

Curso Express

Aprende los fundamentos de las ciencias de la computación y la seguridad en Internet. Al final del curso, crea tu propio juego o historia para compartir.

Enlaces para docentes: [Teacher Videos Playlist][0]

Chapter 1: Actividad previa

Lección 1: Fiesta de baile

Warm-up

En esta lección, los estudiantes programarán su propia fiesta de baile interactiva. Esta actividad requiere el uso de sonido, ya que la herramienta responde a la música.

Chapter Commentary

Actividad previa

Chapter 2: Secuencia

Lección 2: Programación con Angry Birds

Sequencing

Aprende acerca de las secuencias y algoritmos con Angry Birds.

Lección 3: Depuración con Scrat

Sequencing

Encuentra problemas en los desafíos y practica tus habilidades de debugging.

Lección 4: Recoger tesoros con Laurel

Sequencing

Escribe algoritmos para ayudar a Laurel la Aventurera a ¡recolectar muchas gemas!

Lección 5: Crear arte con código

Sequencing

Crear imágenes hermosas programando al Artista.

Chapter Commentary

Secuencia

Chapter 3: Bucles

Lección 6: Bucles con Rey y BB-8

Loops

¡Ayuda BB-8 a través de laberintos usando loops!

Lección 7: Arte con stickers y bucles

Loops

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

Lección 8: Bucles anidados en Laberinto

Loops

Bucles dentro de bucles que están dentro de otros bucles. ¿Qué significa esto? Esta lección te mostrará qué sucede cuando creas un bucle anidado.

Lección 9: Copos de nieve con Anna y Elsa

Loops

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

Lección 10: Mirando hacia el Futuro con Minecraft

Loops

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

Chapter Commentary

Bucles

Chapter 4: Condicionales

Lección 11: Condicionales con Abeja

Conditionals

Ahora que entiendes los condicionales, es hora de programar Bee para usarlos al recolectar miel y néctar.

Lección 12: Bucles 'mientras' con la Granjera

Conditionals

Los loops son tan útiles en la codificación. Esta lección te enseñará sobre un nuevo tipo de loops: ¡loops mientras!

Lección 13: Conditionals in Minecraft: Voyage Aquatic

Conditionals

Here you will learn about conditionals in the world of Minecraft.

Lección 14: Until Loops in Maze

Conditionals

You can do some amazing things when you use `until` loops!

Lección 15: Cosechar con condicionales

Conditionals

No siempre está claro cuándo usar cada condicional. Esta lección te ayudará a tener práctica en decidir qué hacer.

Chapter Commentary

Condicionales

Chapter 5: Funciones

Lección 16: Functions in Minecraft

Functions

Can you figure out how to use functions for the most efficient code?

Lección 17: Functions with Harvester

Functions

¡Las funciones te ahorrarán mucho trabajo mientras ayudas a la granjera con su cosecha!

Lección 18: Functions with Artist

Functions

¡Haz dibujos complejos de manera más fácil usando funciones!

Chapter Commentary

Funciones

Chapter 6: Variables

Lección 19: Variables with Artist

Variables

Don't forget to bring creativity to class! In these puzzles you will be making fantastic drawings using variables.

Lección 20: Changing Variables with Bee

Variables

This bee loves variables!

Lección 21: Changing Variables with Artist

Variables

In this lesson, you'll make drawings using variables that change as the program runs.

Chapter Commentary

Variables

Chapter 7: Bucles desde

Lección 22: For Loops with Bee

For Loops

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

Lección 23: For Loops with Artist

For Loops

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

Chapter Commentary

Bucles desde

Chapter 8: Sprites

Lección 24: Swimming Fish in Sprite Lab

Sprites

Practica haciendo juegos para compartir con tus amigos/as y familia.

Lección 25: Alien Dance Party

Sprites

Practica haciendo juegos para compartir con tus amigos/as y familia.

Lección 26: Behaviors in Sprite Lab

Sprites

Learn to program your own sprite behaviors!

Lección 27: Virtual Pet with Sprite Lab

Sprites

In this lesson, students will create an interactive Virtual Pet that looks and behaves how they wish. Students will use Sprite Lab's "Costumes" tool to customize their pet's appearance. They will then use events, behaviors, and other concepts they have learned to give their pet a life of its own!

Chapter Commentary

Sprites

Chapter 9: Proyecto de fin de curso

Lección 28: Proyecto de fin de curso

End of Course Project

¡Prepara esas manos para programar mucho! Es hora de empezar a construir tu proyecto.

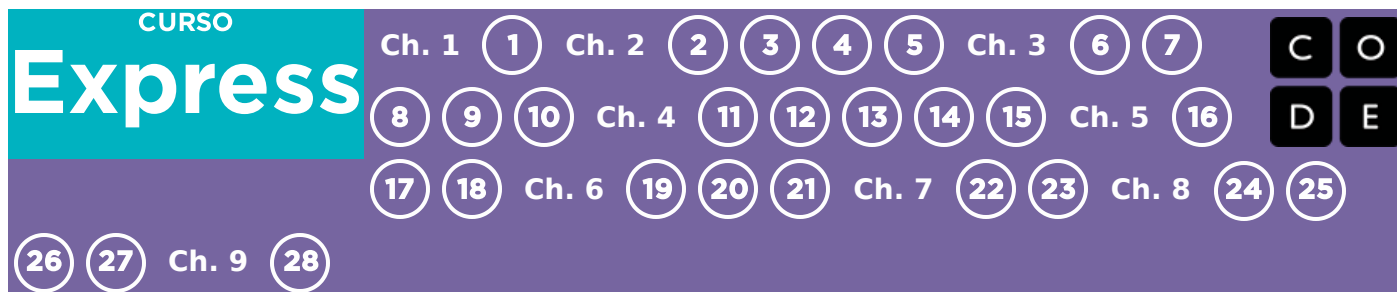
Chapter Commentary

Proyecto de fin de curso



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

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



Lección 1: Fiesta de baile

Actividad previa

Descripción

En esta lección, los estudiantes programarán su propia fiesta de baile interactiva. Esta actividad requiere el uso de sonido, ya que la herramienta responde a la música.

Propósito

Esta lección presenta los conceptos básicos de las ciencias de la computación para la codificación y la programación de eventos (con bloques).

Agenda

Manos a la obra (5 minutos)

Actividad: (30-45 minutos)

Cierre (5 minutos)

Debrief

Evaluación (2 minutos)

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Desarrollar programas que respondan a eventos cronometrados
- Desarrollar programas que respondan a las entradas de los usuarios
- Crear animaciones de baile a través del código

Preparación

- ☐ Play through the puzzles to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.
- ☐ Consider the need for headphones. This activity relies on sound.

Enlaces

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

Para los docentes

- **Lista de Reproducción en Spotify (todas las edades)**

Para los estudiantes

- **Dance Party Project Guide - Ficha**

Hacer una copia ▾

Vocabulario

- **programar** - (v) escribir código, o escribir instrucciones para una computadora.
- **Evento** - una acción que hace que algo suceda.
- **Programa** - un algoritmo que se ha programado como algo que puede ser ejecutado por una máquina.

Guía docente

Manos a la obra (5 minutos)

Actividad: (30-45 minutos)

Cierre (5 minutos)

Debrief

- Pose a prompt that has multiple answers such as “What is something you enjoyed about today's activity?” or “What is the connection between creativity and computer science?”

Evaluación (2 minutos)

Pídele a los alumnos que añadan sus notas adhesivas "Importantes" o tarjetas de notas a su mapa mental de "Ciencias de Computación" antes de que salgan de clase. Trata de llenar el cuadro con muchas ideas acerca de lo que son las Ciencias de Computación y por qué son importantes.

Standards Alignment

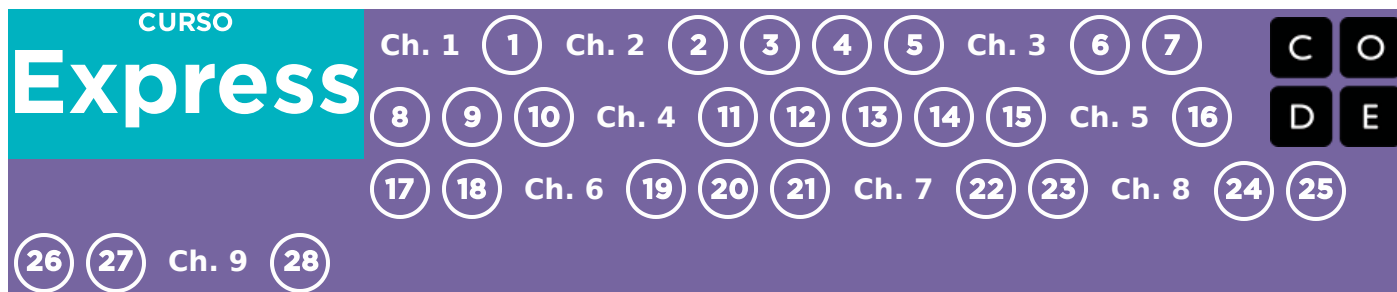
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 2: Programación con Angry Birds

Secuencia

Descripción

Usando los personajes del juego Angry Birds, los/as estudiantes desarrollarán algoritmos secuenciales para mover al pájaro de un lado del laberinto hacia el cerdo al otro lado. Para hacer esto ellos/as apilarán bloques de código en una secuencia lineal, haciendo que se muevan derecho, doblen a la izquierda o doblen a la derecha.

Propósito

En esta lección, los/as estudiantes desarrollarán las habilidades de programación y debugging en una plataforma computacional. El formato basado en bloques de estos desafíos ayuda a los /as estudiantes a aprender sobre la secuencia y los conceptos, sin tener que preocuparse de una sintaxis perfecta.

Agenda

Previsualizar desafíos online como clase (3 min)

Teaching this course as a class?

Actividad Principal (30 min)

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Traducir movimientos en una serie de comandos.
- Identificar y localizar errores en un programa.

Preparación

- ☐ Play through the puzzles to find any potential problem areas for your class.
- ☐ (Optional) Pick a couple of puzzles to do as a group with your class.
- ☐ Make sure every student has a **Diario**

Think Spot - Diario de reflexiones

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Algoritmo** - una lista de tareas para finalizar una tarea.
- **Bug o error** - parte de un programa que no funciona correctamente.
- **Depurar** - encontrar y solucionar problemas en un algoritmo o programa.
- **Secuencia** - poner los comandos en el orden correcto para que las computadoras puedan leerlos.

Guía docente

Previsualizar desafíos online como clase (3 min)

 **Model:** Pull up Puzzle 5 to do in front of the class. While working through this puzzle with the class, remind students that making mistakes is okay and remind them that the only way to be successful is to be persistent.

Discuss: Does anyone know how to solve this puzzle?

As the teacher, you should decide if you will have the students tell you how to solve it from their seats, or come to the computer to drag the actual blocks in one-by-one.

Transition: Now that students have seen an online puzzle in practice, they should be ready to start solving puzzles of their own. Continue to the lab or bring out their classroom machines.

Lesson Tip

Some students may struggle with turning their bird in the correct direction, particularly when the bird isn't facing up. Remind students that when we say turn left or right, we're giving directions from the bird's point of view.

Content Corner

Teaching this course as a class?

Our grade-aligned CS Fundamentals courses use unplugged lessons to build community and introduce tricky computer science concepts, including **sequencing**. Check out the lesson **My Robotic Friends** from **Course C** and **Graph Paper Programming** from **Course D**!

Actividad Principal (30 min)

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

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

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- Dibuja una actividad que te gusta hacer con la que tuviste dificultad la primera vez. Dibuja o describe cómo mejoraste.

Aprendizaje Extendido

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

Crea el tuyo

En grupos pequeños, deje que los/as estudiantes diseñen sus propios laberintos y desafíen a los demás a escribir programas para resolverlos. Para mayor diversión, haga laberintos con los/as estudiantes de tamaño natural como el cerdo y el pájaro.

Standards Alignment

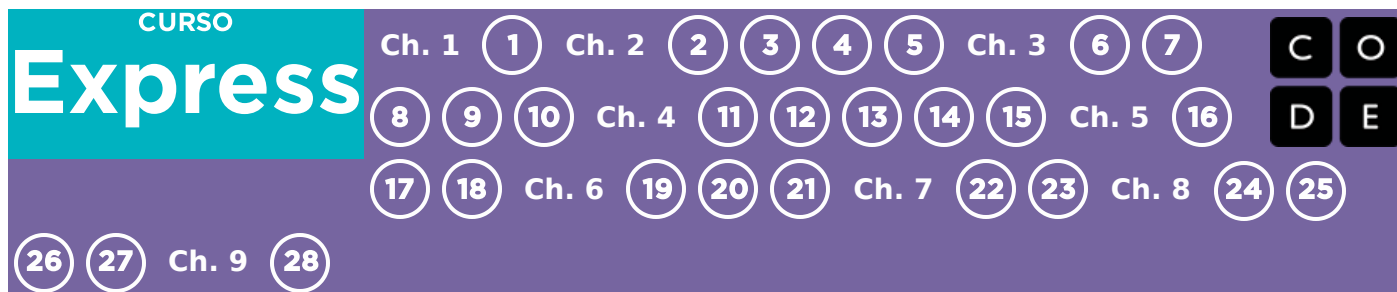
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 3: Depuración con Scrat

Secuencia

Descripción

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

Propósito

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

Agenda

Actividad previa (15 mín)

Introducción
Vocabulario

Actividad Principal (30 mín)

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

Aprendizaje Extendido

[Ver en Code Studio](#)

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ (Optional) Pick a couple of puzzles to do as a group with your class.
- ☐ Make sure every student has a **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los estudiantes

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

[Hacer una copia](#)

Vocabulario

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

Guía docente

Actividad previa (15 mín)

Introducción

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

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

Vocabulario

Esta lección tiene tres nuevas e importantes palabras de vocabulario:

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

Diga:

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

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

Actividad Principal (30 min)

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Aprendizaje Extendido

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

Plantar bugs

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

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

Standards Alignment

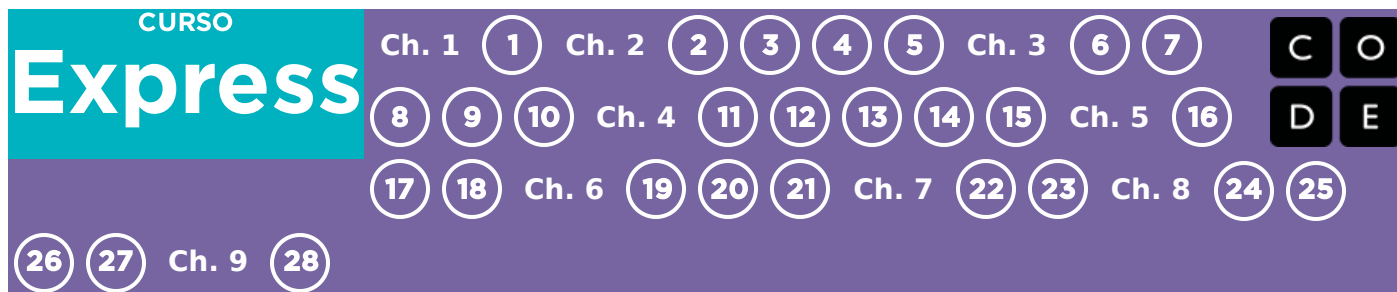
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 4: Recoger tesoros con Laurel

Secuencia

Descripción

En esta serie de desafíos, los/as estudiantes continuarán desarrollando su comprensión de algoritmos y debugging. Con un nuevo personaje, Laurel la Aventurera, los/as estudiantes crearán algoritmos secuenciales para que Laurel recoja el tesoro mientras va por el camino.

Propósito

En esta lección, los/as estudiantes practicarán sus habilidades de programación usando un nuevo personaje, Laurel la Aventurera. Cuando alguien comienza a programar, junta las instrucciones en un orden específico usando algo que una máquina pueda leer. A través del uso de la programación, los/as estudiantes desarrollarán un entendimiento de cómo un computador navega por las instrucciones y el orden. El uso de un nuevo personaje con un objetivo diferente ayudará a los/as estudiantes a ampliar su ámbito de experiencia con la secuenciación y algoritmos en la programación.

Agenda

Actividad previa (5 min)

Introduction

Actividad Puente - Programación (10 min)

Previewing Online Puzzles as a Class

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Ordenar comandos de movimiento como pasos secuenciales en un programa.
- Representar un algoritmo como un programa de computadora.
- Desarrollar habilidades de resolución de problemas y pensamiento crítico mediante la revisión de las prácticas de depuración.

Preparación

- ☐ Play through the lesson to find potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Algoritmo** - una lista de tareas para finalizar 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 docente

Actividad previa (5 min)

Introduction

This lesson uses most of the same blocks from Programming in Maze and adds the ability to `collect`. Tell the students that this block will allow Laurel the Adventurer to pick up the treasure that she is standing over. This new block will be discussed more in the bridging activity.

Actividad Puente - Programación (10 min)

Previewing Online Puzzles as a Class

Pull a puzzle from the corresponding online stage. We recommend puzzle 7. Have students discuss a pattern that they think will get Laurel the Adventurer to collect all the treasure. Ask the students to share. See how many other students had the same answer!

Actividad Principal (30 min)

Online Puzzles

Laurel the Adventurer is looking to collect as much treasure as she can. Instruct the students to traverse the puzzle to collect whatever they can. Some levels will require you to only pick up one piece of treasure, but others will require you to pick up every piece of treasure. Pay attention to the instructions to know what to do!

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Standards Alignment

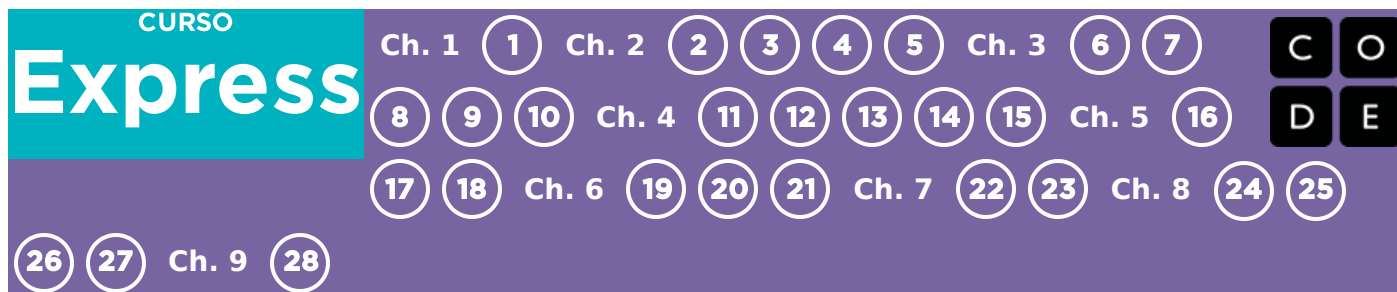
Estándares para ciencias de la computación CSTA K-12 (2017)

- AP - Algoritmos y programación



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

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



Lección 5: Crear arte con código

Secuencia

Descripción

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

Propósito

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

Agenda

Actividad previa (10 min)

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (10 - 15 min)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- Play through the lesson to find any potential problem areas for your class. - (Optional) Obtain protractors for your class to visualize the angles they must use to complete the puzzles. - Print one [r csf-turns-angles-doc] for each student.

Enlaces

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

Para los estudiantes

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

Guía docente

Actividad previa (10 min)

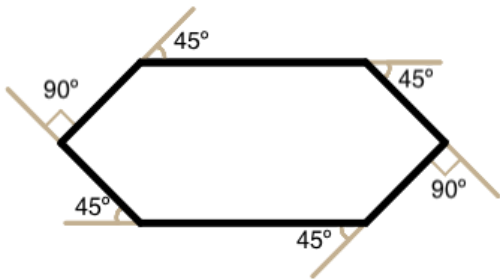
Actividad Principal (30 min)

Online Puzzles

In this set of puzzles, the artist will no longer be constrained to 90 degree angles. Having physical protractors available can help students better visualize the angles they need. Otherwise, the stage provides images of the angles as the student selects which angle to use. (Please note: Angle choices are limited to two inside of the dropdown menu, reducing the number of options students have to work through.)

Before sending the students to the computers to work on the puzzles, it might be beneficial to give a brief presentation of how to use the tools in this level. We recommend puzzle 5 as a good puzzle to show how to use the protractor online.

The eighth puzzle asks the students to draw a 6 sided polygon. This might be challenging for some students. We recommend getting the students to try a few times, ask a peer, then ask the teacher for help. Below is an image that might be helpful for the students.



Actividad de Cierre (10 - 15 min)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Aprendizaje Extendido

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

La fotocopiadora

- Entregue a los/as estudiantes dos hojas de papel.
- En una de las hojas, dibuje una imagen sencilla, usando sólo líneas rectas.

- En la segunda hoja, dibuje las instrucciones para recrear la imagen usando comandos para moverse en línea recta y girar en distintos ángulos.
- Intercambie las hojas de instrucciones e intente recrear la imagen usando sólo las instrucciones entregadas.

Standards Alignment

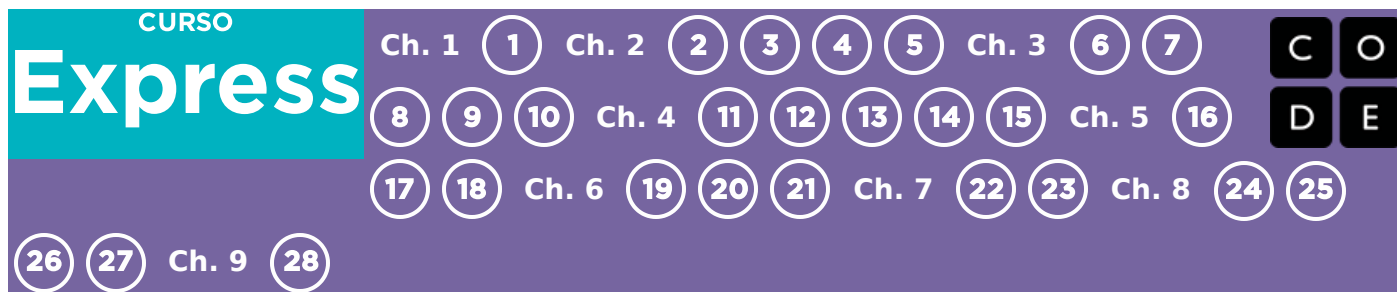
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 6: Bucles con Rey y BB-8

Bucles

Descripción

Basándose en el concepto de repetir las instrucciones de "Volviéndose Loopy", esta etapa tendrá a los/as estudiantes usando loops para ayudar a BB-8 a atravesar un laberinto más eficientemente que antes.

Propósito

En esta lección, los/as estudiantes aprenderán más sobre los loops y cómo implementarlos en código Blockly. El uso de loops es una habilidad importante en la programación porque la repetición manual de comandos es tediosa e ineficiente. Con los desafíos de Code.org, los/as estudiantes aprenderán a añadir instrucciones a los loops existentes, juntar programas repetidos en loops y reconocer patrones que necesitan ser enlazados. Cabe señalar que los/as estudiantes se enfrentarán a desafíos con muchas soluciones diferentes. Esto abrirá discusiones sobre las diferentes formas de resolver los desafíos con ventajas y desventajas de cada enfoque.

Agenda

Actividad Puente - Bucles (10 min)

Teaching this course as a class?

Previewing Online Puzzles as a Class

Actividad Principal (30 min)

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Identificar los beneficios de usar una estructura de bucles en lugar de la repetición manual.
- Dividir una secuencia larga de instrucciones en la secuencia repetible más grande.
- Emplear una combinación de comandos secuenciales y en bucle para llegar al final de un laberinto.

Preparación

- ☐ Play through the lesson to determine if there will be any problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

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

Guía docente

Actividad Puente - Bucles (10 min)

Previewing Online Puzzles as a Class

🎓 Pull up the online puzzles and choose a puzzle to do in front of the class. We recommend puzzle 8 for its staircase pattern. Ask the students to write a program to solve the puzzle on paper.

Content Corner

Teaching this course as a class?

Our grade-aligned CS Fundamentals courses use unplugged lessons to build community and introduce tricky computer science concepts, including **loops**. Check out the lesson **My Loopy Robotic Friends** from **Course C!**

Actividad Principal (30 min)

Actividad de Cierre (5 - 10 minutos)

Escribir en el Diario

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

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Cómo hicieron los loops que tu programa fuera más fácil de escribir?
- Piensa en algo que se repite una y otra vez. ¿Cómo podría ser el programa para eso?

Aprendizaje Extendido

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

¡A moverse!

- Entregue a los estudiantes fotos de acciones o movimientos de baile que puedan hacer.
- Haga que los/as estudiantes organicen movimientos y agreguen loops para coreografiar su propio baile.
- Comparta los bailes con el resto de la clase.

Reconectando

- Encuentre algunos videos en YouTube de bailes populares que se repiten.
- ¿Puede su clase encontrar los loops?
- ¡Intente lo mismo con canciones!

Standards Alignment

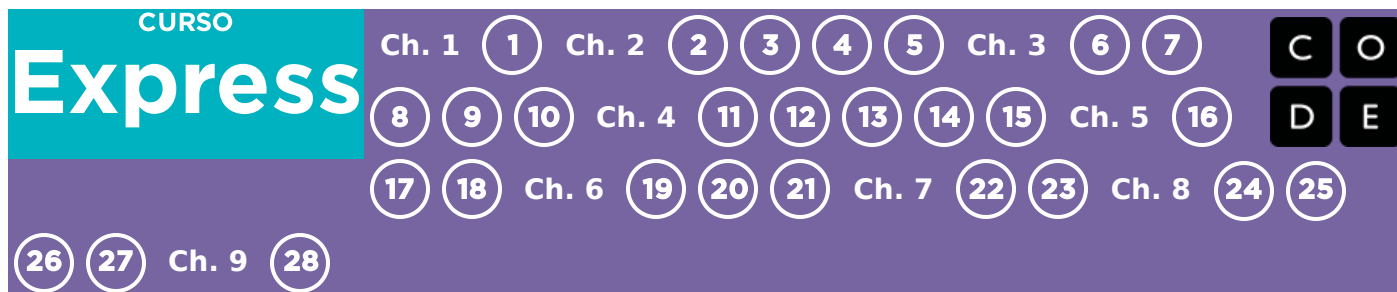
Estándares para ciencias de la computación CSTA K-12 (2017)

- ▶ **AP** - Algoritmos y programación



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

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



Lección 7: Arte con stickers y bucles

Bucles

Descripción

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

Propósito

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

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

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

[Hacer una copia](#)

Vocabulario

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

Guía docente

Actividad previa (15 mín)

Introduction

Students should have had plenty of introduction to loops at this point. Based on what you think your class could benefit from, we recommend:

- As a class, playing through a puzzle from the last lesson, "Loops with Rey and BB-8"
- Reviewing how to use Artist by playing through a puzzle from "Programming in Artist"
- Previewing a puzzle from this lesson

All of these options will either review loops or the artist, which will help prepare your class for fun with the online puzzles!

Actividad Principal (30 min)

Online Puzzles

Some students may discover where to add `repeat` loops by writing out the program without loops then circling sections of repetitions. If the students in your class seem like they could benefit from this, have them keep paper and pencils beside them at their machines. Students might also enjoy drawing some of the shapes and figures on paper before they program it online. (When drawing stamps, it can be easier to symbolize those with simple shapes like circles and squares.)

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Standards Alignment

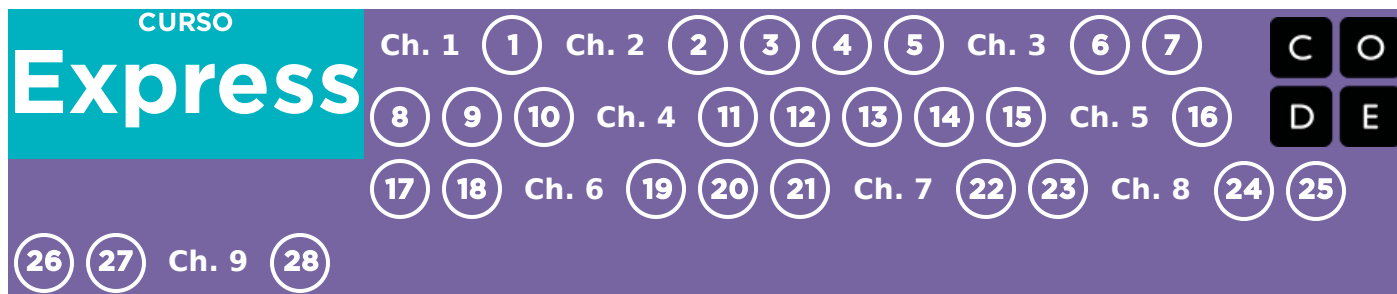
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 8: Bucles anidados en Laberinto

Bucles

Descripción

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

Propósito

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

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

Agenda

Actividad previa (10 min)

Introducción

Actividad Principal (30 min)

Actividad de Cierre (15 min)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Comando** - una instrucción para la computadora. Muchos comandos juntos componen algoritmos y programas de computadora.
- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.

Guía docente

Actividad previa (10 min)

Introducción

Repase brevemente con la clase qué son los loops y por qué los usamos.

- ¿Qué hacen los loops? - Los loops repiten un conjunto de comandos. (ver vocabulario en comando si los/as estudiantes no lo reconocen)
- ¿Cómo utilizamos los loops? - Utilizamos loops para crear un patrón hecho de acciones repetidas.

Dígale a la clase que ahora van a hacer algo súper genial: usar loops dentro de los loops. Pida a la clase que prediga para qué tipo de cosas usaríamos un loop dentro de un loop.

"Si un loop repite un patrón, entonces un loop repetirá un patrón de patrones".

Los/as estudiantes no necesitan entender esto de inmediato, así que siéntase libre de pasar a los desafíos online incluso si los/as estudiantes todavía parecen un poco confundidos. .

Actividad Principal (30 min)

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Standards Alignment

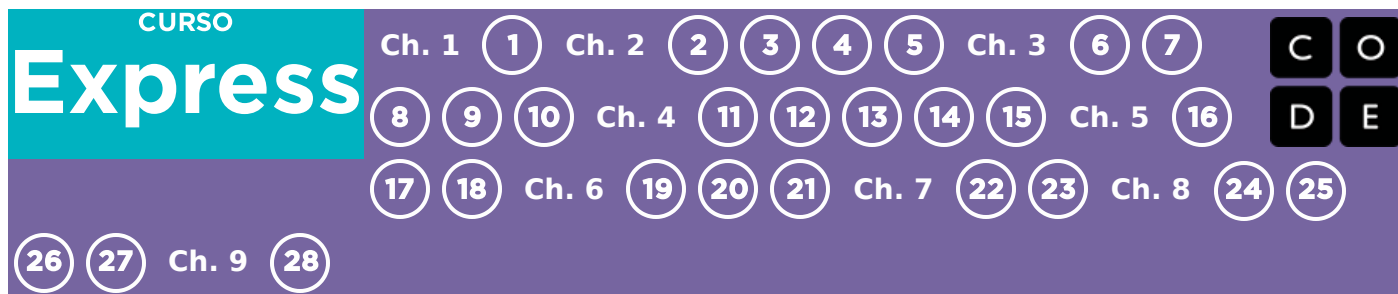
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 9: Copos de nieve con Anna y Elsa

Bucles

Descripción

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

Propósito

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

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

Agenda

Actividad previa (15)

Introducción

Actividad Principal (30)

Online Puzzles

Actividad de Cierre (15)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find and potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

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

Guía docente

Actividad previa (15)

Introducción

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

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

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

Actividad Principal (30)

Online Puzzles

This set of puzzles is set up as a progression. This means every puzzle builds a foundation for the next puzzle. Students will enjoy making more and more interesting designs by making small and simple changes to code they have already written.

Actividad de Cierre (15)

Escribir en el Diario

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

Sugerencias para el diario:

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

Standards Alignment

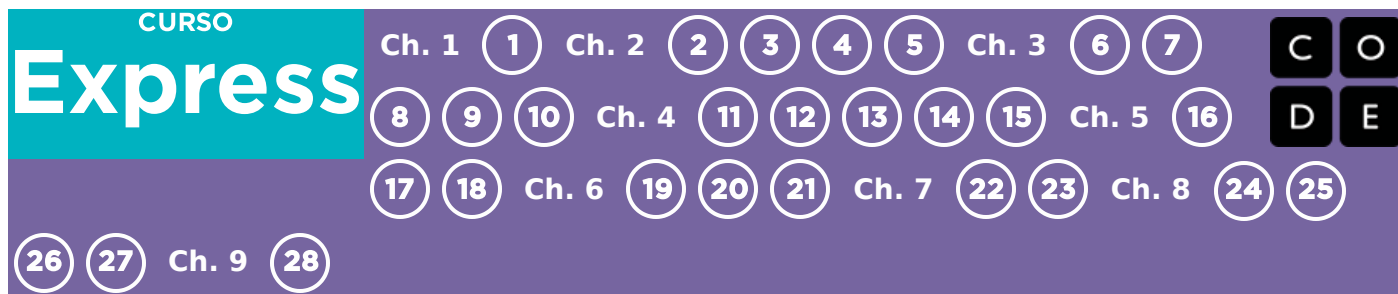
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 10: Mirando hacia el Futuro con Minecraft

Bucles

Descripción

This lesson was originally created for the Hour of Code, alongside the Minecraft team. Students will get the chance to practice ideas that they have learned up to this point, as well as getting a sneak peek at conditionals!

Propósito

This set of puzzles will work to solidify and build on the knowledge of loops, and introduce conditionals. By pairing these two concepts together, students will be able to explore the potential for creating fun and innovative programs in a new and exciting environment.

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Journaling

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

- Diario Think Spot** - Diario de reflexiones

Hacer una copia ▾

Vocabulario

- Condición** - algo que un programa verifica para ver si es verdadero antes de permitir una acción.
- Condicionales** - sentencias que solo se ejecutan bajo ciertas condiciones.

Guía docente

Actividad previa (15 mín)

Introduction

Gather the class together and ask two volunteers to walk straight in some direction in the classroom. If they encounter a chair out of place, they must step over it. If they reach a wall, they must sit down.

Once all of the students are sitting down, ask how you would program a robot to respond to a wall or a chair. Remind students that you cannot simply say "Step over chair" unless you know there is a chair, and you will not always know there is a chair. It might be helpful to translate the task into instructions like:

- while there is a path ahead
 - walk forward
 - if there is a chair, step over it
- sit down

Tell students they will be using *conditionals* during this lesson. Give the definition of:

- **Condition:** A statement that a program checks to see if it is true or false. If true, an action is taken. Otherwise, the action is ignored.
- **Conditionals:** Statements that only run under certain conditions.

Open up a discussion of when you might use a conditional in your code.

Actividad Principal (30 min)

Online Puzzles

Students are in for a real treat with this lesson. It's likely most of your students have heard of Minecraft, but give a brief introduction for those that may not know.

Minecraft is a game of cubes. You can play as Alex or Steve as you work through mazes. You'll need to avoid lava, pick up items, and explore in a world made up of cubes of things.

Demonstrate one of the puzzles to the class (we recommend puzzle 11.) Once all questions have been addressed, transition students to computers and let them start pair programming.

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- Draw a feeling face to show how you felt during today's lesson.
- Draw something else you could have built in this Minecraft world.
- Can you draw a scene where someone is using a conditional?

Aprendizaje Extendido

Más Minecraft

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

Standards Alignment

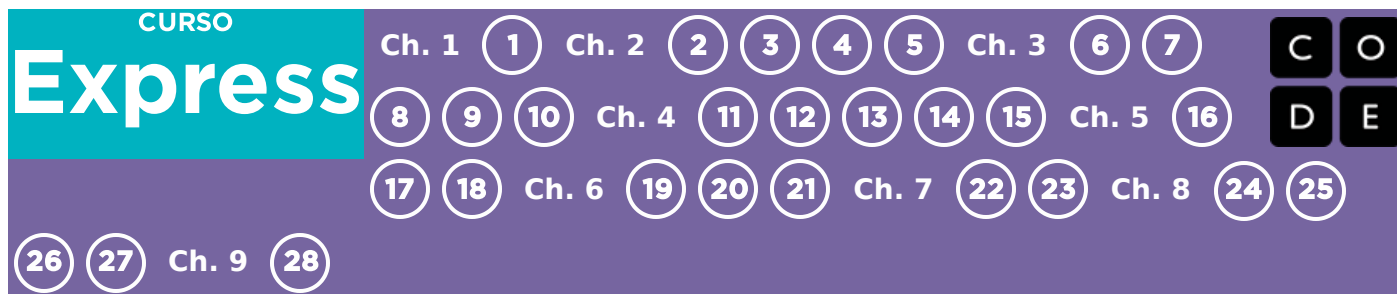
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 11: Condicionales con Abeja

Condicionales

Descripción

Hasta ahora los/as estudiantes han estado escribiendo código que se ejecuta exactamente del mismo modo cada vez que se ejecuta - confiable, pero no muy flexible. En esta lección, su clase comenzará a codificar con condicionales, permitiéndoles escribir código que funcione de forma diferente dependiendo de las condiciones específicas que el programa encuentre.

Propósito

Después de ser introducidos a condicionales en "Condicionales con Tarjetas", los/as estudiantes practicarán usándolos en sus programas. Los bloques `if / else` permitirán un programa más flexible. La abeja sólo recogerá néctar **si** hay una flor o hacer miel **si** hay un panal de abejas. Los/as estudiantes también practicarán y reconocerán una conexión entre bloques `if / else` y loops `while` en este conjunto de desafíos.

Agenda

Preview of Online Puzzles (15 mín)

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Convertir sentencias condicionales de lenguaje hablado en un programa.
- Resolver desafíos mediante una combinación de secuencias en bucle y condicionales.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
 - ☐ Make sure every student has a **Diario**
- Think Spot - Diario de reflexiones.**

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

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

Guía docente

Preview of Online Puzzles (15 mín)

Pull up a puzzle from the lesson, we recommend puzzle 9.

- Ask the class what the bee should do when it gets to the cloud.
 - The bee should use a conditional to check for a flower or a honeycomb.
- Use the `if at flower / else` block. Ask the class what the bee should do if there's a flower. If there's not a flower, there will be a honeycomb. What should the bee do then?
 - The bee should `get nectar` if there is a flower and `make honey` if there is a honeycomb.

Fill in the rest of the code and press `Run`. Discuss with the class why this worked.

Actividad Principal (30 min)

Online Puzzles

These puzzles might sprout some questions, so have the students work in pairs or implement the "Ask three before you ask me" rule (have the students ask three other peers for help before they go to the teacher.) This will spark discussions that will develop each student's understanding.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué condicionales usaste en tu código de hoy?
- ¿Cuáles son otros condicionales que la abeja podría usar? Algunos ejemplos incluyen:
 - si hay un árbol frente mío, salir del camino
 - si mi ala está herida, descansar en el suelo
 - si veo otra abeja, decir "¡Hola!"

Aprendizaje Extendido

Utilice estas actividades para mejorar el aprendizaje de los estudiantes. Pueden ser utilizados como fuera de las actividades de clase o como otro tipo de enriquecimiento.

**** Etiqueta de Verdadero/Falso ****

- Ponga a los estudiantes en fila como si fuera a jugar **luz roja / luz verde**.
- Seleccione una persona para pararse en frente como la persona que elige.
- La persona elige una condición y le pide a todos los que cumplan esa condición que den un paso adelante.
 - Si tienes un cinturón rojo, paso adelante.
 - Si usted está usando sandalias, tomar un paso hacia adelante.
- Intente hacer cambios diciendo cosas como "Si no eres rubio, paso adelante."

**** Anidación ****

- Ponga a los estudiantes en parejas o en grupos pequeños.
- Haga que escriban declaraciones para jugar tarjetas sobre tiras de papel, tales como:
 - Si el traje es de clubes
 - Si el color es rojo -Haga que los estudiantes creen tiras similares para los resultados.
 - Añadir un punto.-Reste un punto
- Una vez hecho esto, haga que los estudiantes elijan tres de cada tipo de tira y tres naipes, de acuerdo al orden de la selección.
- Usando tres pedazos de papel, haga que los estudiantes escriban tres diferentes programas utilizando sólo los conjuntos de tiras que seleccionaron, en cualquier orden.
 - Anime a los estudiantes a poner algunas declaraciones if dentro de otra declaración if.
- Ahora, los estudiantes deben ejecutar los tres programas usando las tarjetas que dibujaron, en el mismo orden para cada programa.
 - ¿Alguno de los dos programas dio la misma respuesta?
 - ¿Alguno dio algo diferente?

Standards Alignment

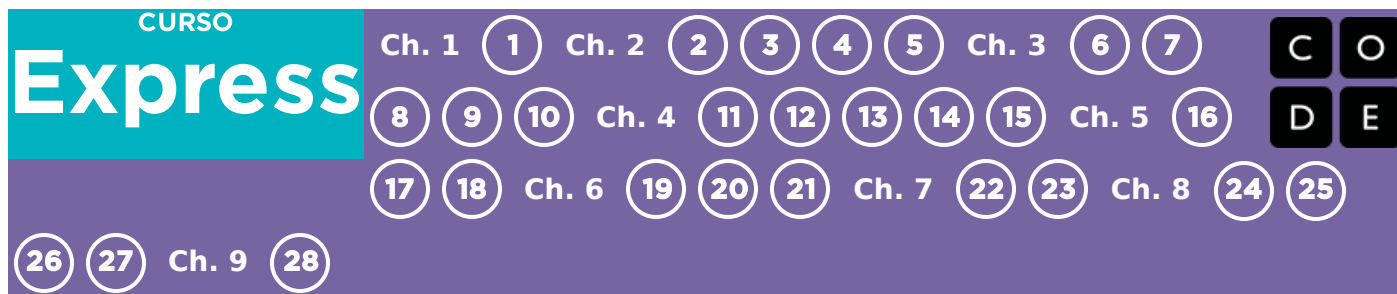
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 12: Bucles 'mientras' con la Granjera

Condicionales

Descripción

Para el momento en que los/as estudiantes lleguen a esta lección, ya deberían haber practicado lo suficiente usando loops repetir, así que ahora es tiempo de mezclar las cosas. Los loops mientras son loops que continúan repitiendo comandos mientras se cumpla una condición. Los loops mientras se usan cuando el/la programador/a no sabe el número exacto de veces que es necesario repetir los comandos, pero sí sabe qué condición necesita ser cierta para que el loop continúe repitiendo. Por ejemplo, los/as estudiantes trabajarán llenando hoyos y sacando tierra en Granjera. No sabrán el tamaño de los hoyos ni la altura de las montañas de tierra, pero sabrán que necesitan seguir llenando los hoyos y sacando tierra mientras el suelo no esté plano.

Propósito

A medida que sus estudiantes sigan profundizando sus conocimientos acerca de los loops, encontrarán problemas en los que un comando debe repetirse, pero es desconocido cuántas veces necesita repetirse. Aquí es donde vienen loops mientras. En la lección de hoy, los estudiantes desarrollarán la comprensión de los loops basados en las condiciones y también ampliarán su conocimiento de loops en general.

Agenda

Actividad previa (10 min)

Actividad Principal (30 min)

Actividad de Cierre (15 min)

Escribir en el Diario

Actividad Puente (15 min)

Preview of Online Puzzles

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Distinguir entre bucles que se repiten un número fijo de veces y bucles que se repiten siempre que una condición sea verdadera.
- Usar un bucle mientras para crear programas que puedan resolver problemas con valores desconocidos.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Condición** - algo que un programa verifica para ver si es verdadero antes de permitir una acción.
- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.
- **Bucles mientras** - un bucle que se sigue repitiendo mientras una condición es verdadera.

Guía docente

Actividad previa (10 min)

Use “mientras” en una oración en frente de la clase. Pregúnteles que significa la palabra “mientras”. Si usted dijera “mientras exista un hoyo, llénalo con tierra” ¿Qué harían? ¿Por cuánto tiempo lo harían?

Cuando usas una palabra como “mientras”, estas confiando en una condición que le diga al computador por cuánto tiempo debería ejecutarse el loop. Una condición es una afirmación que se prueba y se descubre si es verdadera o falsa. En el caso anterior, la condición es si hay un hoyo. Sólo es posible que exista un hoyo o que no exista un hoyo, así la afirmación puede ser sólo verdadera o falsa. Cuénteles a los/as estudiantes que aprenderán acerca de un nuevo tipo de loops. Previamente, los/as estudiantes solo usaron loops para repetir un comando un cierto número de veces. Aquí, no siempre sabrán cuántas veces repetir el comando, sin embargo, sabrán cuando detenerse o cuando continuar. Los loops mientras le permiten al/a programador/a repetir un comando siempre que una condición sea verdadera.

Si hay tiempo, haga que los/as estudiantes discutan otras veces en las que el uso de un “loop mientras” sería útil. Por ejemplo:

- Correr hacia una pelota mientras esté frente a ti.
- Llenar un vaso mientras exista espacio para más líquido.
- Avanzar mientras exista un camino adelante.

Actividad Principal (30 min)

Actividad de Cierre (15 min)

Escribir en el Diario

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

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Cuál es la diferencia entre un loop mientras y un loop repetir normal?
- Da un ejemplo de un desafío en el que usarías un loop mientras , pero no un loop repetir . ¿Puedes dar un ejemplo de un desafío en el que usarías un loop de repetir , pero no un loop mientras ?

Actividad Puente (15 min)

Preview of Online Puzzles

Pull up a puzzle from the lesson, we recommend Puzzle 6.

- Ask the class what the farmer should do when she gets to the pile of dirt.
 - The farmer should use a conditional in a while loop to check if there is a pile.
- Use the while there is a pile / do block. Ask the class what the farmer should do if there is no pile. When should the farmer stop?
 - The farmer should remove 1 if there is a pile. The while loop will continue until there is no pile, so then your code is finished!

Fill in the rest of the code and press Run . Discuss with the class why this worked.

Standards Alignment

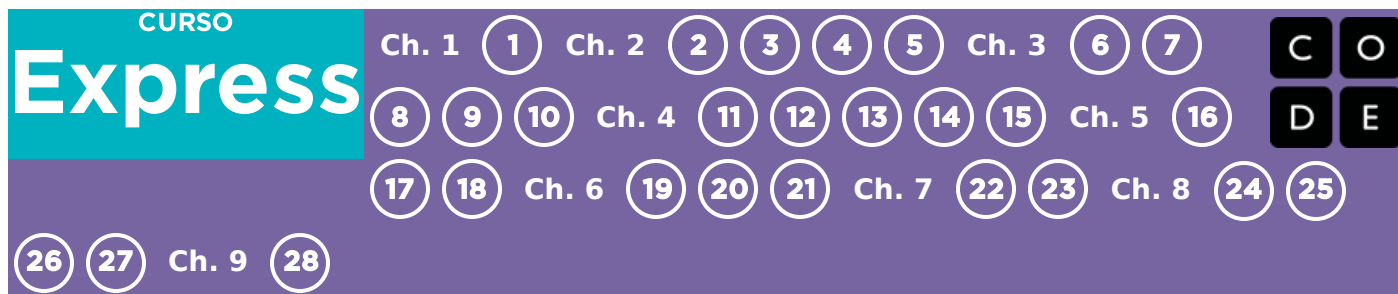
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 13: Conditionals in Minecraft: Voyage Aquatic

Condicionales

Descripción

This lesson was originally created for the Hour of Code, alongside the Minecraft team. Students will get the chance to practice ideas that they have learned up to this point, as well as getting a sneak peek at conditionals!

Propósito

This set of puzzles will work to solidify and build on the knowledge of loops, and introduce conditionals. By pairing these two concepts together, students will be able to explore the potential for creating fun and innovative programs in a new and exciting environment.

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Journaling

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the puzzles associated with this lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

- Diario Think Spot** - Diario de reflexiones

Hacer una copia ▾

Vocabulario

- Condición** - algo que un programa verifica para ver si es verdadero antes de permitir una acción.
- Condicionales** - sentencias que solo se ejecutan bajo ciertas condiciones.

Guía docente

Actividad previa (15 mín)

Introduction

Gather the class together and ask two volunteers to walk straight in some direction in the classroom. If they encounter a chair out of place, they must step over it. If they reach a wall, they must sit down.

Once all of the students are sitting down, ask how you would program a robot to respond to a wall or a chair. Remind students that you cannot simply say "Step over chair" unless you know there is a chair, and you will not always know there is a chair. It might be helpful to translate the task into instructions like:

- while there is a path ahead
 - walk forward
 - if there is a chair, step over it
- sit down

Tell students they will be using *conditionals* during this lesson. Give the definition of:

- **Condition:** A statement that a program checks to see if it is true or false. If true, an action is taken. Otherwise, the action is ignored.
- **Conditionals:** Statements that only run under certain conditions.

Open up a discussion of when you might use a conditional in your code.

Actividad Principal (30 min)

Online Puzzles

Students are in for a real treat with this lesson. It's likely most of your students have heard of Minecraft, but give a brief introduction for those that may not know.

Minecraft is a game of cubes. You can play as Alex or Steve as you work through mazes. You'll need to pick up items, and explore in a world made up of cubes of things.

Demonstrate one of the puzzles to the class (we recommend puzzle 11.) Once all questions have been addressed, transition students to computers and let them start pair programming.

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- Draw a feeling face to show how you felt during today's lesson.
- Draw something else you could have built in this minecraft world.
- Can you draw a scene where someone is using a conditional?

Aprendizaje Extendido

Más Minecraft

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

Standards Alignment

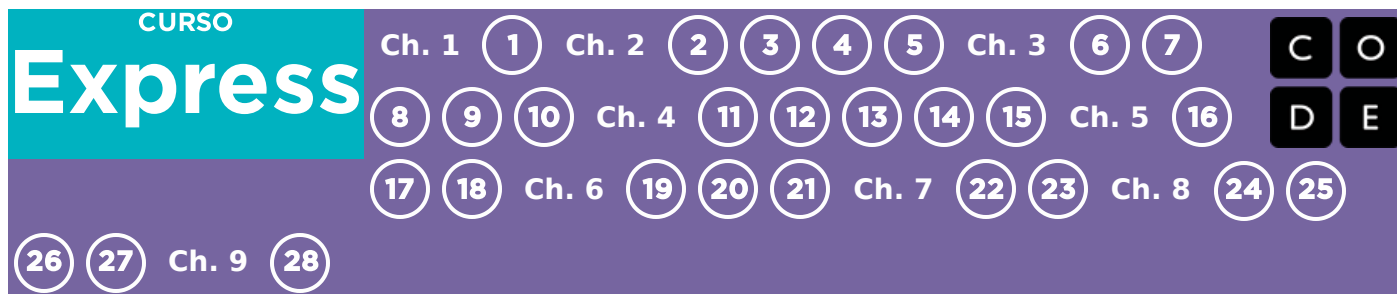
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 14: Until Loops in Maze

Condicionales

Descripción

In this lesson, students will learn about `until` loops. Students will build programs that have the main character repeat actions `until` they reach their desired stopping point.

Propósito

This set of puzzles will work to solidify and build on the knowledge of loops by adding the `until` conditional. By pairing these concepts together, students will be able to explore the potential for creating complex and innovative programs.

Agenda

Actividad previa (10 min)

Introduction

Actividad Puente (15 min)

Preview of Online Puzzles

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 min)

Journaling

[Ver en Code Studio](#)

Objetivos

Los estudiantes podrán:

- Desarrollar programas con la comprensión de varias estrategias para implementar condicionales.
- Convertir sentencias condicionales de lenguaje hablado y bucles en un programa.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a **Diario** **Think Spot - Diario de reflexiones**.

Enlaces

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

Para los estudiantes

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

[Hacer una copia](#)

Vocabulario

- **Condición** - algo que un programa verifica para ver si es verdadero antes de permitir una acción.
- **Condicionales** - sentencias que solo se ejecutan bajo ciertas condiciones.
- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.
- **Until** - A command that tells you to do something only up to the point that something becomes true.

Guía docente

Actividad previa (10 min)

Introduction

In this lesson, students will be creating loops that only run `until` a condition is true. Help the students understand how this works by leading them in group activities and having them do an action `until` some condition is true. *For example: Have students touch their nose until you tell them to stop.*

Actividad Puente (15 mín)

Preview of Online Puzzles

Pull up a puzzle. We recommend Puzzle 4.

- Ask the class what the bird should repeat to get to the pig.
 - The bird should repeat `move forward, turn right, move forward, and then turn left`.
- Ask the class what they can use to repeat this code.
 - The bird should repeat this pattern *until* it reaches the pig.

Fill in the rest of the code using the `repeat until` loop and press `Run`. Discuss with the class why this worked.

Actividad Principal (30 min)

Online Puzzles

Bringing together concepts is not easy, but this set of lessons is meant to help students see the endless possibilities of coding when using conditions. If students struggle at all with understanding the similarities or differences between `while` loops and `until` loops, have them try to think of how they would use similar statements in their real lives.

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- How do you feel about today's lesson?
- What's the difference between an `until` loop and a `while` loop?

Standards Alignment

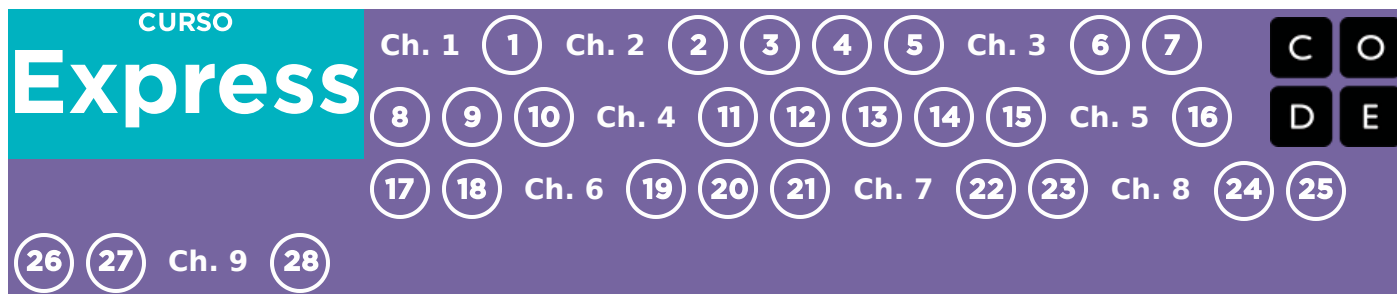
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 15: Cosechar con condicionales

Condicionales

Descripción

Los/as estudiantes practicarán loops `while`, loops `hasta`, y las declaraciones `if / else`. Todos estos bloques usarán condicionales. Al practicar los tres, los/as estudiantes aprenderán a escribir código complejo y flexible.

Propósito

Practicar el uso de condicionales en diferentes escenarios ayuda a desarrollar la comprensión de un/a estudiante de lo que puede hacer. En la lección anterior, los/as estudiantes sólo usaron condicionales para moverse alrededor de un laberinto. En esta lección, los/as estudiantes usarán condicionales para ayudar al agricultor a saber cuándo cosechar cultivos. Los nuevos patrones surgirán y los estudiantes usarán creatividad y pensamiento lógico para determinar las condiciones en las que el código debe ejecutarse y repetirse.

Agenda

Actividad previa (5 min)

Introducción

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 min)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Anidar condicionales para analizar condiciones de valor múltiple mediante la lógica, `si`, `si no si`, `si no`.
- Integrar un bucle y una sentencia condicional.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
 - ☐ Make sure every student has a **Diario**
- Think Spot - Diario de reflexiones.**

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Condición** - algo que un programa verifica para ver si es verdadero antes de permitir una acción.
- **Condicionales** - sentencias que solo se ejecutan bajo ciertas condiciones.
- **Bucle** - la acción de hacer algo una y otra vez.
- **Repetir** - hacer algo de nuevo.
- **Bucles mientras** - un bucle que se sigue repitiendo mientras una condición es verdadera.

Guía docente

Actividad previa (5 min)

Introducción

Los/as estudiantes no deberían necesitar una introducción a los conceptos de hoy, pues han practicado con ellos/as en la lección previa. En lugar de eso, puede compartir la historia de la granjera. La granjera está tratando de cosechar los cultivos como el zapallo, lechuga y maíz. Sin embargo, la granjera ha olvidado dónde plantó estos cultivos, así que necesita revisar cada planta antes de cosechar.

Actividad Principal (30 min)

Online Puzzles

Students will continue to work with `if / else` statements, `while` loops, and `until` loops. These puzzles are a bit more challenging, though, so encourage students to stick with them until they can describe what needs to happen for each program.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Cómo puedes ver que los condicionales son útiles en los programas?
- ¿Qué pasaría si la gente sólo hablara en declaraciones de `si/si no`? ¿Cuáles serían algunas de las ventajas y desventajas de esto?

Standards Alignment

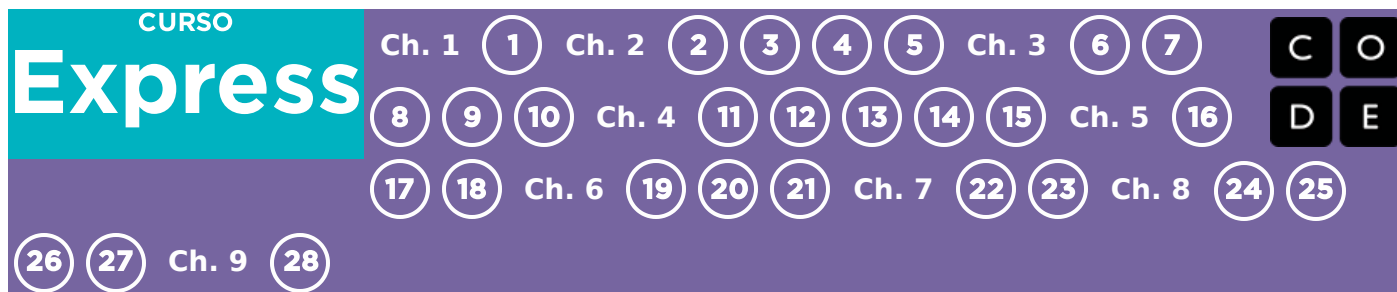
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 16: Functions in Minecraft

Funciones

Descripción

Students will begin to understand how functions can be helpful in this fun and interactive Minecraft adventure!

Propósito

Students will discover the versatility of programming by practicing functions in different environments. Here, students will recognize reusable patterns and be able to incorporate named blocks to call pre-defined functions.

Agenda

Actividad previa (10 min)

Introduction

Teaching this course as a class?

Actividad Puente - Funciones (15 mín)

Preview of Online Puzzles

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

[Ver en Code Studio](#)

Objetivos

Los estudiantes podrán:

- Usar funciones para simplificar programas complejos.
- Utilizar funciones predeterminadas para completar tareas que se repiten mucho.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a **Diario**

Think Spot - Diario de reflexiones.

Enlaces

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

Para los estudiantes

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

[Hacer una copia](#)

Vocabulario

- **Función** - código que puedes llamar fácilmente una y otra vez.

Guía docente

Actividad previa (10 min)

Introduction

🎓 Help the class understand that functions are simply a chunk of code that has a name. Once defined, you can use that name over and over in your program to tell the computer to run the chunk of code that you assigned to it.

Content Corner

Teaching this course as a class?

Our grade-aligned CS Fundamentals courses use unplugged lessons to build community and introduce tricky computer science concepts, including **functions**. Check out the lesson **Songwriting** from **Course E**!

Actividad Puente - Funciones (15 mín)

Preview of Online Puzzles

Pull up a puzzle from the lesson. We recommend puzzle 9 of this lesson. As a class, work through the puzzle without using functions. Once you have gotten the solution, display it on a white board or overhead. Ask the class to point to the repeated code. Ask the class how they would simplify the program. Why can you not just use a loop?

On the white board or overhead, rewrite the program without the repeated code, but leaving one line space. In that/those line space(s), call a function. Off to the side, declare the function like the left example block in the lesson tip. Ask the class what they think the code will do now.

Open up a discussion with the class on why functions could be useful in programming. Invite students to discuss the difference between functions and loops.

Actividad Principal (30 min)

Online Puzzles

We recommend providing paper and pencils for students to write (or draw) out ideas. Also, if students are having trouble recognizing patterns, have them work with a partner on the harder puzzles.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

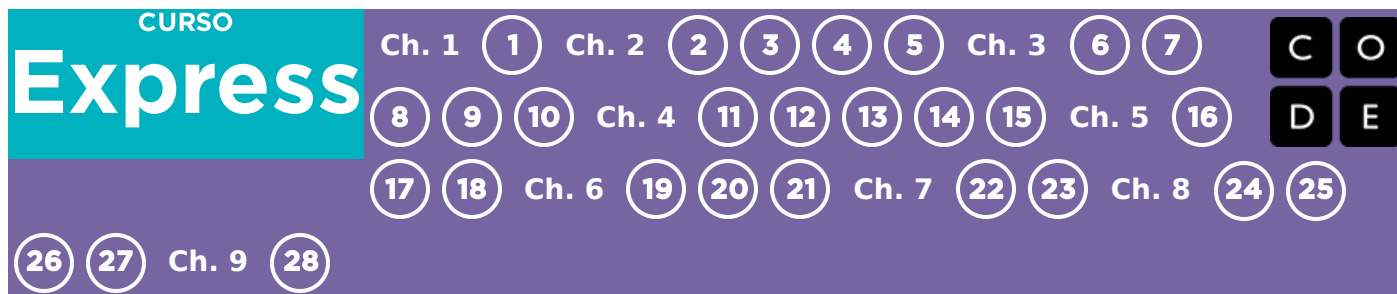
- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué hicieron tus funciones en los programas que escribiste hoy? ¿Cómo te ayudó eso?
- ¿Cuándo se debería utilizar una función en lugar de un loop?

Standards Alignment



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

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



Lección 17: Functions with Harvester

Funciones

Descripción

Los/as estudiantes han practicado creando diseños maravillosos en Abeja, pero hoy usarán las funciones para cosechar los cultivos en Granjera. Esta lección los llevará a usar las funciones en formas nuevas combinándolas con los “loops mientras” y las afirmaciones “sí/ si no”.

Propósito

Esta lección está preparada para llevar a los/as estudiantes más allá en el uso de las funciones en formas más creativas. Usando también los condicionales y los loops, aprenderán que hay muchas formas de enfocar un problema, pero algunas son más eficientes que otras. Estos desafíos están hechos para mejorar las habilidades de resolución de problemas y el pensamiento crítico.

Agenda

Actividad previa (10 min)

Introducción

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 min)

Escribir en el Diario

[Ver en Code Studio](#)

Objetivos

Los estudiantes podrán:

- Reconocer cuándo una función podría ayudar a simplificar un programa.
- Utilizar funciones predeterminadas para completar tareas que se repiten mucho.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a **Diario**

Think Spot - Diario de reflexiones.

Enlaces

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

Para los estudiantes

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

[Hacer una copia](#)

Vocabulario

- **Función** - código que puedes llamar fácilmente una y otra vez.

Guía docente

Actividad previa (10 min)

Introducción

En este punto, sus estudiantes deberían ser introducidos a las funciones. Tómese este tiempo para que discutan las ventajas y desventajas de usar funciones en un programa. Hágalos discutir en parejas o como clase. Intenta usar ejemplos de desafíos difíciles y fáciles en Artista o Abeja.

Pregunte a la clase:

- ¿Cuándo usarías una función?
- ¿Por qué una función ayuda a simplificar tu programa?
- ¿Crees que las funciones hacen la programación más fácil o más difícil? ¿Por qué?

Actividad Principal (30 min)

Online Puzzles

Some puzzles will have a function pre-declared for the students to fill in. It may be helpful for the students to write the entire program without a function first, then determine where a function would be useful in the program.

It's important to make sure that every student is completing each puzzle with a dark green dot. If some of your students are struggling to simplify code and use functions, set up teams of expert students within your class to go around and answer questions.

Don't forget to provide pencils and paper to help students sketch out possible solutions.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué hace que te des cuenta de que una función podría ayudar a tu programa?
- ¿Cómo ayudan a tu programa los loops mientras y las afirmaciones si / si no ?

Standards Alignment

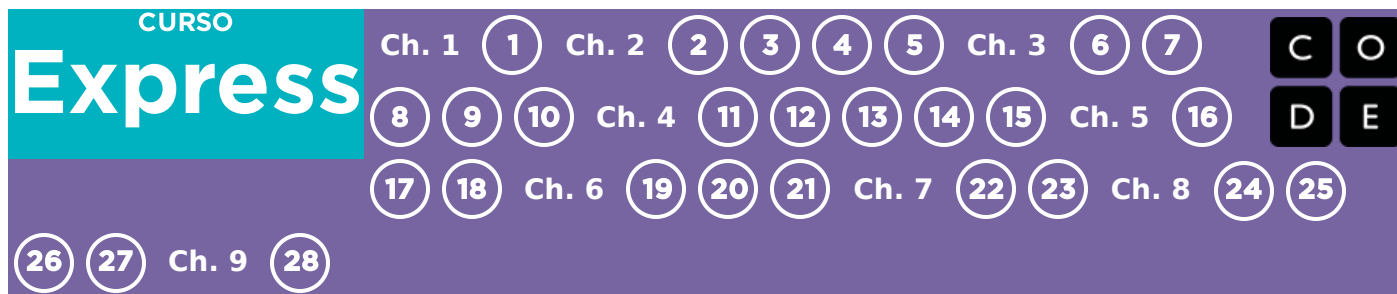
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 18: Functions with Artist

Funciones

Descripción

Se introduce a los/as estudiantes al uso de las funciones en Code.org. Se crearán y modificarán imágenes extraordinarias con funciones en Artista. Para patrones más complejos, los/as estudiantes aprenderán acerca de anidar funciones llamando una función desde otra función.

Propósito

Uno de los componentes más importantes de esta lección es proporcionar a los/as estudiantes un espacio para crear algo de lo que se sientan orgullosos/as. Estos desafíos progresan a imágenes más y más complejas, pero cada desafío sólo se construye a partir del desafío anterior. Al final de esta lección, los/as estudiantes sentirán confianza en ellos/as mismos/as y estarán orgullosos/as de su trabajo.

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through puzzles in the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Función** - código que puedes llamar fácilmente una y otra vez.

Guía docente

Actividad previa (15 mín)

Introduction

Tell the class that there are two main components to using functions.

1. **The Declaration:** Function declarations are what create a function. In a function declaration, you fill in the function with code and you give the function a name. You must declare a function before you can use it.
2. **The Call:** Function calls are what makes the program run the code in the function. To call a function, you place the name of the function in your program. Make sure your function is properly defined before calling it in your program.

Continue the conversation until students have a basic understanding of functions being declared and called. If students don't get to this point, make sure to do one of the bridging activities before moving into the Code.org puzzles.

Actividad Principal (30 min)

Online Puzzles

Students may benefit from writing code without functions then creating functions from the repeated code. If students don't enjoy doing this in the Code.org workspace, we recommend providing paper and pencils for students to write (or draw) out their ideas.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Cuáles son algunas de las diferencias entre las funciones y los loops?
- Haz un dibujo que hiciste hoy. ¿Puedes escribir el código necesario para crear esto?
- Haz un dibujo que te gustaría crear con código. Intenta escribir o redactar el código que haría ese dibujo.

Aprendizaje Extendido

Dibujar con Funciones

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

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

Standards Alignment

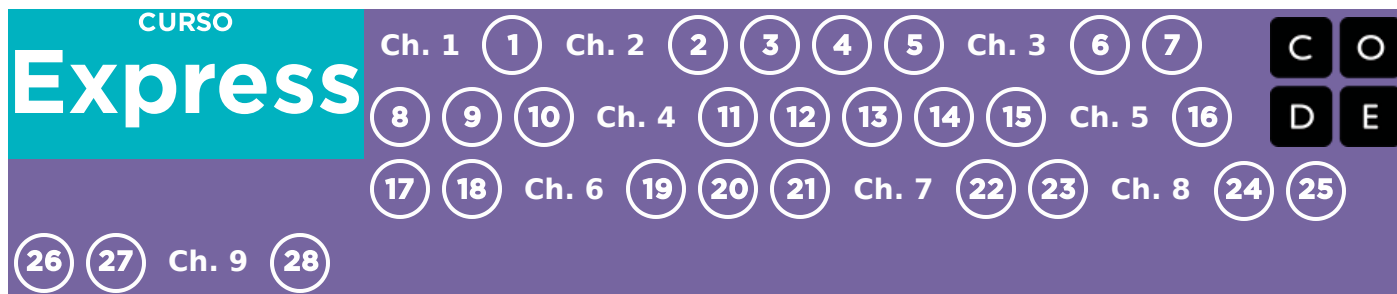
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 19: Variables with Artist

Variables

Descripción

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

Propósito

This stage teaches the most basic use for variables, as a constant that reoccurs frequently in a program.

Agenda

Actividad previa (15 mín)

Introduction

Teaching this course as a class?

Actividad Puente - Variables (15 mín)

Preview of Online Puzzles as a Class

Actividad Principal (30 mín)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Constant** - A variable used throughout a program that never changes value
- **Variable** - un marcador de posición para una información que puede cambiar.

Guía docente

Actividad previa (15 mín)

Introduction

Discussion:

- What is a variable? (A placeholder for a piece of information that can change.)
- When can a variable be helpful? (When you don't know what information is going to be used in a certain place until runtime, or when you have lots of places that one piece of information will be used, but that information might change someday.)

Ask the class when they could see a variable being helpful in programming. When would they NOT want to use a variable?

Content Corner

Teaching this course as a class?

Our grade-aligned CS Fundamentals courses use unplugged lessons to build community and introduce tricky computer science concepts, including **variables**. Check out the lesson **Envelope Variables** from **Course F!**

Actividad Puente - Variables (15 mín)

Preview of Online Puzzles as a Class

Demo: Display a puzzle for the class. We recommend the 6th puzzle. Go over the code with the students to make sure they understand what's happening before they help you convert the code to use variables. Can they think of something that might happen that would make them really glad that they used variables instead of hardcoded numbers?

Transition: Now it's time for your students to move to their own machines and get started!

Actividad Principal (30 min)

Online Puzzles

Notice that this stage first covers the idea of a variable as a constant (a variable that you use in many places, but it does not change.) This might be something that students find helpful as they're creating their own projects.

Watch out for puzzle #6. It is the first time that students will be expected to set a variable on their own. This can be tricky if they don't have a true grasp on the concept. If they're having trouble, send them back to the prediction level (#5) and have them explain to their partners why the answer ended up what it was. Once both partners are convinced, let them continue back to puzzle #6.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué es una variable? ¿Por qué es útil en la programación?

- ¿Qué tan bien crees que entiendes las variables? (Responde en una escala del 1 al 5 o con un emoticón.) Si tienes problemas, ¿puedes poner en palabras lo que no entiendes?

Standards Alignment

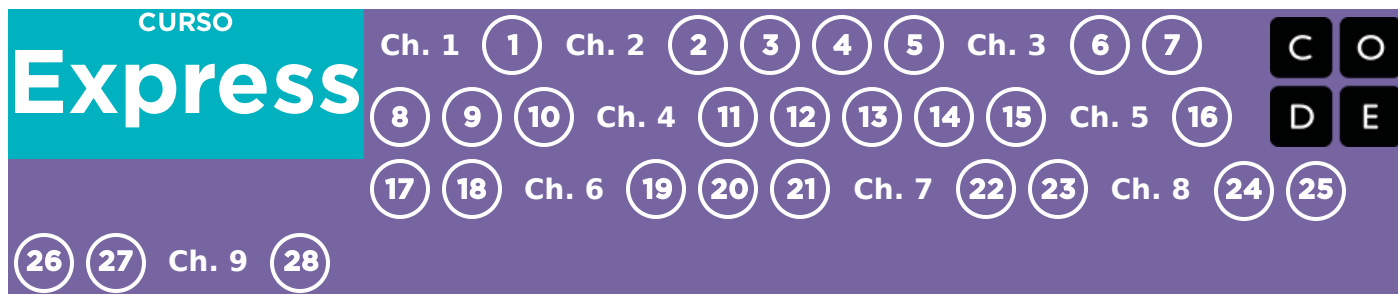
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 20: Changing Variables with Bee

Variables

Descripción

This lesson will help illustrate how variables can make programs more powerful by allowing values to change while the code is running.

Propósito

You don't always know what a value is going to be before you begin your program. Sometimes, values change while your code is running. This lesson will illustrate how code with changing values can be helpful.

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Identify areas where they can use variables to modify quantities during runtime.
- Examine code to find places where variables can be substituted for specific values.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

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

Guía docente

Actividad previa (15 mín)

Introduction

This series is a little different than what students have done in the past. Now, instead of simply assigning a value to a variable and running your code, you'll need to help students see how a variable can be modified during program runtime.

Display Show students the play area from one of the later puzzles.



There are several things to unpack here, so you might need to give your students a chance to look at it critically before you expect them to do anything with it.

Think/Pair: Suppose all of the flowers/honeycomb in this picture had the same amount of nectar/honey. How would you solve this puzzle?

Share: Let students share their ideas until you decide on one that the entire class is good with.

Discuss: Now, imagine that we didn't want to have to write a separate chunk of code for each set of flowers and honeycomb. How could we use a variable to have our loop do this for us?

(Eventually, you'll want to get to the place where you initialize a variable to the original value, then change it each time through the loop so that it's ready for the next time.)

Actividad Principal (30 min)

Online Puzzles

This set of puzzles takes some serious computational thinking skills. If you find that students are getting stuck, help them break down the puzzles into the individual pieces:

- What would it look like if the flowers/honeycomb all had the same amount of nectar/honey?
- What would it look like without the functions?
- Now how can you use a variable to get the quantities the way you want them?
- Now can you build it back up to use a function?

Hint: Puzzle 7 becomes much easier if students utilize the `while` path ahead loop instead of a variable.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el diario:

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

Standards Alignment

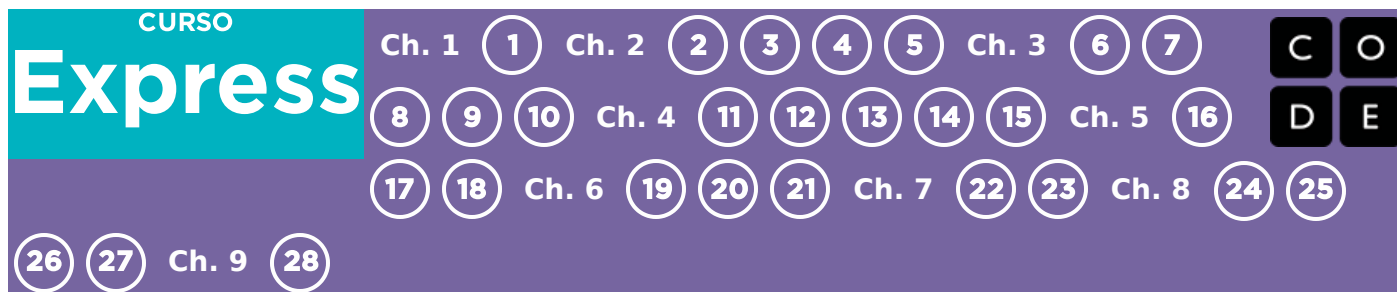
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 21: Changing Variables with Artist

Variables

Descripción

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

Propósito

Variables are essentially placeholders for values that might be unknown at the time that you run your program or for values that can change during the execution of a program. These are vital to creating dynamic code because they allow your program to change and grow based on any number of potential modifications. This stage reinforces the use of variables, using the most basic capabilities of setting and using them.

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

- **Variables** - Video para el estudiante ([descargar](#))
- **Diario Think Spot** - Diario de reflexiones

[Hacer una copia](#)

Vocabulario

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

Guía docente

Actividad previa (15 mín)

Introduction

It might be helpful to remind students of "Variables that Change in Bee," since variables will be used in a similar way here.

- How can we change the value of a variable inside of a loop?
- Do we have to change the value of a variable only by one each time?

Actividad Principal (30 min)

Online Puzzles

The latter half of this series is made up of freeplay puzzles. Students will have the opportunity to play with variables, shape, and color to create something unique.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Standards Alignment

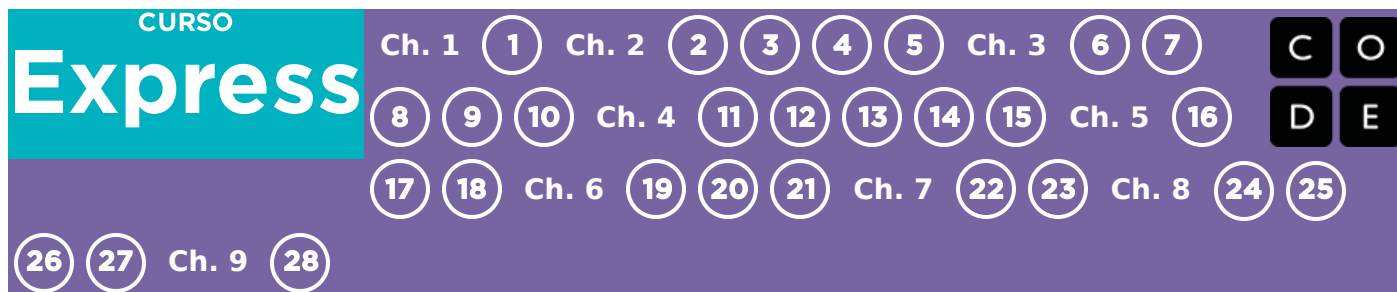
Estándares para ciencias de la computación CSTA K-12 (2017)

- AP - Algoritmos y programación



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

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



Lección 22: For Loops with Bee

Bucles desde

Descripción

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

Propósito

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

Agenda

Actividad Puente - Loops Para (15 mín)

Previewing Online Puzzles as a Class

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

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

Guía docente

Actividad Puente - Loops Para (15 mín)

Previewing Online Puzzles as a Class

Display a puzzle from the lesson. We recommend puzzle #4 because it displays a potential solution and asks the user to evaluate it.

Using a number line, mark the start and ending values of the given `for` loop (if you aren't using puzzle #4, you will need to come up with a potential solution first). With the class's help, circle the values between the start and end that the `for` loop will touch. If you are working on puzzle #4, ask the class what they think the answer is to the question, given what they found with the number line.

Actividad Principal (30 min)

Online Puzzles

Some students may have a hard time differentiating between `repeat` loops and `for` loops. We recommend having scratch paper out for students to make guesses on values like the start, stop, and step. Implementing pair programming amongst the class might also be helpful for your students.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Standards Alignment

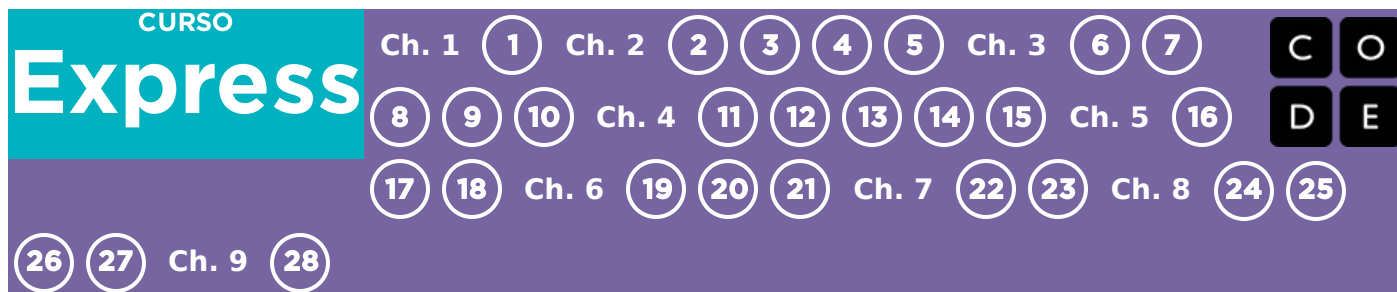
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 23: For Loops with Artist

Bucles desde

Descripción

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

Propósito

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

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (30 mín)

Online Puzzles

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find and potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

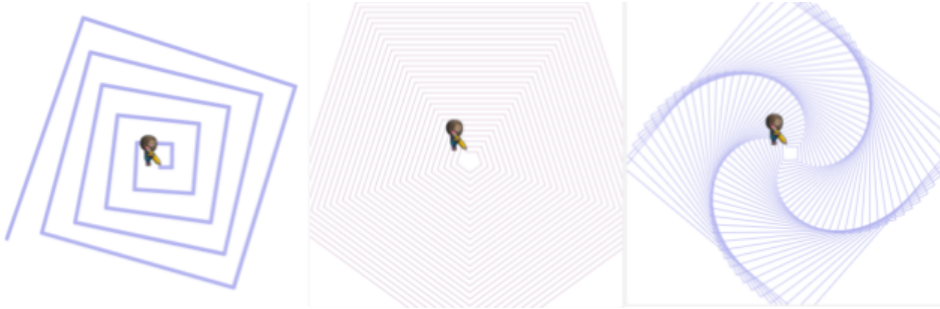
- **Bucles desde** - bucles que tienen un comienzo, un final y un incremento predeterminados (intervalo de paso).

Guía docente

Actividad previa (15 mín)

Introduction

On a board displayed to the entire class, draw (or display via projector) one of the final projects from the lesson. We recommend one of the following:



Ask the class how a computer might draw the drawing you displayed.

After a few predictions have been said, reply with `for` loops of course!

Tell the students they will soon be learning how to create these fine drawings using `for` loops and variables.

Actividad Principal (30 min)

Online Puzzles

These puzzles are super fun, but it may be helpful for students to have protractors and scratch paper to see these designs made in the physical form. If that isn't an option in your class, try to get the students to trace on the computer screen with their fingers.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Standards Alignment

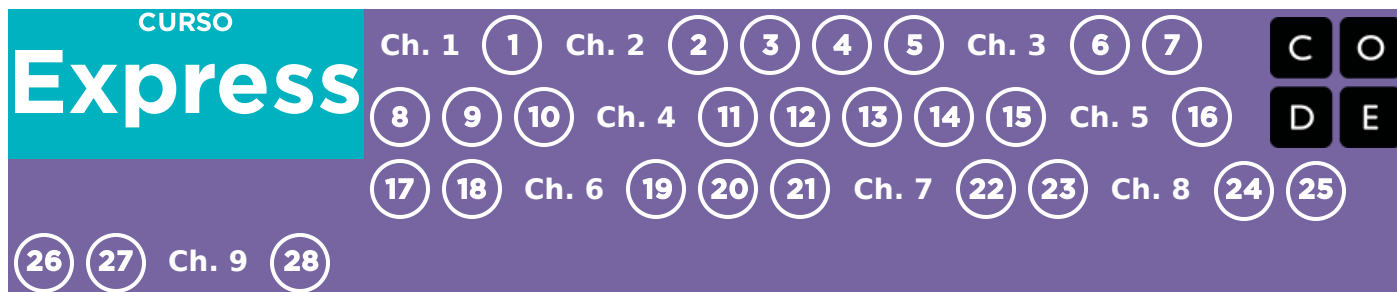
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 24: Swimming Fish in Sprite Lab

Sprites

Descripción

In this lesson, students will learn about the two concepts at the heart of Sprite Lab: sprites and behaviors. Sprites are characters or objects on the screen that students can move, change, and manipulate. Behaviors are actions that sprites will take continuously until they are stopped.

Propósito

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

Agenda

Actividad previa (10 min)

Introduction

Teaching this course as a class?

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 min)

Journaling

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Define “sprite” as a character or object on the screen that can be moved and changed.
- Create a new sprite and choose its appearance.

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Vocabulario

- **Behavior** - An action that a sprite performs continuously until it's told to stop.
- **Sprite** - A graphic on the screen with a location, size, and appearance.

Guía docente

Actividad previa (10 min)

Introduction

🎓 Today students will learn how to work with sprites in Sprite Lab.

Display: Pull up a previous puzzle from Code.org, ideally one containing a "main character" like Scrat from Ice Age or one of the Angry Birds.

Discuss: Let the students know that this character on the screen is a "sprite." It is a graphic that is controlled by a program. In this lesson, students will have the opportunity to choose their own sprites to control.

Display: Begin by showing Puzzle 1 to your students.

Think/Pair: Ask them to predict what will happen when the code is run, and to discuss with their neighbors. Run the code, and discuss the outcome.

Now is a great time to introduce the lesson vocabulary.

Content Corner

Teaching this course as a class?

Our grade-aligned CS Fundamentals courses use unplugged lessons to build community and introduce tricky computer science concepts, including **sprite behaviors**. Check out the lesson **Simon Says** from **Course E!**

Actividad Principal (30 min)

Goal: Today, students will be creating their own Fish Tank. They'll begin by learning how to put some sprites on the screen, then they will make them move. Finally, they'll customize their fish tank to add whatever creatures and objects they want.

Online Puzzles

🔗 **Transition:** Move students to their machines. Encourage students to follow the instructions for each puzzle. Help them realize that this is a creative activity, intended to help them learn Sprite Lab. It is not an assessment activity of any sort.

Reminder: If puzzles are sharable, remind the students to only share their work with their close friends or family. For more information watch or show the class **Detente y Piensa Online - Video**.

Teaching Tip

Encourage students with questions/challenges to start by asking their partner. Unanswered questions can be escalated to a nearby group, who might already know the solution. Have students describe the problem that they're seeing:

- What is it supposed to do?
- What does it do?
- What does that tell you?

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- How do you feel about today's lesson?
- How did it feel to make a scene that was more creative?

- Was it difficult to finish a lesson where there was no clear "right" and "wrong"?

Standards Alignment

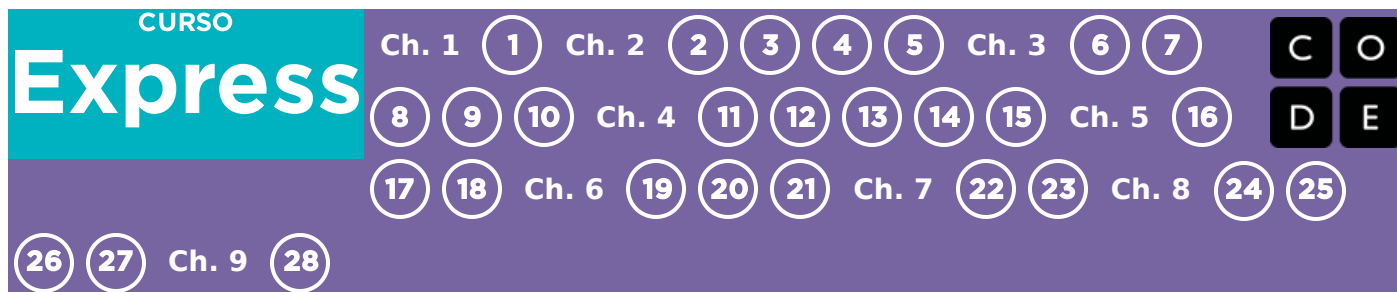
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 25: Alien Dance Party

Sprites

Descripción

This lesson features Sprite Lab, a platform where students can create their own alien dance party with interactions between characters and user input. Students will work with events to create game controls.

Propósito

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

Agenda

Actividad previa (10 min)

Introduction

Actividad Principal (30 min)

Online Puzzles

Actividad de Cierre (15 min)

Journaling

[Ver en Code Studio](#)

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the lesson to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

[Hacer una copia](#)

Vocabulario

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

Guía docente

Actividad previa (10 min)

Introduction

Today students will visit *events* in programming.

Demo: Ask the students to raise their hands in the air.

What you did was declare an event. When you say "raise your hands in the air" the students responded by raising their hands. In coding, you would declare this by saying something like "when I say 'raise your hands,' you raise your hands".

You can also think of cities as declaring events. There are laws that say "when there is a green light, cars move through the intersection".

Discuss: Ask the students why they think this is an event.

Today, students will play in Play Lab, but the events they will be working on will be more like the video games they are used to playing. Events will take the form of actions, such as clicking the screen or two characters running into each other.

Display: Begin by showing Puzzle 1 to your students.

Think/Pair: Ask them to predict what will happen when the code is run, and to discuss with their neighbors. Run the code, and discuss the outcome.

Actividad Principal (30 min)

Goal: Today, students will be creating their own alien dance party! They'll begin by reviewing how to put sprites on the screen, then they will assign them behaviors and learn to change those behaviors when an event is initiated.

Online Puzzles

🔗 **Transition:** Move students to their machines.

Encourage students to follow the instructions for each puzzle. Help them realize that this is a creative activity, intended to help them learn Sprite Lab. It is not an assessment activity of any sort.

Reminder: If puzzles are sharable, remind the students to only share their work with their close friends or family. For more information watch or show the class **Detente y Piensa Online - Video**.

💡 Teaching Tip

Encourage students with questions/challenges to start by asking their partner. Unanswered questions can be escalated to a nearby group, who might already know the solution. Have students describe the problem that they're seeing:

- What is it supposed to do?
- What does it do?
- What does that tell you?

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- How do you feel about today's lesson?
- How did it feel to have control over what your characters were able to do?
- Did you change the program in any way to make it feel more like your own?

Standards Alignment

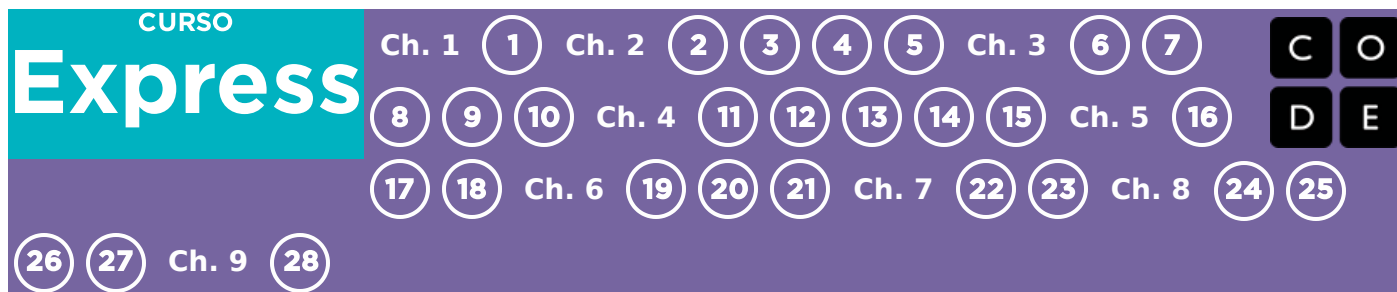
Estándares para ciencias de la computación CSTA K-12 (2017)

► **AP** - Algoritmos y programación



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

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



Lección 26: Behaviors in Sprite Lab

Sprites

Descripción

Here, students will use Sprite Lab to create their own customized behaviors.

Propósito

Students will use events to make characters move around the screen, change size, and change colors based on user input. This lesson offers a great introduction to events in programming and even gives a chance to show creativity!

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (20 min)

Online Puzzles

Actividad de Cierre (15 mín)

Journaling

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Play through the puzzles to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▾

Guía docente

Actividad previa (15 mín)

Introduction

Today students will revisit *sprite behaviors*, this time learning how to edit behaviors directly and even make new ones.

Review: Ask students questions about the Swimming Fish and Alien Dance Party lessons.

- What are some of the behaviors we can assign to our sprites?
- What do you imagine the code *inside* a behavior might look like?

Display: Begin by showing Puzzle 1 to your students.

Think/Pair: Ask them to predict what will happen when the code is run, and to discuss with their neighbors. Be sure to open the behavior editor by clicking "edit" on the `mystery behavior` block. Run the code, and discuss the outcome.

Discuss Ask students questions about how they might change this behavior's code to create a different effect.

- What would happen if you change the -1 to another number? -5? Positive 1? 0?
- What other properties might we be able to change about a sprite besides its size?

Actividad Principal (20 min)

Goal: Today, students will be editing and creating their own behaviors! They'll begin by making small changes to some familiar but new behaviors and gradually move towards writing their own behaviors from scratch.

Online Puzzles

🔗 **Transition:** Move students to their machines. Encourage students to follow the instructions for each puzzle. Help them realize that this is a creative activity, intended to help them learn Sprite Lab. It is not an assessment activity of any sort.

Reminder: If puzzles are sharable, remind the students to only share their work with their close friends or family. For more information watch or show the class **Detente y Piensa Online - Video**.

🔗 Teaching Tip

Encourage students with questions/challenges to start by asking their partner. Unanswered questions can be escalated to a nearby group, who might already know the solution. Have students describe the problem that they're seeing:

- What is it supposed to do?
- What does it do?
- What does that tell you?

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- How do you feel about today's lesson?
- What other options would you like to be able to have your pet do?

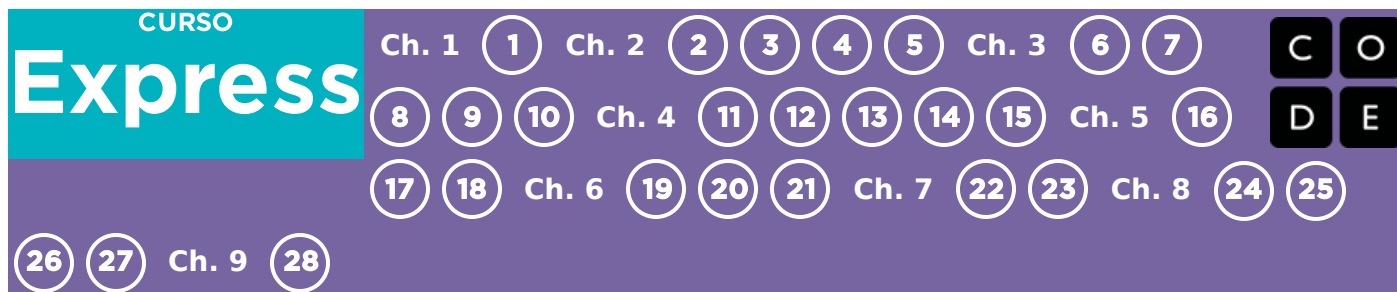
Standards Alignment

- **AP** - Algoritmos y programación



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

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



Lección 27: Virtual Pet with Sprite Lab

Sprites

Descripción

In this lesson, students will create an interactive Virtual Pet that looks and behaves how they wish. Students will use Sprite Lab's "Costumes" tool to customize their pet's appearance. They will then use events, behaviors, and other concepts they have learned to give their pet a life of its own!

Propósito

This lesson allows students to apply programming concepts from prior lessons in a more creative context. For example, students will use a variable to store their pet's "happiness" as an integer, which should help them understand how variables can be used in other applications. Last, completing this lesson should prepare students to engage with the open-ended final project.

Agenda

Actividad previa (15 mín)

Introduction

Actividad Principal (20 min)

Online Puzzles

Actividad de Cierre (15 mín)

Journaling

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Create an interactive virtual pet using events, behaviors, variables, and custom art.
- Program solutions to problems that arise when designing a virtual pet, like feeding it or monitoring its happiness.

Preparación

- ☐ Play through the puzzles to find any potential problem areas for your class.
- ☐ Make sure every student has a journal.

Enlaces

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

Para los estudiantes

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

Hacer una copia ▼

Guía docente

Actividad previa (15 mín)

Introduction

Revisit *events* and *behaviors* in programming. Additionally, introduce the Sprite Lab "Costumes" tool that allows students to draw their own costumes.

Review: Ask students questions about events and behaviors. Show them the Alien Dance Party mini-project (from the *Events with Sprite Lab* lesson) as an example.

- Do you remember what an event is?
- Can you name any of the events that you used to make the aliens dance? What do they do?
 - when clicked
 - when blue alien touches pink alien
 - when arrow pressed
- Do you remember what a behavior is?
- Can you remember some of the behaviors you used to make the aliens dance? What do they do?
 - patrolling
 - jittering
 - spinning right/left

Display: Begin by showing Puzzle 1 of today's lesson to your students.

Think/Pair: Ask students to predict what will happen when the code is run, and to discuss with their neighbors. Run the code, and discuss the outcome.

Display: Show Puzzle 2. Briefly demonstrate how to do the following:

- Navigate between the *Code* and *Costumes* tabs.
- Draw a costume.
- Choose a costume from the costume library.
- Change the virtual pet's sprite's costume to a custom one.

Actividad Principal (20 min)

Goal: Today, students will be creating their own virtual pet! They will begin by drawing or selecting a new costume for a sprite. Then they will create events that cause actions and behaviors upon interaction.

Online Puzzles

🔗 **Transition:** Move students to their machines. Encourage students to follow the instructions for each puzzle. Help them realize that this is a creative activity, intended to help them learn Sprite Lab. It is not an assessment activity of any sort.

Reminder: If puzzles are sharable, remind the students to only share their work with their close friends or family. For more information watch or show the class **Detente y Piensa Online - Video**.

💡 Teaching Tip

Encourage students with questions/challenges to start by asking their partner. Unanswered questions can be escalated to a nearby group, who might already know the solution. Have students describe the problem that they're seeing:

- What is it supposed to do?
- What does it do?
- What does that tell you?

Actividad de Cierre (15 mín)

Journaling

Having students write about what they learned, why it's useful, and how they feel about it can help solidify any knowledge they obtained today and build a review sheet for them to look to in the future.

Journal Prompts:

- What was today's lesson about?
- How do you feel about today's lesson?
- What other options would you like to be able to have your pet do?

Standards Alignment

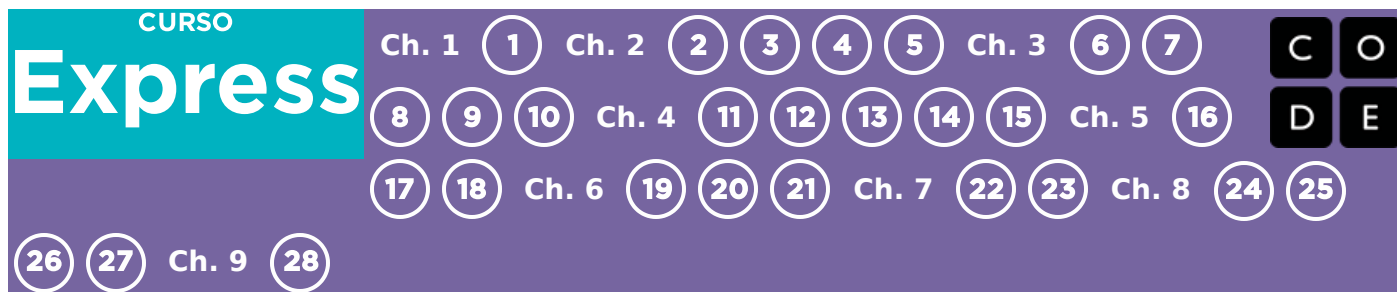
Estándares para ciencias de la computación CSTA K-12 (2017)

- **AP** - Algoritmos y programación



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

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



Lección 28: Proyecto de fin de curso

Proyecto de fin de curso

Descripción

This capstone lesson takes students through the process of designing, developing, and showcasing their own projects!

Propósito

This lesson provides students with space to create a project of their own design, using a step-by-step process that requires planning but also allows for broad creativity.

Agenda

Actividad previa (10 min)

Planning

Actividad Principal (25 min)

Coding

Actividad de Cierre (10 min)

Showcase

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Superar los obstáculos, como limitaciones de tiempo o errores.

Preparación

- ☐ Spend time making your own project with both the project tools available to students. Familiarize yourself with the capabilities and limitations of each tool.
- ☐ Modify the rubric to fit your class goals and print one copy for each student or group.
- ☐ Modify the project planning guide to fit your class and print one for each student or group.

Enlaces

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

Para los docentes

- **Proyecto Final de Fundamentos de Ciencias de la Computación** - Rúbrica

Para los estudiantes

- **CSF Express End of Course Project Planning Guide** - Project Guide

Hacer una copia ▾

Vocabulario

- **Definir** - descubrir los detalles de los problemas que estás tratando de resolver.
- **Preparar** - investigar, planificar y adquirir materiales para la actividad que estás a punto de realizar.

- **Reflexionar** - pensar cuidadosamente en algo con la intención de mejorar el resultado en el futuro.
- **Probar** - intentar hacer algo.

Guía docente

Actividad previa (10 min)

Planning

Get students excited and ready for today's activity!

Remarks

We have already had a chance to build a variety of projects. Today, this experience will be much more open-ended, so it will require planning beforehand! Planning is a very important part of coding a game or any other kind of software. So, before we jump onto computers, we will spend some time planning the projects we want to build.

 **Distribute:** Distribute one **CSF Express End of Course Project Planning Guide** to each student or pair. With students, go over the steps listed on the guide, then allow them to complete it. Refer to the included exemplar if needed.

Teaching Tip

If students are pair programming for this assignment, this warm up is a great opportunity for them to practice sharing and respecting others' ideas. Ensure students are following group work norms you already have in place in your classroom. Otherwise, spend a brief moment going over your expectations.

Actividad Principal (25 min)

Coding

 Equipped with their completed planning guides, students are now ready to bring their projects to life. These levels correspond to the structure of the planning guide, and help navigate students through the process of transforming their ideas into code.

Teaching Tip

Students will experience plenty of trial and error while coding. Their projects are likely to become truncated versions of their original scope. Remind students that this kind of compromise is common in software design. It's okay if they don't get to build in every feature they planned!

Code Studio levels

Example Projects

 1

(click tabs to see student view)

Sprite Lab Project

 2

(click tabs to see student view)

Artist Project

 3

(click tabs to see student view)

Actividad de Cierre (10 min)

Showcase

To celebrate students' work, spend the last 10 minutes or so allowing them to showcase their projects. This can be done in many ways, but here are a few:

- **Public Demo:** Select a few exemplary volunteers to briefly demo their projects in front of the class. As they do so, have them touch on what the planning-to-coding experience was like for them, including ideas they'd still like to implement.

- **Pair Playtesting:** Have students or groups pair up and playtest each other's projects. As they do, ask them to provide positive and constructive feedback to each other. The benefit here is that students will have the opportunity to provide and respond to feedback in a smaller setting.
- **Gallery Walk:** Ensure all students have their projects ready for testing. Have students move "musical chairs"-style to another computer and playtest the project there for a few minutes, until they receive a signal from you to move to another computer. Repeat this every few minutes. While there is less opportunity for structured communication here, this ensures students get to demo as many of their peers' projects as possible.

Standards Alignment

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

- **AP** - Algoritmos y programación



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

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