terminadas en línea.
  - ¡También crea una historia para compartir!
- Cuando los estudiantes carguen un enlace, que hagan clic en el botón “Cómo funciona” para ver el código detrás de la historia.
- En grupos, conversen sobre las diferentes formas en que los compañeros de clase programaron sus historias.
  - ¿Qué te llamó la atención?
  - ¿Qué te gustaría probar a ti?
- Elige la historia de alguien más y haz clic en [Remix](#) para jugar con ella (no te preocupes, la historia original está guardada).



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 14: Gráficos de datos

45 minutos

## Resumen

Los estudiantes recopilarán datos de un proyecto de Play Lab y los visualizarán usando diferentes gráficos en esta lección de **exploratoria**.

## Propósito

Los computadores fueron creados para ayudar a procesar datos. Los datos están en un constante incremento en el mundo, por lo que ser capaz de leerlos y analizarlos es muy importante. En esta lección, los estudiantes vivirán una experiencia básica de recolección, visualización y análisis de un conjunto simple de datos.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **DA** - Data & Analysis

## Oportunidades multidisciplinarias

### Normas básicas comunes para matemáticas

- ▶ **MD** - Using Probability To Make Decisions

## Agenda

### Actividad previa (5 minutos)

La necesidad de visualizar

### Actividad Principal (35 minutos)

Gráficos de datos

### Actividad de cierre (5 minutos)

Reflexión

## Objetivos

Los estudiantes podrán:

- Recolectar y registrar datos sobre una cantidad de objetos reales o personajes en la pantalla.
- Crear un gráfico de barra y circular para representar datos simples.
- Comparar datos representados por otros y usarlos para hacer una predicción.

## Preparación

- Imprime una **\*guía de trabajo** para cada estudiante.
- Prueba la lección de hoy en Code Studio. Esta lección está diseñada para ser usada como una herramienta para la actividad de hoy. Prepárate para proyectarla frente a la clase, o en su defecto, permite que los estudiantes la vean desde sus propios computadores.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.

## Enlaces

**iAviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los profesores

- **Gráficos de datos de Play Lab** - Guía de trabajo

## Guía Didáctica

### Actividad previa (5 minutos)

La necesidad de visualizar

#### Observaciones

*En un momento, voy a pedirles que cada uno de ustedes me diga su estación favorita. A la cuenta de 3, todos van a decir su respuesta y vamos a intentar ver si podemos descifrar cuál es la estación más popular en este curso. ¿Ya decidieron cuál es su estación favorita? 1, 2, 3...*

Los estudiantes responderán todos al mismo tiempo.

**Análisis:** *¿Podemos saber cuál es la estación que a la mayoría de la clase le gusta? ¿Cómo podemos estar seguros?* Alienta a los estudiantes a justificar por qué creen en sus respuestas.

**Pregunta:** *¿Qué podemos hacer para estar más seguros de cuál es la estación favorita de la mayoría?*

Escuche las ideas de los estudiantes, pero guíalos hacia la idea de escribir los datos usando marcas de conteo en la pizarra o en una hoja.

#### Observaciones

*Las marcas de conteo nos ayudan a llevar registro de las respuestas de todos, lo cual es genial. Sin embargo, toma algo de tiempo contarlas para saber qué opinión fue elegida por el mayor y el menor número de estudiantes. Imaginen que alguien nos mostrase las marcas de conteo de toda la escuela. ¡Nos tomaría mucho tiempo contarlas todas!*

### Actividad Principal (35 minutos)

Gráficos de datos



Gráficos de datos

**Distribuir:** entrega las **\*guías de trabajo** junto con lápices y materiales para colorear de cualquier tipo. Indica a los estudiantes entrar al sitio de Code Studio o proyecta el nivel en la pizarra.

 Consejo didáctico ▲

Este programa genera una cantidad aleatoria de mascotas cada vez que se ejecuta. Si estás proyectando la página para toda la clase, considera ejecutar varias veces y pedirles a grupos de estudiantes que lleven registro de cada vez que el programa sea ejecutado. De esta manera, habrá

más variedad en las respuestas para una futura discusión.

Esta actividad está dividida en dos partes. En la primera página, los estudiantes usarán marcas de conteo para llevar registro de cuántas mascotas aparecen en el programa. En la segunda página, usarán una serie de gráficos para visualizar los datos que recolectaron.

**Interacción:** al terminar, alienta a los estudiantes a compartir sus resultados con un compañero y buscar similitudes y diferencias. Cuando toda la clase haya realizado la actividad, reagrupalos a todos.

**Análisis:** pide a la clase analizar y conversar sobre las siguientes preguntas:

- ¿Qué gráfico les muestra la información más útil? ¿Por qué?
- ¿Qué mascota fue la que más vieron? ¿Todos tienen la misma respuesta?
- ¿Qué creen que hubiese pasado si este programa mostrara 100 animales? ¿Y si mostrara 1000?
- ¿Qué preguntas pueden responder con sus datos?
- ¿Qué preguntas les gustaría poder responder?

El objetivo de este análisis es que los estudiantes reflexionen sobre la utilidad de los datos para identificar o responder preguntas, además de que empiecen a preguntarse por qué querían usar diferentes medios de visualización (en este caso, gráficos).

## Actividad de cierre (5 minutos)

### Reflexión

**Comenta:** Recolectar y graficar datos nos puede ayudar a preguntar y responder preguntas interesantes.

**Sugerencia:**

- ¿Cuál es una pregunta que te gustaría responder y qué datos deberías recolectar para responderla?

Por ejemplo, si juegas al fútbol, quizás te gustaría saber si los goles se hacen mayormente desde la izquierda, la derecha o el centro. ¿Qué datos deberías recolectar para responder esa pregunta? ¿Cómo los mostrarías?



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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

# Lección 15: Pulseras binarias

55 minutos

## Resumen

Esta lección de **exploratoria** introduce el concepto de binario que es como las computadoras de hoy almacenan todo tipo de información.

## Propósito

En esta lección, los estudiantes aprenderán cómo se representa la información de manera que un computador pueda interpretarla y almacenarla. Al aprender los valores binarios, los estudiantes tendrán la oportunidad de escribir códigos y compartirlos con sus compañeros como mensajes secretos. Así, se puede relacionar con la manera en que los computadores leen un programa, lo traducen a valores binarios, usan la información de alguna forma y responden de una manera en que los humanos podamos entender. Por ejemplo, cuando escribimos una oración en un documento y luego presionamos “guardar”, el computador traduce esa oración a binario, almacena la información y luego muestra un mensaje que indica que el documento ha sido guardado.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (15 minutos)

#### Introducción

### Actividad principal (20 minutos)

#### Pulseras binarias

### Actividad de cierre (5 minutos)

#### Reflexión

### Evaluación (15 minutos)

### Ampliación del aprendizaje

## Objetivos

Los estudiantes podrán:

- Decodificar binario a letras.
- Codificar letras en binario.
- Relacionar la idea de almacenar letras en un papel con la idea de almacenar información en un computador.

## Preparación

- (Opcional) Mira el video **\*Lección en acción**.
- Prepara plumones para las pulseras. Otras decoraciones, como cuentas o limpiapiipas, son opcionales.
- Imprime una **\*guía de trabajo** y una **\*evaluación** para cada estudiante.
- Asegúrate de que cada estudiante tenga su Diario de apuntes.
- (Opcional) Escribe un breve mensaje en binario en la pizarra.

## Enlaces

**¡Aviso!** Por favor, haga una copia de cualquier documento que planee compartir con los estudiantes.

Para los profesores

- **Pulseras binarias** - Guía de respuestas de la evaluación



- **Pulseras binarias** - Video - Lección en acción

Para los estudiantes

- **Pulseras binarias** - Guía de trabajo
- **Pulseras binarias** - Evaluación
- **Pulseras binarias** - Video sin conexión ([Descarga](#))
- **Bits versus Bytes** - Video del estudiante

## Vocabulario

- **Binario** - Una forma de representar información usando sólo dos opciones.

## Guía Didáctica

### Actividad previa (15 minutos)

#### Introducción

#### Vocabulario

Esta lección introduce una nueva e importante palabra:

**Binario:** *díganlo conmigo, BII - NAA - RIO.*

Una forma de representar información usando sólo dos opciones.

#### Encendido y apagado

- Si escribiste un breve mensaje en binario en la pizarra, pide a los estudiantes que le presten atención y pregunta si alguien sabe qué es o qué significa.
  - Deja el mensaje a un lado y procede a preparar la actividad.
- Puedes comenzar preguntándole a la clase si alguna vez han visto un computador por dentro.
  - ¿Qué hay dentro?
  - Esta es una buena oportunidad para mostrarles el interior de un computador (o imágenes).



## 🎤 Observaciones

- Los cables transportan información, en forma de electricidad, dentro de la máquina.
  - Las dos opciones que el computador usa respecto a esta información eléctrica son “on” y “off”. “Encendido” y “apagado”, icómo las luces de la sala!
    - Cuando un computador representa información usando sólo dos opciones, a ese medio de representación de información se le llama “binario”.
  - La modalidad de dos opciones continua incluso cuando la información llega a su destino.
- Los computadores también *almacenan* información usando este sistema binario.
  - Binario no siempre es “off” y “on”.
    - Las unidades de disco duro almacenan información usando magnetismo positivo y magnetismo negativo.
    - Los DVD almacenan información como reflectiva o no reflectiva.
  - ¿Cómo creen que podríamos convertir cosas de la vida real que queremos almacenar en un computador, en binario?
    - Empecemos con letras.
    - Usa la **\*guía de trabajo** para mostrar cómo un computador podría representar las letras mayúsculas.
      - Este es un buen momento para mencionar que cada espacio donde hay una opción binaria se llama “dígito binario” (Binary digit en inglés), o “bit”.
      - Pregunta si alguien sabe cómo se llama un grupo de 8 bits (se llama byte)
      - Dato curioso: un grupo de 4 bits se llama nibble.
      - Miren el video **\*Bits versus Bytes** (1 minuto de duración)
    - Revisa algunos ejemplos de letras escritas en binario; luego regresa a la lección.

- A continuación, escribe una letra codificada en binario y dale a la clase unos segundos para que descifren de qué letra se trata.
- Una vez que la clase logre descifrar la letra codificada, estarán listos para realizar la actividad.

## Actividad principal (20 minutos)

### Pulseras binarias

Actividad sin cone...

Pulseras Binarias - Actividad Desconectada

💡 Consejo didáctico ▲

Tú eres quien mejor conoce a tu clase. Como docente, decide si los estudiantes deben hacerlo individualmente o si deben trabajar en parejas o en pequeños grupos.

### Guía de trabajo – Pulseras binarias

No es necesario cubrir todo el sistema **binario**, como contar y convertir números hacia y desde decimales. Esta lección está pensada para ser una introducción divertida a cómo los computadores almacenan información, no una lección frustrante de fundamentos.

Instrucciones:

- Encuentra la primera letra de tu nombre en la guía de la actividad.
- Rellena los cuadrados en blanco de la pulsera, replicando el patrón de cuadrados que esté al lado de la letra que elegiste.
- Recorta la pulsera.
- ¡Colócala en tu muñeca para lucirla! (puedes pegarla con cinta adhesiva).
- Comparte tu pulsera con el resto de la clase para ver si logran descifrar cuál es tu letra.

A	■□■ ■■■□	N	■□■ ■■■■
B	■□■ ■■■■	O	■□■ ■■■■
C	■□■ ■■■■	P	■□■ ■■■■
D	■□■ ■■■■	Q	■□■ ■■■■
E	■□■ ■■■■	R	■□■ ■■■■
F	■□■ ■■■■	S	■□■ ■■■■
G	■□■ ■■■■	T	■□■ ■■■■
H	■□■ ■■■■	U	■□■ ■■■■
I	■□■ ■■■■	V	■□■ ■■■■
J	■□■ ■■■■	W	■□■ ■■■■
K	■□■ ■■■■	X	■□■ ■■■■
L	■□■ ■■■■	Y	■□■ ■■■■
M	■□■ ■■■■	Z	■□■ ■■■■

#### 💡 Consejo didáctico

Si la clase tiene presupuesto para materiales extras, intenta realizar esta actividad usando hilo (o limpiapiipas) y cuentas para crear las pulseras binarias, en lugar de plumones y papel. Pueden utilizar cualquier combinación de dos colores, pero blanco y negro tiende a ser la combinación más sencilla, dada la forma en que está representada la clave.

Al final de la actividad, retoma el mensaje que escribió en la pizarra al comienzo y mira si los estudiantes pueden descifrarlo aplicando lo que han aprendido.

## Actividad de cierre (5 minutos)

### Reflexión

#### Charla rápida: ¿qué aprendimos?

- ¿Qué más crees que es representado en forma binaria dentro de un computador?
- ¿De qué otra forma podrías representar un sistema binario, en lugar de usar cuadrados blancos y negros?
- ¿Cuál fue tu parte favorita de la actividad?

#### Sugerencias:

- ¿Sobre qué se trataba la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- Usa la guía de trabajo para escribir el resto de tu nombre o tu palabra favorita en binario.
- Imagina un mundo en donde habláramos en binario, diciendo sólo “on” y “off”, nada más. Dibuja dos personajes intentando conversar en binario.

## Evaluación (15 minutos)

- Distribuye las **\*evaluaciones** y permite que los estudiantes las contesten de forma individual, luego de haber explicado claramente las instrucciones.
- Gracias a la actividad anterior, esta actividad debería ser familiar para ellos.

## Ampliación del aprendizaje

Usa estas actividades para ampliar el aprendizaje de los estudiantes. Se pueden usar como actividades extraprogramáticas.

### Imágenes binarias

- Hay muchos recursos en internet para llevar esta actividad al siguiente nivel.
- Si tus estudiantes están interesados en saber cómo se pueden representar imágenes o canciones en binario, puede encontrar más información en Thinkersmith's **Binary Baubles**.



Esta obra está disponible bajo una **Licencia Creative Commons (CC BY-NC-SA 4.0)**.

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

# Lección 16: Proyecto de fin de curso

45 minutos

## Resumen

Esta lección de **proyecto** lleva a los estudiantes a diseñar, desarrollar y exhibir nuevos proyectos de Play Lab.

## Propósito

Esta lección les brinda a los estudiantes la oportunidad de desarrollar un proyecto diseñado por ellos mismos, a través de un proceso paso a paso que requiere planificación y fomenta la creatividad.

## Estándares

Curso Completo Alineamiento

### Estándares de Ciencias de la Computación CSTA K-12 (2017)

- ▶ **AP** - Algorithms & Programming

## Agenda

### Actividad previa (10 minutos)

#### Planificación

### Actividad Principal (25 minutos)

#### Proyecto de programación

### Actividad de cierre (10 minutos)

#### Exhibición

## Objetivos

Los estudiantes podrán:

- Superar obstáculos como limitaciones de tiempo o errores.
- Usar un diseño planificado como plano para crear.

## Preparación

- Realiza los desafíos para encontrar cualquier área potencialmente problemática para tu clase.
- Imprime una **\*Guía de planificación de proyectos** para cada estudiante (o pareja).
- Revisa el documento **\*Guía de planificación de proyectos (ejemplo)**.
- (Opcional) Completa una guía de planificación y programa un proyecto tú mismo, para mostrarle a tus estudiantes.

## Enlaces

**iAviso!** Por favor, haga una copia de cualquier documento que planea compartir con los estudiantes.

Para los profesores

- **Guía de planificación de proyectos de Play Lab (ejemplo)** - Recursos de la lección

Para los estudiantes

- **Guía de planificación de proyectos en Play Lab** - Guía de trabajo

## Guía Didáctica

### Actividad previa (10 minutos)

#### Planificación

¡Motiva y prepara a los estudiantes para la actividad de hoy!

#### *Observaciones*

Ya tuvimos la oportunidad de desarrollar un proyecto en Play Lab. El ejercicio de hoy será mucho más libre, por lo que requiere una planificación previa. La planificación es un factor muy importante en la programación de un juego o cualquier otro tipo de software. Entonces, antes de pasar a los computadores, vamos a dedicar algo de tiempo a planificar los proyectos que queremos desarrollar.

**Distribuir:** entrega una **\*Guía de planificación de proyectos** a cada estudiante o pareja. Junto con los estudiantes, revisa los puntos de la guía y luego permíteles completarlos. Si es necesario, recurre al ejemplo.

#### Consejo didáctico

Si en esta actividad los estudiantes programarán en parejas, esta actividad inicial es una gran oportunidad para que practiquen cómo compartir y respetar las ideas de otros. Asegúrate de que los estudiantes estén siguiendo las normas para trabajar en equipo que hayas puesto para la clase. De lo contrario, explícales brevemente lo que esperas de ellos.

### Actividad Principal (25 minutos)

#### Proyecto de programación

Con sus guías de planificación a mano, los estudiantes están listos para que sus proyectos vean la luz. Estos niveles se corresponden con la estructura de la guía de planificación y ayudan a los estudiantes a abordar el proceso de transformar sus ideas en código.

#### Consejo didáctico

Los estudiantes experimentarán muchos desafíos y cometerán errores mientras programan. Es muy probable que sus proyectos se vuelvan versiones truncadas de su visión original. Recuérdales que esta clase de compromiso es normal en el diseño de software. ¡Está bien si no pueden desarrollar todas las características que planearon!



1

Exploración: Ejemplos de Proyectos



2-7

Proyecto

2

3

4

5

6

7

## Actividad de cierre (10 minutos)

### Exhibición

Para celebrar el trabajo de los estudiantes, dedica los últimos 10 minutos, más o menos, a que muestren sus proyectos. Esto puede hacerse de muchas maneras, pero aquí hay algunas:

- **Demostración pública:** selecciona a algunos voluntarios para que hagan una breve demostración de sus proyectos delante de la clase. Mientras lo hacen, pídeles que comenten cómo ha sido para ellos la experiencia de planificar y programar, incluyendo las ideas que todavía les gustaría poner en práctica.
- **Prueba de juego por parejas:** agrupa a los estudiantes en parejas y haz que prueben los proyectos de los demás. Mientras lo hacen, pídeles que hagan comentarios positivos y constructivos entre ellos. La ventaja es que los estudiantes tendrán la oportunidad de proporcionar y responder a los comentarios en un entorno más pequeño.
- **Paseo por la galería:** asegúrate de que todos los estudiantes tienen sus proyectos listos para la prueba. Pide a los estudiantes que se muevan al estilo de las "sillas musicales" a otro computador y que prueben el proyecto allí durante unos minutos, hasta que reciban una señal tuya para que se muevan a otro equipo. Repite esta operación cada pocos minutos. Aunque aquí hay menos oportunidades de comunicación estructurada, esto garantiza que los estudiantes puedan probar el mayor número posible de proyectos de sus compañeros.



Esta obra está disponible bajo una [Licencia Creative Commons \(CC BY-NC-SA 4.0\)](https://creativecommons.org/licenses/by-nc-sa/4.0/).

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