

Curso D

El curso D se creó para estudiantes que leen aproximadamente al nivel de 4° básico. Se introducen ángulos y conceptos matemáticos con la ayuda de videos y consejos útiles. El curso comienza con una revisión de los conceptos de los Cursos A, B y C. Esta revisión ayuda a introducir o refrescar ideas básicas como los loops repetir y los eventos. Los/as estudiantes desarrollarán la comprensión de los algoritmos, loops anidados, loops mientras, los condicionales y los eventos. También se incluyen las lecciones sobre ciudadanía digital. Este curso está diseñado para construir unos cimientos fuertes de conceptos básicos antes de abrirse a un amplio rango de temas nuevos y emocionantes. Links para el/la profesor/a: [Lista de Videos para el/la Profesor/a](#)

Lección 1: Programación con Papel Cuadrulado

¡En esta lección, programarás a tu amigo/a para dibujar!

Lección 2: Introducción a los Desafíos Online

[Sequencing](#) | [Debugging](#) | [Loops](#) | [Angry Bird](#) | [Collector](#) | [Artist](#) | [Harvester](#)

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

Lección 3: Eventos en Bounce

[Event](#) | [Bounce](#)

¿Alguna vez has pensado poder jugar videojuegos en la escuela? ¡En esta lección, harás tu propio videojuego!

Lección 4: Loops Anidados

[Nested Loops](#) | [Loops](#) | [Bee](#) | [Maze](#)

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

Lección 5: Loops Anidados en Artista

[Nested Loops](#) | [Loops](#) | [Artist](#)

¡Más loops anidados! Esta vez, harás algunos SORPRENDENTES dibujos con loops anidados.

Lección 6: Loops Anidados con Frozen

[Loop](#) | [Nested Loop](#) | [Frozen](#)

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 7: Programación por Relevos

[Unplugged](#) | [Relay Programming](#) | [Algorithms](#)

¿Recuerdas al comienzo del curso cuando hiciste dibujos con código? ¡En esta lección, trabajarás con un equipo para hacer algo muy similar!

Lección 8: Debugging en Recolectora

[Debugging](#) | [Bug](#) | [Collector](#) | [Laurel](#)

¿Alguna vez has tenido problemas al programar? En esta lección, aprenderás sobre los secretos de debugging. Debugging es el proceso de encontrar y arreglar problemas en tu código.

Lección 9: Loops Mientras en la Granjera

While Loops | Loops | Farmer

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

Lección 10: Si/si no: Condicionales con Cartas

Conditionals | Unplugged

¡Es hora de jugar un juego donde ganas puntos sólo bajo ciertas condiciones!

Lección 11: Condicionales en Abeja

Conditional | Bee | Maze

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

Lección 12: Condicionales & Loops en Laberinto

Conditional | Loop | Maze | Angry Bird | Zombie

¡Puedes hacer algunas cosas asombrosas cuando usas condicionales y loops juntos!

Lección 13: Condicionales & Loops en Cosechadora

Conditional | Loop | Harvester

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

Lección 14: Ciudadanía Digital

Common Sense Education | Unplugged

Algunas informaciones no son seguras para compartir online. Esta lección te ayudará a aprender la diferencia entre información segura y privada.

Lección 15: Construir un Juego de Play Lab

Play Lab | Event

Esta lección te guiará haciendo tu propio videojuego.

Lección 16: Más Allá de a Programación: Binario

Binary | Unplugged

Aprende cómo las computadoras almacenan imágenes usando ideas simples como encendido y apagado.

Lección 17: Artista Binario

Binary | Artist

En esta lección, aprenderás a crear imágenes usando on y off.



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

Español (Latinoamérica) ▼

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

Lección 1: Programación con Papel Cuadrulado

Descripción

“Programándose” unos a otros para hacer dibujos, los/as estudiantes comenzarán a comprender de qué se trata realmente codificar. La clase comenzará haciendo que los/as estudiantes se den instrucciones mutuamente para colorear cuadrados en un papel cuadrulado, en un esfuerzo por reproducir un dibujo ya existente. Si hay tiempo, la lección puede concluir con imágenes que los/as estudiantes creen por sí mismos/as.

Propósito

El objetivo de esta actividad es desarrollar las habilidades de pensamiento crítico y entusiasmo por el curso. Presentando conceptos básicos como programar y algoritmos a la clase en una actividad sin conexión, los/as estudiantes que se sienten intimidados por los computadores pueden aún construir una base de comprensión de estos temas. Programar y algoritmos son esenciales en la ciencia de la computación. En esta lección, los/as estudiantes aprenderán cómo transformar las instrucciones en un programa y a reconocer un algoritmo.

Agenda

Actividad previa (10 min)

Introducción a la programación en papel cuadrulado

Actividad Principal (30 min)

Practicar juntos

El turno de los/as estudiantes

Actividad de Cierre (15 min)

Escribir en el Diario / Conversación Rápida

Evaluación Opcional (10 min)

Programación con Papel Cuadrulado - Evaluación

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Replantear una secuencia de pasos como un programa codificado
- Explicar las limitaciones de traducir problemas del lenguaje humano al lenguaje de máquina

Preparación

- Vea [teacher-video-graph-paper-2][0].
- Vea [lesson-in-action-graph-paper][1].
- Imprima una copia [csf-graph-paper-programming-worksheet][2] para cada grupo.
 - Imprima una copia [csf-graph-paper-programming-assessment][3] para cada estudiante.
 - Entregue a cada grupo copias suficientes de papeles cuadrulados, lápices y lapiceras.
 - Asegúrese que cada estudiante tenga un [think-spot-journal][4].

Enlaces

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

Para los docentes

- **Programación con Papel Cuadrulado** - Video de lección en acción
- **Programación con Papel Cuadrulado** - Guía de respuestas en fichas

Hacer una copia ▾
- **Programación con Papel Cuadrulado** - Guía de respuestas de la evaluación

Hacer una copia ▾

Para los estudiantes

- **Programación con Papel Cuadrulado** - Video sin conexión (**descargar**)

- **Programación con Papel Cuadrulado** - Ficha
- **Programación con Papel Cuadrulado** - Evaluación
- **Diario Think Spot** - Diario de reflexiones

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.

Guía docente

Actividad previa (10 min)

Introducción a la programación en papel cuadriculado

En esta actividad, los/as estudiantes codificarán instrucciones para guiarse entre sí hacia la realización de dibujos sin permitir que el resto de su grupo vea la imagen original. Esta actividad previa enmarca la actividad para la clase.

**** Mostrar: **** Mire uno de los videos a continuación para darles a lo/as estudiantes el contexto de los tipos de cosas que pueden hacer los robots:

- **Asimo por Honda** (3:58)
- **Robot de dibujo de huevos** (3:15)
- **Dancing Lego Robot** (1:35)

🗨️ <! - lugar donde le gustaría que aparezca el icono ->

**** Discutir: **** ¿Cómo supones que los robots saben cómo hacer las cosas que hacen? ¿Tienen cerebros que funcionan de la misma manera que lo hacen los nuestros?

Trabaje esto en una discusión sobre cómo las personas tienen que programar robots para hacer cosas específicas, utilizando comandos específicos. .

Discussion Goal

El objetivo de esta rápida discusión es señalar que mientras los robots parecen comportarse como personas, en realidad sólo responden a su programación. Los estudiantes probablemente se referirán a robots de películas y TV que se comportan más como seres humanos. Pídales que consideren los robots que han visto o escuchado en la vida real, como Roombas, o incluso asistentes digitales como Amazon Alexa.

Actividad Principal (30 min)

Practicar juntos

En esta actividad, los estudiantes actuarán como programadores y robots, coloreando cuadrados de acuerdo con los programas que han escrito entre ellos.

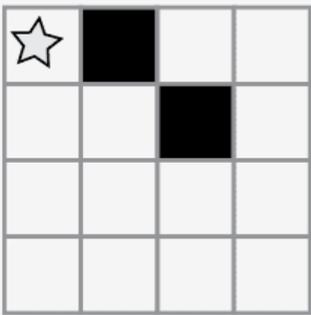
**** Distribuir: **** Los estudiantes usarán cuadrículas de 4x4 (u hojas de papel cuadriculado con cajas de 4x4 cortadas). También necesitarán la hoja de trabajo de la imagen.

**** Mostrar: **** Projete estos comandos, o escríbalos en la pizarra. No persistirán por mucho tiempo, pero ayudarán a los estudiantes a hacer la transición del algoritmo al programa.

Mueve un cuadrado a la derecha Mueve un cuadrado a la izquierda Mover una casilla hacia arriba Mover una casilla hacia abajo Rellenar cuadrado con color.

**** Decir: **** Hoy, todos podemos programar robots... ¡y ya están aquí en la sala! ¡Eres tu! Vamos a escribir programas usando símbolos con significados especiales para ayudarnos mutuamente a recrear una imagen. Primero, practicaremos juntos como si yo fuera el robot y ustedes los programadores, luego podemos dividirnos en grupos para que todos puedan obtener un turno.

**** Mostrar: **** Muestre tanto la imagen que va a hacer que los/as estudiantes lo guíen como una cuadrícula en blanco que llenará con su ARM. Asegúrese de que las instrucciones, la cuadrícula y la imagen permanezcan visibles al mismo tiempo.



Remarks

Aquí hay una imagen. Pretenda que soy el robot con una Máquina de Realización Automática (ARM). Estas son las únicas instrucciones que entiendo.

Comenzando en la esquina superior izquierda, guíe mi ARM en voz alta con sus palabras.

**** Modelar: **** La clase puede darle instrucciones como estas a continuación. Cuando escuche una instrucción que pretende seguir, asegúrese de repetirla en voz alta para que los alumnos puedan realizar un seguimiento de lo que está haciendo.

Mueve un cuadrado a la derecha Rellenar cuadrado con color Mueve un cuadrado a la derecha Mover un cuadrado hacia abajo Rellenar cuadrado con color

Continúe con la actividad hasta que haya completado el cuadrado de muestra.

**** Registrar: **** Anote cada uno de los comandos para que los estudiantes puedan ver todos los pasos que se incluyeron en la imagen.

Mueve un cuadrado a la derecha Rellenar cuadrado con color Mueve un cuadrado a la derecha Mover un cuadrado hacia abajo Rellenar cuadrado con color

**** Decir: **** Acabas de darme una lista de pasos para terminar una tarea. En programación, llaman a eso un algoritmo. Los algoritmos son geniales, porque son fáciles de entender como programador. PERO, ¿qué sucede cuando queremos escribir el algoritmo para un dibujo como este?

**** Mostrar: **** Mostrar a los estudiantes una imagen más complicada, como la que se muestra a continuación.

[][2]

A continuación, comience a escribir algunas de las instrucciones que tomaría para replicar esa imagen. Con suerte, los estudiantes verán que escribir todo a largo plazo se convertirá rápidamente en una pesadilla.

Mueve un cuadrado a la derecha Rellenar cuadrado con color Mueve un cuadrado a la derecha Mueve un cuadrado a la derecha Rellenar cuadrado con color Mover un cuadrado hacia abajo Mover un cuadrado a la izquierda Rellenar cuadrado con color Mover un cuadrado a la izquierda Mover un cuadrado a la izquierda Rellenar cuadrado con color MÁS 12 MÁS INSTRUCCIONES!

**** Mostrar: **** Mostrar a los alumnos esta lista de símbolos.



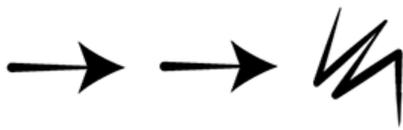
 <! - lugar donde le gustaría que aparezca el icono ->

**** Discusión: **** ¿Cómo podríamos usar estos símbolos para facilitar nuestras instrucciones?

Dibuje ideas que se relacionen con la transición de las instrucciones verbales a los símbolos. Una vez que los estudiantes hayan llegado a ese lugar, señale que este texto:

- “Mueve un cuadrado a la derecha, Mueve un cuadrado a la derecha, Rellena el cuadrado con color”

Correspondería ahora al programa:

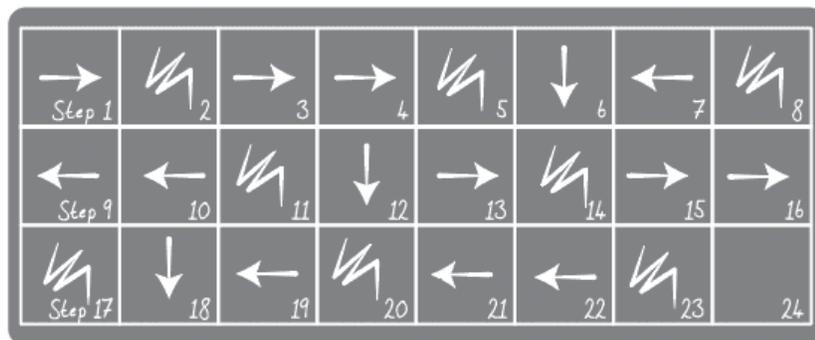
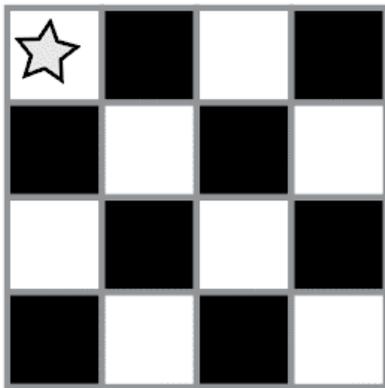


📍 <! - lugar donde le gustaría que aparezca el icono - >

**** Modelar: **** Ahora, haga que la clase le ayude a dibujar la imagen más grande usando sólo símbolos. No se preocupe por los pasos innecesarios por ahora. Si su programa final funciona para crear la imagen, considérela una victoria.

El aula puede estar repleta de sugerencias para este punto. Si la clase obtiene la esencia del ejercicio, este es un buen lugar para discutir formas alternativas de completar la misma cuadrícula. Si todavía hay confusión, guarde esa pieza para otro día y trabaje con otro ejemplo.

Vea una solución de muestra a continuación:



Discussion Goal

El objetivo de esta discusión es llegar a la idea de que lo/as estudiantes pueden usar símbolos para representar frases completas. Una vez que entiendan eso, comparta con ellos que hacer el cambio de los pasos de la lista en detalle para codificarlos se denomina "programación".

Teaching Tip

Tenga en cuenta que hemos escrito nuestro programa de izquierda a derecha como si leyera un libro en inglés. Algunos estudiantes prefieren este método. A otros les gusta comenzar cada línea de la cuadrícula en una nueva línea de papel. La forma en que escriben su programa no importa tanto si las otras personas en sus grupos pueden seguirlo.

El turno de los/as estudiantes

**** Agrupar: **** Divida a los estudiantes en parejas o grupos pequeños.

- Haga que cada pareja / grupo elija una imagen de la hoja de trabajo.
- Discuta el algoritmo para dibujar la imagen elegida con compañero (s).
- Convierta el algoritmo en un programa utilizando símbolos.
- Intercambie programas con otro par / grupo y dibujar la imagen del otro

Actividad de Cierre (15 mín)

Escribir en el Diario / Conversación Rápida

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 otra imagen que puedas codificar. ¿Puedes escribir el programa que va con este dibujo?
- ¿Qué otros tipos de robots podríamos programar si cambiáramos el significado de las flechas?

Evaluación Opcional (10 min)

Programación con Papel Cuadrulado - Evaluación

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

Aprendizaje Extendido

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

Mejor y Mejor

- Pida a la clase que intente inventar sus propias imágenes.
- ¿Pueden descifrar cómo programar las imágenes que crearon?

Desafío para la Clase

- Como profesor/a, dibuje una imagen en una cuadrícula de 5x5.
- ¿Puede la clase codificarla con usted?

Normas de aprendizaje

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

- ▶ **AP** - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 2: Introducción a los Desafíos Online

Secuencia | Depurar | Bucles | Angry Bird | Recolectora | Artista | Cosechadora

Descripción

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

Propósito

Reconocemos que cada aula tiene un espectro de comprensión de cada asignatura. Algunos/as estudiantes de su clase pueden ser magos/as de la computación, mientras otros/as no han tenido mucha experiencia. Con el propósito de crear un campo de juego (y aprendizaje) igualitario, hemos desarrollado esta “Etapa de Reforzamiento” para el Curso E. Puede ser usado como introducción o revisión de cómo usar Code.org y conceptos básicos de la ciencia de la computación. Esta etapa, en conjunto con las tres siguientes, cubren todos los pre-requisitos necesarios para empezar el Curso E.

Agenda

Actividad previa (10 min)

Introducción
Vocabulario

Actividades Puente - Programación (10 min)

Actividad desconectada usando bloques de papel
Vista previa de desafíos online como una clase

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Actividad de Cierre (10 min)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Juegue a través de **Curso D Desafíos Online - Sitio web** para encontrar cualquier problema potencial para su clase.
- ☐ Asegúrese de que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

- **Bloques Blockly Sin Conexión (Cursos 2-5)** - Manipuladores
- **Diario Think Spot** - Diario de reflexiones

Vocabulario

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

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

Guía docente

Actividad previa (10 min)

Introducción

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

Vocabulario

Esta lección tiene cuatro palabras nuevas e importantes de vocabulario:

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

Actividades Puente - Programación (10 min)

Esta actividad ayudará a llevar los conceptos desconectados de "Programación en papel cuadriculado" al mundo online en el que los/as estudiantes se están moviendo. Elija una de las siguientes opciones para hacer con su clase:

Actividad desconectada usando bloques de papel

Al igual que en "Programación en papel cuadriculado", haga que los/as estudiantes de su clase se pongan en parejas. Pase varios bloques fill 1 y move ___ de **Bloques Sin conexión para Laberinto/Granjera/Abeja - Cursos 2-5 - Manipuladores (PDF)** a cada pareja. Haga que cada pareja de estudiantes dibuje un diseño en una gráfica de cuatro por cuatro de **Programación con Papel Cuadriculado - Ficha**. Luego, haga que los/as estudiantes trabajen juntos para escribir el programa necesario para dibujar este diseño utilizando los bloques de bloques de papel. Los estudiantes deberán escribir up, down, right o left en el bloque move ___. ¡Asegúrese de que los/as estudiantes sepan que el programa va de arriba a abajo y que los bloques deben tocarse!

Vista previa de desafíos online como una clase

Tome un desafío de **Course D Online Puzzles 2018 - Sitio web**. Recomendamos el desafío 6 para esta actividad. Divida a los estudiantes en grupos de tres o cuatro. Pídales que "programen" Red, el pájaro enojado, para llegar al cerdo con las flechas de "Programación en papel cuadriculado".



La clase no necesitará usar la última flecha.

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

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Los maestros desempeñan un papel vital en la educación en Ciencias de la Computación y apoyan un entorno de clase colaborativo y dinámico. Durante las actividades online, el rol del maestro es principalmente de aliento y apoyo. Las lecciones online están destinadas a centrarse en el estudiante, por lo que los maestros deben evitar intervenir cuando los estudiantes se atascan. Algunas ideas sobre cómo hacer esto son:

- Utilice **Programación en Parejas - Video para el estudiante** siempre que sea posible durante la actividad.
- Anime a los/as estudiantes con preguntas / desafíos para comenzar preguntando a su compañero/a.
- Las preguntas sin respuesta se pueden escalar a un grupo cercano, que quizás ya sepa la solución.
- Recuerde a los/as estudiantes que utilicen el proceso de debugging antes de acercarse.
- Haga que los/as estudiantes describan el problema que están viendo. ¿Qué se supone que haga? ¿Qué hace? ¿Qué te dice eso?
- Recuerde a los/as estudiantes frustrados que la frustración es un paso en el camino hacia el aprendizaje y que la perseverancia dará sus frutos.
- Si un/a estudiante sigue atascado después de todo esto, haga preguntas importantes para que el/a estudiante detecte un error por su cuenta.

💡 Teacher Tip:

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

- No te sientes en la silla del compañero de clase.
- No uses el teclado del compañero de clase.
- No toques el ratón del compañero de clase.
- Asegúrate de que el compañero de clase pueda describirte la solución en voz alta antes de alejarte.

Actividad de Cierre (10 min)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Normas de aprendizaje

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

- ▶ **AP** - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 3: Eventos en Bounce

Evento | Rebotar

Descripción

En esta actividad online, los/as estudiantes aprenderán cuáles son los eventos y cómo los computadores los utilizan en programas como videojuegos. Los/as estudiantes trabajarán a través de desafíos y harán que el programa reaccione a eventos (como presionar los botones de flecha.) Al final del desafío, los/as estudiantes tendrán la oportunidad de personalizar su juego con sonidos y diferentes velocidades.

Propósito

Los eventos son muy comunes en los programas computacionales, especialmente en los videojuegos.

En esta lección, los/as estudiantes desarrollarán su comprensión de los eventos haciendo un juego basado en el deporte. Los/as estudiantes aprenderán a hacer que su paddle se mueva de acuerdo con las teclas de flecha, y a hacer ruidos cuando los objetos chocan. Al final, ellos tendrán que personalizar su juego para hacerlo más única!

Agenda

Actividad previa (10 min)

Actividad Principal (30 min)

Actividad de Cierre (10 min)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Identificar acciones que se correlacionan con eventos de entrada.
- Crear un juego interactivo mediante secuencia y controladores de eventos.
- Compartir un artefacto creativo con otros estudiantes.

Preparación

- ☐ Asegúrese de que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.
- ☐ Juegue a través de **Course D Online Puzzles 2018 - Sitio web** en la fase 3 para encontrar cualquier área de potenciales problemas para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

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

Vocabulario

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

Guía docente

Actividad previa (10 min)

Pida a los/as estudiantes que se sienten cerca de usted. ¡Ahora pídeles que se pongan de pie!

Diga a los/as estudiantes que lo que acaba de hacer fue declarar un evento y una acción. Cuando usted les dice que se sienten, eso es un evento. La acción que responde a este evento es que la clase se sienta. Esto es lo mismo cuando les pide que se pongan de pie. Los eventos y las acciones son muy fácilmente identificables en nuestras vidas.

Algunos de los eventos y de las acciones incluyen:

- Tener hambre y comer.
- Golpearte el dedo y gritar “¡Auch!”
- ¡Pasar la pelota de basquetbol por el aro y marcar un punto para tu equipo!

Pídale a la clase que piensen un par de eventos más. ¡Cuénteles que van a hacer un juego donde el programa tendrá acciones asociadas a los eventos que codifiquen!

Actividad Principal (30 min)

Al final de grupo de desafíos, los/as estudiantes tendrán la oportunidad de hacer que su juego sea único. Pida a los/as estudiantes que intenten formas nuevas de hacer su juego más desafiante. Por ejemplo, que intenten jugar con varias pelotas a la vez, o que cada vez que la pelota rebote en el muro, que se lancen más pelotas.

🔗 Teacher Tip

Recuerde a los/as estudiantes solamente compartir su trabajo con sus amigos y familiares. Para más información muestre a su clase **Detente y Piensa Online - Video**.

Esto es lo que hizo Justin Trudeau, el Primer Ministro de Canadá, cuando completó una Hora del Código en 2016[3].
[4]

Actividad de Cierre (10 min)

Escribir en el Diario

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

Sugerencias para el Diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué hiciste para que tu juego fuera súper genial?
- ¿Qué tipo de juego quieres codificar en el futuro?

Aprendizaje Extendido

Llévame al Juego de Pelota.

Lleve a los/as estudiantes al exterior para jugar un tipo de juego de pelota. Lleve registro de los eventos y de las acciones. Por ejemplo, no hacer “dribbling” en basquetbol resulta en una infracción y el otro equipo se queda con la pelota. En fútbol, patear la pelota fuera de los límites resulta en que el otro equipo patee la pelota hacia la cancha.

¡Llevar la pelota al arco resulta en un punto! Invente más eventos si sus estudiantes lo disfrutan. Pida a todos/as las estudiantes que griten “Yipii” cuando el capitán de un equipo marque un punto. ¡Haga que todo caigan al suelo y rueden si un /a estudiante hace dos goles seguidos!

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 4: Loops Anidados

Loops Anidados | Bucles | Abeja | Laberinto

Descripción

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

Propósito

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

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

Agenda

Actividad previa (10 min)

Introducción

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Actividad de Cierre (15 min)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** para encontrar áreas de potenciales problemas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

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

Vocabulario

- **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)

Course D Online Puzzles 2018 - Sitio web

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

Asegúrese de tener a mano lápices y papel para que los/as estudiantes anoten sus planes antes de codificar. Algunos desafíos tienen un límite en el número de ciertos bloques que pueden usar, así el papel puede ser útil si a los/as estudiantes les gusta anotar las respuestas completas antes de buscar patrones repetidos.

Actividad de Cierre (15 mín)

Escribir en el Diario

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

Sugerencias para el Diario:

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

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 5: Loops Anidados en Artista

Loops Anidados | Bucles | Artista

Descripción

Los/as estudiantes crearán diseños complicados usando Artista en el conjunto de los desafíos de hoy. Al continuar practicando loops anidados con nuevos objetivos, los/as estudiantes verán más usos de loops en general. Este conjunto de desafíos también ofrece mucho más potencial para la creatividad con la oportunidad para que los/as estudiantes creen su propio diseño al final de la etapa.

Propósito

En esta actividad online, los/as estudiantes crearán diseños en Artista que pueden compartir orgullosamente con sus seres queridos.

El propósito de esta actividad es utilizar loops anidados como una manera de inspirar a los/as estudiantes con mentes artísticas para ver la codificación como otro medio creativo. Este conjunto de desafíos se construyó para desarrollar habilidades de pensamiento crítico, un entendimiento de geometría elemental y creatividad -- todo dentro del alcance de los loops anidados!

Agenda

Actividad previa (10 min)

Introducción

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Actividad de Cierre (15 min)

Charla Exprés: ¿Qué hiciste hoy?

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Combinar formas simples en diseños complejos con bucles anidados.
- Contar el número de veces que una acción debería repetirse y representarla como un bucle.
- Dividir las tareas complejas en secciones repetibles más pequeñas.

Preparación

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** para encontrar cualquier problema potencial para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese de que cada estudiante tenga un **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Reparta o imprima copias de **Giros & Ángulos - Guía para el estudiante** para los/as estudiantes como referencia mientras trabajan en los desafíos online.

Enlaces

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

Para los docentes

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

Para los estudiantes

- **Giros & Ángulos** - Guía para el estudiante
- **Giros & Ángulos** - Video para el estudiante
- **Diario Think Spot** - Diario de reflexiones

Vocabulario

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

Guía docente

Actividad previa (10 min)

Introducción

Revise el uso de loops anidados en laberinto.

Pregunte a los/as estudiantes cómo se sintieron acerca de los loops anidados.

- ¿Qué les gustó y qué no les gustó de ellos?
- ¿Cuáles son algunas ventajas de usar loops anidados?

Cuéntele a los/as estudiantes que usarán loops anidados de nuevo, pero esta vez en Artista. ¡Harán proyectos increíbles hoy!

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Los/as estudiantes podrían beneficiarse de resolver un desafío como clase. Si usted cree que su clase podría beneficiarse con eso, le recomendamos el desafío 2 de la lección 5. Recomendamos altamente el Video para el/la Estudiante - Programación en Parejas en esta lección. Este puede no ser un tema fácil para la mayoría de sus estudiantes. Trabajar con un/a compañero/a y discutir sobre las posibles soluciones a los desafío podría aliviar a los/as estudiantes. Asegúrese de tener a mano lápices y papel para que los/as estudiantes anoten sus planes antes de codificar. Algunos desafíos tienen un límite en el número de ciertos bloques que pueden usar, así el papel puede ser útil si a los/as estudiantes les gusta anotar las respuestas completas antes de buscar patrones repetidos.

💡 Teacher Tip

Los/as estudiantes podrán compartir su propio trabajo al final de la lección. Estas obras de arte pueden ser compartidas de forma virtual o impresa. Recomendamos imprimir el trabajo de la clase y mostrarla para que los seres queridos de los/as estudiantes la vean.

Actividad de Cierre (15 mín)

Charla Exprés: ¿Qué hiciste hoy?

Junte a su clase y asigne tiempo a sus estudiantes para mostrar sus dibujos en Artist! Asegúrese de que cada uno se sienta incluido chequeando que todos los estudiantes hayan terminado sus dibujos en Artist antes de las presentaciones. Discutan acerca de cómo hicieron los dibujos y que había en el loop anidado de los estudiantes.

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

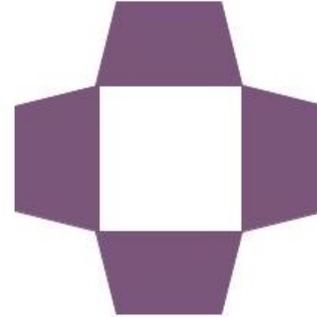
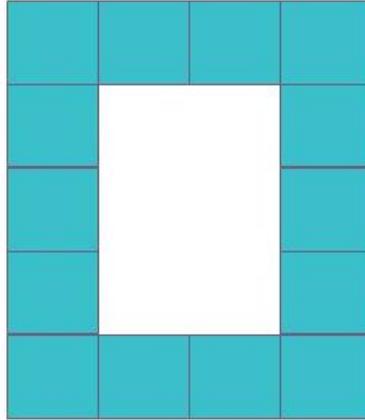
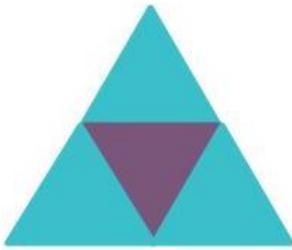
- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- Dibuja una imagen en la que utilizaste loops anidados para hacerla.
- ¿De qué manera los loops anidados te ayudan a codificar imágenes complejas?

Aprendizaje Extendido

Juntos Dibujamos.

Ponga a los/as estudiantes en parejas con dos trozos de papel. Cada integrante de la pareja debe dibujar de forma individual una forma o un patrón simple. Una vez que lo han hecho, pida a los/as integrantes de la pareja que intercambien los papeles. Ahora cada uno/a debe repetir ese patrón cuántas veces quiera. Por ejemplo, si un miembro de la pareja dibuja un cuadrado, ¡el otro miembro puede crear un rectángulo hecho de cuadrados! Si un miembro dibuja el patrón de una escalera, ¡el/la otro/a estudiante puede llenar la página con escaleras! Cada pareja tendrá un set de dibujos únicos. Si hay tiempo, pida a los/as estudiantes discutir cómo podrían codificar sus dibujos.

Aquí hay algunos ejemplos:



Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 6: Loops Anidados con Frozen

Bucle | Bucle anidado | Frozen

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)

Curso E Desafíos Online - Sitio web

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

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** correspondiente a esta lección para encontrar cualquier potencial área problemática para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese de cada estudiante tenga un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

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)

Curso E Desafíos Online - Sitio web

Este conjunto de desafíos se configura como una progresión. Esto significa que cada desafío construye una base para el siguiente desafío. Los/as estudiantes disfrutarán haciendo diseños cada vez más interesantes al hacer pequeños y simples cambios en el código que ya han escrito.

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.

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 7: Programación por Relevos

Sin conexión | Programación por Relevos | Algoritmos

Descripción

Esta actividad comenzará con una breve revisión de "Programación En Papel Cuadrulado", luego se moverá rápidamente a una carrera contra el reloj, a medida que los estudiantes se formen en equipos y trabajen juntos para escribir un programa de una instrucción a la vez.

Propósito

El trabajo en equipo es muy importante en las Ciencias de la Computación. Los equipos escriben y debug código entre ellos, en lugar de trabajar individualmente. En esta lección, los/as estudiantes aprenderán a trabajar juntos siendo lo más eficientes posible.

Esta actividad también proporciona un sentido de urgencia que enseñará a los/as estudiantes a equilibrar su tiempo cuidadosamente y a evitar errores para no quedar tan atrás.

Agenda

Actividad previa (15 mín)

Actividad Principal (20 min)

Programación en Relevos

Actividad de Cierre (15 mín)

Escribir en el Diario

Evaluación (10 min)

resource csf-relay-programming-assessment not found

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Definir ideas mediante código y símbolos.
- Verificar el trabajo realizado por los compañeros de equipo.

Preparación

- Vea [teacher-video-relay-programming-2][0].
- Seleccione un espacio abierto para esta actividad, como el gimnasio o el patio.
- Imprima la guía [csf-relay-programming-activity-packet][1] para cada grupo.
- Proporcione suficientes lápices y papel a cada grupo.
- Imprima una [csf-relay-programming-assessment][2] para cada estudiante.
- Asegúrese que cada estudiante tenga un [think-spot-journal][3].

Enlaces

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

Para los docentes

- **Programación por Relevos** - Guía de respuestas en fichas [Hacer una copia](#)

Para los estudiantes

- **Programación por Relevos** - Video sin conexión (**descargar**)
- **Paquete de Actividad de la Programación con Relevos** - Paquete de actividades [Hacer una copia](#)
- **Programación por Relevos** - Ficha [Hacer una copia](#)
- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#)

Vocabulario

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

Guía docente

Actividad previa (15 mín)

Recuerde que en la 'Programación en Papel Cuadrulado' guiamos a la Máquina de Realización Automática (ARM) de nuestros/as compañeros/as de equipo, usando flechas. Tómese un momento para revisar una imagen de la 'Programación en Papel Cuadrulado' como recordatorio. Puede ser una que ya hayan cubierto u otra nueva. Vamos a hacer lo mismo hoy, pero en lugar de controlarnos mutuamente, vamos a trabajar juntos para crear un programa y un símbolo a la vez.

Actividad Principal (20 min)

Programación en Relevos

Con la Programación en Papel Cuadrulado en mente, ¡es hora de dividirse en equipos y prepararse para ejecutar la actividad como un relevo!

**** Configurar: **** Prepare el **Paquete de Actividad de la Programación con Relevos - Paquete de actividades** imprimiendo una copia para cada equipo de 4-5 estudiantes. Cortar o doblar cada página a lo largo de la línea de puntos central.

Repase las reglas del juego con su clase:

- Divida a los alumnos en grupos de 3-5.
- Haga que cada grupo haga cola en el estilo de relevo.
- Coloque una imagen idéntica en el otro lado de la sala / gimnasio / campo de cada equipo.
- Haga que el primer alumno de la línea se dirija a la imagen, la revise y escriba el primer símbolo en el programa para reproducir esa imagen.
- El/a primer/a estudiante luego corre hacia atrás y etiqueta a la siguiente persona en la fila, luego va al final de la fila.
- La siguiente persona en la línea se desplaza a la imagen, revisa la imagen, revisa el programa que ya se ha escrito, luego elimina el programa eliminando un símbolo incorrecto o agrega uno nuevo. Ese estudiante luego se apresura para etiquetar a la siguiente persona, y el proceso continúa hasta que un grupo termina su programa.

🔍 Clarifications

Aquí hay algunas aclaraciones que deben compartirse de vez en cuando:

- Sólo una persona de cada grupo puede estar en la imagen a la vez.
- Está bien discutir los algoritmos con el resto del grupo en línea, incluso hasta el punto de planificar quién escribirá qué cuando llegue a la imagen.
- Cuando un/a estudiante realiza un debug a un programa al tachar una instrucción incorrecta (o una agrupación de instrucciones incorrectas), esto cuenta como su turno completo. El siguiente jugador deberá averiguar cómo corregir el elemento eliminado.

¡El primer grupo que termine con un programa que coincida con la imagen es el ganador! Juega a través de esto varias veces, con imágenes de dificultad creciente.

Repase el juego tantas veces como sea posible antes de que se acabe el tiempo o sus estudiantes comiencen a sentirse agotados.

**** Transición: **** Una vez que el juego haya terminado, encierra en un círculo a todos para compartir las lecciones aprendidas.

**** Discutir: **** ¿Qué aprendimos hoy?

- ¿Qué pasaría si a cada persona en un equipo se le permitiera hacer cinco flechas a la vez? - ¿Qué tan importante sería hacer debug a nuestro propio trabajo y el trabajo del programador que tenemos ante nosotros?
 - ¿Qué tal con 10 flechas? - ¿10,000? ¿Sería más o menos importante?
- ¿Crees que un programa es mejor o peor cuando más de una persona ha trabajado en él?
- ¿Crees que las personas cometen más o menos errores cuando tienen prisa?

- Si encuentra un error, ¿tiene que desechar todo el programa y volver a empezar?

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?
- ¿Cómo el trabajo en equipo jugó un papel en el éxito de escribir el programa de hoy?
- ¿Cómo usaste tus habilidades para hacer debugging en la lección de hoy?

Evaluación (10 min)

resource csf-relay-programming-assessment not found

Distribuya esta evaluación y pida que la contesten en forma individual. Al final, puede tomarse un momento para revisarla y discutir las respuestas.

Aprendizaje Extendido

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

Pasar el Papel

- Si no tiene el tiempo o el espacio para un relevo, puede pedirle a los/as estudiantes pasar un papel a su grupo estando sentados alrededor de una mesa. Cada uno puede escribir una flecha antes de pasar el papel al siguiente.

Llévalo, Pásalo

- Como profesor/a, dibuje una imagen con tantos cuadrados llenos, como estudiantes en un grupo.
- Pida a los/as estudiantes escribir tantas flechas en el programa como sea necesario, para llegar a un cuadrado rellenado (incluyendo el llenado de ese cuadrado) antes de pasarlo a la siguiente persona.

Haciendo Debugging Juntos

Dibuje una imagen en la pizarra. Pida a cada estudiante crear un programa para la imagen. Pida a los/as estudiantes intercambiar con su compañero de banco y hacer debugging al código del otro.

- Encierre en un círculo el primer paso incorrecto, luego devuelva la hoja de papel.
- De a los/as estudiantes otra oportunidad de revisar y hacer debugging a su propio trabajo.
- Pida un/a voluntario/a para compartir su programa.

Pregunte a la clase:

- ¿Cuántos tienen el mismo programa?
- ¿Alguien tiene algo diferente?

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 8: Debugging en Recolectora

Depurar | Bug o error | Recolectora | Laurel

Descripción

En esta actividad online, los/as estudiantes practicarán debugging en el entorno "recolector". Los/as estudiantes podrán practicar la lectura y edición de código para arreglar desafíos con algoritmos simples, loops y loops anidados.

Propósito

El propósito de esta lección es enseñar a los/as estudiantes que el fracaso es normal al aprender una nueva habilidad. Los/as estudiantes recibirán programas pre-escritos que NO funcionen. Se les pedirá que corrijan estos programas. Este proceso, llamado "debugging", enseña a los/as estudiantes problemas esenciales para resolver y habilidades de pensamiento crítico. Estas habilidades se transfieren a medida que los estudiantes pasan a proyectos de programación cada vez más difíciles.

Agenda

Actividad previa (15 mín)

Introducción

Actividades Puente - Debugging (15 mín)

Actividad Sin Conexión con Bloques de Papel
Previsualización del Desafío En Línea como Clase

Actividad Principal (30 mín)

Course D Online Puzzles 2018 - Sitio web

Actividad de Cierre (15 mín)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Leer y comprender el código dado.
- Identificar un error y los problemas que causa en un programa.
- Describir e implementar un plan para depurar un programa.

Preparación

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

Enlaces

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

Para los docentes

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

Para los estudiantes

- **Paquete de Actividad de la Programación con Relevos** - Paquete de actividades [Hacer una copia](#)
- **Bloques Blockly Sin Conexión (Cursos 2-5)** - Manipuladores
- **Diario Think Spot** - Diario de reflexiones [Hacer una copia](#)

Vocabulario

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

Guía docente

Actividad previa (15 mín)

Introducción

Una de las partes más importantes de aprender a programar es aprender a debug/depurar. Pregunte a su clase si alguna vez que han tenido que aprender algo nuevo han fracasado.

Por ejemplo:

- Aprender a andar en bicicleta y caerse
- Aprender a cocinar y quemar la comida
- Aprender un deporte y no ganar el juego

Enfrentar el fracaso es muy habitual cuando se aprenden nuevas cosas. Discuta con sus estudiantes acerca de algún fracaso y como lo superó.

En programación, los programadores generalmente se encuentran con "bugs/errores" en sus códigos.

- ***Bug/Error***: Parte de un programa que no se ejecuta correctamente.

Un bug puede echar a perder un programa, por eso es importante a aprender a "debug"/"depurar" su código.

- **Debug/Depurar**: Encontrar y arreglar problemas en su algoritmo o programa.

Continúe la conversación si considera que su clase necesita más que una introducción, pero deje tiempo para alguna de las actividades puente.

Actividades Puente - Debugging (15 mín)

Esta actividad ayudará a traer los conceptos sin conexión del "Debugging Sin Conexión: Programación de Relevos" al mundo online al que los alumnos están entrando. Escoja uno de los siguientes para hacer con su clase:

Actividad Sin Conexión con Bloques de Papel

Separe a su curso en equipos de 3 - 5 y llévelos a un espacio grande. Este espacio puede ser al aire libre o en el gimnasio. Ponga a los equipos en fila como en la "Programación de Relevos". Elija un diseño semi difícil de **Paquete de Actividad de la Programación con Relevos - Paquete de actividades**. Muestre este diseño al final de una gran distancia entre cada equipo. Junto con mostrar la imagen, proporcione a cada equipo suficientes bloques de papel de **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores**. Cada equipo necesitará suficientes bloques 'llenar1' y 'mover **_'. Los bloques 'mover **_' se pueden completar ya sea de antemano o durante el juego. De cualquier manera, asegúrese que estos bloques están definidos adecuadamente cuando se esté jugando.

Una vez que los equipos están en fila, muestre o lea las siguientes reglas:

- El primer alumno en la fila debe correr hacia la imagen, revisarla, y poner el primer bloque de código en el programa para reproducir la imagen.
- El primer alumno corre de vuelta y toca al siguiente en la fila, luego se va al final de la fila.
- La siguiente persona en la fila corre hacia la imagen, la revisa, revisa el programa que ya ha sido escrito, luego debug el programa quitando el bloque de código incorrecto o agrega un bloque nuevo. Ese alumno corre de vuelta para tocar a la siguiente persona, y el proceso continúa hasta que un grupo termine su programa.

Asegúrese que los alumnos sólo estén usando los bloques 'llenar 1' o ' mover ___` y que estén poniendo sólo uno por turno. El primer grupo que escriba correctamente el código para su imagen gana.

Previsualización del Desafío En Línea como Clase

Agrupe a los alumnos en equipos de 3. Escoja un desafío de **Course D Online Puzzles 2018 - Sitio web** asociado con esta lección. Recomendamos el tercer desafío. Haga que cada equipo se sienta frente a un computador mostrando el desafío. Cada equipo usa sólo un computador y sólo un alumno puede estar mirando a la pantalla. Muestre o lea las siguientes reglas:

- Sólo un alumno de cada equipo puede mirar la pantalla.
- Esta persona sólo puede borrar o agregar un bloque a la vez. Una vez que esa persona ha agregado o removido un bloque, puede tocar el hombro de la persona siguiente.
- La siguiente persona puede proceder a jugar su turno.
- No se pueden saltar ni repetir los turnos, todos deben jugar una cantidad igual.

¡El primer equipo en finalizar el desafío_correctamente_gana!

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Podría ser útil para los estudiantes sentarse con sus equipos desde las actividades puente. Cada estudiante debe trabajar en estos desafíos individualmente o en pares, pero tener un grupo muy unido para hacer y responder preguntas puede ayudar a desarrollar confianza y comprensión acerca del tema.

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 bug? ¿Cómo sabes que hay un bug en tu programa?
- ¿Qué significa debug un código? ¿Cómo se hace debug a un programa?

Normas de aprendizaje

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

- ▶ **AP** - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 9: Loops Mientras en la Granjera

Bucles mientras | Bucles | Granjera

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

Aprendizaje Extendido

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

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** para encontrar áreas de potenciales problemas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

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

Vocabulario

- **Condición** - una sentencia que un programa verifica si es verdadera o falsa.

Si es verdadera, se realiza una acción. De lo contrario, se ignora.

- **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)

Los “loops mientras” no siempre son un concepto difícil de entender para los/as estudiantes, pero si cree que su clase podría tener problemas con estos desafíos, recomendamos el video del/la estudiante - Programación en Parejas. Esto le permitirá a los/as estudiantes compartir las ideas de todos antes de implementar el código. La programación en parejas sirve para incrementar la confianza y la comprensión de los temas como los “loops mientras”.

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?
- ¿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?

Aprendizaje Extendido

Simón manda “Mientras”

Salga al patio y ponga a los/as estudiantes en fila frente de usted. Asegúrese que cada estudiante pueda verlo/a. Declare un par de “loops mientras”, como: • Mientras mi mano derecha esté arriba, pueden caminar hacia mí. • Mientras cubro mis ojos pueden saltar hacia mí. • Mientras mi cabeza este girada hacia la derecha, tienen que caminar hacia atrás.

El/la primer/a estudiante que llegue a usted, gana. Si hay tiempo, deje que otros/as estudiantes hagan de “Simón” frente a la clase.

Normas de aprendizaje

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

► AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 10: Si/si no: Condicionales con Cartas

Condicionales | Sin conexión

Descripción

Esta lección demuestra cómo se pueden usar los condicionales para moldear un programa a alguna información específica. No siempre tenemos toda la información que necesitamos cuando escribimos un programa. A veces querrá hacer algo diferente en una situación y en otra, incluso si no sabe cuál situación será verdadera cuando ejecute el código. Ahí es donde entran los condicionales. Los condicionales permiten al computador tomar una decisión, basada en la información que es verdadera cada vez que su código se está ejecutando.

Propósito

Una de las mejores formas de enseñar condicionales es que los/as estudiantes ya comprenden este concepto en su vida diaria. Esta lección mezcla la ciencia de la computación con el mundo real, mediante la construcción de su habilidad de decir si una condición es verdadera o falsa. Los/as estudiantes aprenderán a usar las afirmaciones “Si” para declarar cuando cierto comando debe ser ejecutado, así como afirmaciones “Si / Si no”, para declarar cuando se debe ejecutar un comando y que debe ser ejecutado de otra manera. Puede que los/as estudiantes no reconozcan la palabra condicionales, pero la mayoría de los/as estudiantes comprenderán la idea de usar “Si” para asegurarse que algunas acciones sólo ocurran cuando se supone que deben hacerlo.

Agenda

Actividad Previa (20 min)

Vocabulario
Introducción

Actividad Principal (20 min)

Programa Condicionales con Cartas de Muestra - Guía de preparación para el docente

Actividad de Cierre (15 min)

Conversación Rápida: ¿Qué aprendimos?
Escribir en el Diario

Evaluación (5 min)

Si/si no: Condicionales con Cartas - Evaluación

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.
- Recorrer un programa y predecir el resultado dado un conjunto de información.

Preparación

- Vea **Si/si no: Condicionales con Cartas - Video para el docente**.
- Vea **Si/si no: Condicionales con Cartas - Video de lección en acción**.
- Junte mazos de cartas o algo similar.
- Una **Programa Condicionales con Cartas de Muestra - Guía de preparación para el docente** para mostrar a la clase.
- Imprima una copia **Si/si no: Condicionales con Cartas - Evaluación** para cada estudiante.
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

- **Si/si no: Condicionales con Cartas - Video de lección en acción**
- **Programa Condicionales con Cartas de Muestra - Guía de preparación para el docente**
- **Si/si no: Condicionales con Cartas - Video de la evaluación**
- **Condicionales con Cartas - Guía de respuestas de la evaluación**

Para los estudiantes

- **Si/si no: Condicionales con Cartas** - Video sin conexión (**descargar**)
- **Si/si no: Condicionales con Cartas** - Evaluación ▾
- **Diario Think Spot** - Diario de reflexiones ▾

Vocabulario

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

Guía docente

Actividad Previa (20 min)

Vocabulario

Esta lección tiene una nueva palabra:

Condicionales – Díganlo conmigo: Con-di-cio-na-les

Afirmaciones que sólo se ejecutan bajo ciertas condiciones.

Introducción

- Podemos comenzar esta lección de inmediato.
- Diga a la clase que si pueden estar completamente en silencio por treinta segundos, usted hará una de las siguientes acciones:
 - Cantar una canción de una ópera.
 - Darles cinco minutos más de recreo.
 - Hacer la posición invertida.
 - Comience a contar de inmediato.
 - Si los/as estudiantes tienen éxito, señale que lo hicieron muy bien, así que obtendrán su recompensa.
 - De lo contrario, señale que no completaron el tiempo de treinta segundos de silencio, así que no obtendrán su recompensa.
- Pregunte a la clase “¿Cuál fue la condición para la recompensa?”
- La condición fue Si estaban en silencio por 30 segundos.
- Si lo estuvieron, la condición sería verdadera, y obtendrían la recompensa.
- Si no lo estuvieron, la condición sería falsa, y la recompensa no aplicaría.
- ¿Podemos crear otro condicional?
- Si pueden adivinar mi edad, la clase les puede aplaudir.
- Si sé una respuesta, puedo levantar mi mano.
- ¿Qué ejemplos puedes crear?
- A veces, queremos tener una condición extra, en caso que la afirmación “Si” no fuera verdadera.
- Esta condición extra se llama una afirmación “Si no”.
- Cuando la condición “Si” no se cumple, podemos mirar al “Si no” para saber qué hacer.
- Ejemplo: Si dibujo un rey de este mazo de cartas, todos aplauden. Si no, todos dicen “Ooooooh”.
- Intentémoslo. (Dibuje una carta y vea si su clase reacciona apropiadamente)
- Pida a la clase analizar lo que sucedió.
- ¿Qué fue el “Si”?
- ¿Qué fue el “Si no”?
- ¿Qué condición se cumplió?
- Aunque no lo crean, tenemos incluso una opción más.
- ¿Qué pasaría “Si” ustedes aplauden cuando saco un 7 del mazo, “Si no” dicen “¡YAY!”, cuando saco un número menor que siete, o “Si” ninguna de las anteriores se cumple ustedes dicen “¡oooooh!”?
- Por esto tenemos el término “Si”, “Si / Si no” y “Si no”.
- “Si” es la primera condición.
- “Si no / Si” se usa sólo cuando “Si” no es verdad.
- “Si” se usa sólo cuando nada de lo anterior es verdad.

Ahora, hagamos un juego.

Actividad Principal (20 min)

Programa Condicionales con Cartas de Muestra - Guía de preparación para el docente

** Instrucciones**

- Cree unos pocos programas con su clase, que dependan de cosas como la pinta, el color o el valor de las cartas, para dar o quitar puntos. Puede escribir el programa como un algoritmo, pseudocódigo o un código real. Aquí hay un ejemplo de algoritmo:

Aquí hay una muestra de algoritmo:

Si (Carta Roja) Premia a tu equipo con 1 punto

Si no Premia al equipo contrario con 1 punto

Aquí hay una muestra del mismo programa en pseudocódigo:

```
Si (carta.color == Rojo){ puntos.tuyo = puntos.tuyo + 1; }
```

```
Si no { puntos.otro = puntos.otro + 1; }
```

- Decida cómo quiere dividir a su clase en equipos.
- Cada equipo debería tener una pila de cartas (al menos tantas cartas como miembros del equipo) a mano.
- Ponga uno de sus “programas” en la pizarra para que todos los vean.
- Haga que los equipos tomen turnos para tomar cartas y seguir el programa para ver cuántos puntos obtienen en cada vuelta.
- Juegue varias veces con diferentes programas para ayudar a los/as estudiantes a comprender realmente los condicionales.

Una vez que la clase ha tenido algo de práctica, puede animarlos/as a anidar los condicionales dentro de otro. Asegúrese que comprendan que si la carta es roja, SU equipo es recompensado con un punto, y luego *nada más ocurre* ya que se cumplió la condición.

```
Si (Carta es Roja){ Premia a tu equipo con 1 punto
```

```
Si no Si (Carta es mayor que 9) Premia al otro equipo con 1 punto Si no Premia a tu equipo con el mismo número de puntos de la carta
```

Aquí está el mismo programa en pseudocódigo:

```
Si (carta.color == ROJO ){ puntos.tuyo = puntos.tuyo + 1; } Si no { Si (carta.valor > 9){ puntos.otros = puntos.otros + 1; } Si no { puntos.tuyo = puntos.tuyo + carta.valor; }
```

Actividad de Cierre (15 mín)

Conversación Rápida: ¿Qué aprendimos?

- Si iban a codificar esto en Blockly, ¿qué necesitarían agregar a sus condicionales para que el código se ejecute más de una vez? (loop)
- ¿Qué otras cosas hacen durante el día, bajo ciertas condiciones?
- Si se supone que hagan algo cuando el valor de la carta es más que 5, y sacas un 5, ¿cumples esa condición?
- Note que las condiciones pueden ser “Verdaderas” o “Falsas”. No hay evaluación de una condición que evalúe al “Plátano”.
- Cuando necesiten cumplir varias combinaciones de condiciones, podemos usar algo llamado “condicionales anidados”.
 - ¿Qué crees que significa eso?
 - ¿Puedes dar un ejemplo de dónde vimos eso durante el juego?
- ¿Qué parte del juego te gustó más?

💡 Lesson Tip

Las preguntas de la Charla Exprés tienen como objetivo incentivar el pensamiento a gran escala pensando en cómo la lección se relaciona con el gran mundo de los/as estudiantes y su incluso, más grande futuro. Utilice el conocimiento de su aula para decidir si quiere discutir las como clase, en grupos pequeños o compañeros/as de banco.

Escribir en el Diario

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

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué es un condicional? ¿Cómo usaste un condicional hoy?
- ¿Cuáles son algunos condicionales que usaste hoy? ¿Puedes crear algunos más que podrías usar con un mazo de cartas?

Evaluación (5 min)

Si/si no: Condicionales con Cartas - Evaluación

Distribuya la hoja de la evaluación y permita que los/as estudiantes completen la actividad en forma independiente luego que las instrucciones queden claras. Esto debería serles familiar debido a las actividades en lecciones previas. Aquí hay un **Si/si no: Condicionales con Cartas - Video de la evaluación** para ver un video como guía.

Aprendizaje Extendido

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

Etiqueta Verdadero/Falso

- Ponga a los/as estudiantes en fila para jugar Luz Roja/Luz Verde.
- Seleccione a una persona para pararse delante como el que dirige.
- El que dirige escoge una condición y le pregunta a todos quien cumple la condición para que dé un paso al frente.
- Si tienen un cinturón rojo, un paso al frente.
- Si estás usando sandalias, un paso al frente.
- Intente cambiar diciendo cosas como “si te gustan las manzanas, un paso al frente”.

Anidar

- Separe a los/as estudiantes en parejas o pequeños grupos.
- Pídales que escriban afirmaciones “Si” en tiras de papel para jugar cartas, como:
 - la pinta es trébol.
 - el color es rojo.
- Pida a los/as estudiantes crear tiras de papel para los resultados.
- Agregar un punto.
- Quitar un punto.
- Una vez que eso está hecho, pida a los/as estudiantes que escojan tres de cada tipo de tira y tres cartas, poniendo atención al orden seleccionado.
- Usando estos papeles, pida a los/as estudiantes escribir tres programas diferentes usando sólo las tiras de papel seleccionadas, en cualquier orden.
- Anime a los/as estudiantes a poner algunas afirmaciones “Si” dentro de otras afirmaciones “Si”.
- Ahora, los/as estudiantes deberían ejecutar los tres programas usando las cartas que escogieron, en el mismo orden para cada programa.
- ¿Entregó la misma respuesta, alguno de los dos programas?
- ¿Alguno respondió algo diferente?

Normas de aprendizaje



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

Español (Latinoamérica) ▼

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

Lección 11: Condicionales en Abeja

Condicional | Abeja | Laberinto

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

Actividad previa (10 min)

Actividad Puente - Condicionales (15 min)

Actividad Sin Conexión Usando Bloques de Papel
Visión Previa de los Rompecabezas En Línea

Actividad Principal (30 min)

Actividad de Cierre (15 min)

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

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** para encontrar áreas de potenciales problemas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

- **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores**
- **Diario Think Spot - Diario de reflexiones**

Vocabulario

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

Guía docente

Actividad previa (10 min)

Revise la actividad "Condicionales con Cartas" con sus estudiantes.

- ¿Qué es una afirmación condicional?
- ¿Cuándo es útil un condicional?
- ¿Cuáles son algunos de los condicionales que usaste en la última actividad?

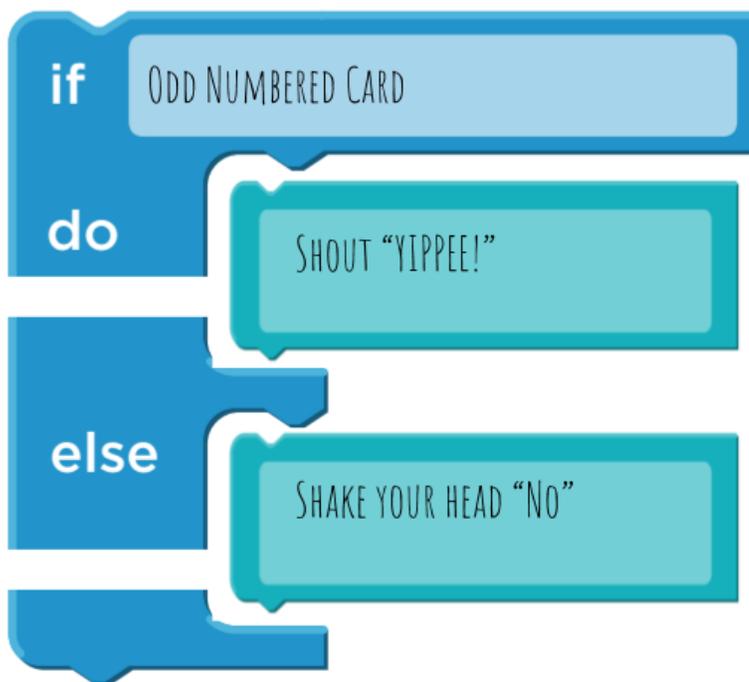
Ahora vamos a usar condicionales con la abeja de Code.org para ayudarla a lidiar con algunas nubes misteriosas. No sabemos si sus flores tienen néctar o no, así que necesitaremos usar condicionales para asegurarnos de recoger el néctar si está ahí, pero que no intentaremos recoger néctar en una flor que no tiene.

Actividad Puente - Condicionales (15 mín)

Esta actividad ayudará a traer los conceptos sin conexión de los "Condicionales con Cartas" al mundo en línea al que los alumnos están entrando. Escoja uno de los siguientes para hacer con su clase:

Actividad Sin Conexión Usando Bloques de Papel

Imprima y corte 2 -3 bloques 'si/más' y de acción en blanco de **Bloques Blockly Sin Conexión (Cursos 2-5) - Manipuladores** y saque un mazo de cartas. Pida a la clase que propongan un par de condicionales para usar con el mazo de cartas como lo hicieron en "Condicionales con Cartas". Cuando los condicionales hayan sido elegidos como clase, complete la parte en blanco del bloque 'if' con distintos valores de las cartas que los niños propusieron. Algunos ejemplos incluyen "Rey de Corazones", "Números Pares" o "Diamantes". Complete los bloques de acción con las acciones que los alumnos propusieron. Asegúrese que los alumnos conozcan los bloques de acción que se necesitan directamente bajo el bloque 'si' o 'más'. Más abajo un ejemplo.



Ahora mezcle el mazo de cartas y juegue nuevamente "Condicionales con Cartas". Voltee el mazo carta por carta, reaccionando a la scartas si se ha hecho un condicional para ella.

Visión Previa de los Rompecabezas En Línea

Saque un rompecabezas del Curso D, recomendamos el rompecabezas 9.

- Pregunte al curso lo que la abeja debería hacer cuando llegue a la nube.
 - La abeja debería usar un condicional para revisar si hay una flor o un panal.
- Use el bloque 'si en la flor/más'. Pregunte a la clase que debería hacer la abeja si hay una flor. si no hay una flor, habrá un panal. ¿Qué debería hacer la abeja entonces?
 - La abeja debería 'recoger néctar' si hay una flor y 'hacer miel' si hay un panal.

Complete el resto del código y presione 'ejecutar'. discuta con la clase por qué esto funcionó.

Actividad Principal (30 min)

Estos desafíos podrían generar algunas preguntas, así que ponga a los/as estudiantes a trabajar en parejas o implemente la regla "Pregúntale a tres antes que a mí" (pídales que le pregunten a tres compañeros/as antes de acudir al/a profesor/a). Esto producirá discusiones que posibilitarán el desarrollo de la comprensión de los/as estudiantes.

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 tú 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?

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 12: Condicionales & Loops en Laberinto

Condicional | Bucle | Laberinto | Angry Bird | Zombi

Descripción

En esta lección, los/as estudiantes unirán dos conceptos clave: loops y condicionales. Este conjunto de desafíos disminuye las brechas en la comprensión de qué ocurre al trabajar en desafíos que usan múltiples tipos de bloques. Al unir dos ideas, los/as estudiantes crearán un código más complejo que muestra impresionante creatividad y pensamiento crítico!

Propósito

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

Agenda

Actividad previa (10 min)

Introducción

Actividad Principal (30 min)

Actividad de Cierre (15 min)

Escribir en el Diario

Aprendizaje Extendido

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

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

Enlaces

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

Para los docentes

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

Para los estudiantes

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

Vocabulario

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

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

Introducción

Los/as estudiantes unirán los nuevos “Loops mientras” y las afirmaciones “si/si no”, así que esta instrucción repasará lo que hacen estos bloques en un sentido general.

Loops Mientras

Pregunte a la clase si recuerdan lo que significa “mientras” en codificación. Repase la definición de un “loop mientras”.

- *Loops Mientras*: Un loop que continúa repitiendo mientras una condición sea verdadera.

Además de los loops mientras, los/as estudiantes practicarán con los “loops hasta” en este grupo de desafíos. Explíqueles que los “loops mientras” continúan repitiendo el código mientras una condición sea verdadera, mientras los loops hasta continúan hasta que una condición es verdad.

Por ejemplo, con un loop mientras, el zombi continuará caminando por el camino mientras haya un camino por delante. Con un loop hasta, el zombi continuará caminando hasta que alcance la flor al final del camino. Eso se mostrará más en detalle dentro de los desafíos.

Actividad Principal (30 min)

Unir los conceptos no es fácil, pero este grupo de lecciones está hecho para unir las afirmaciones “si/si no” y los “loops mientras” para que los/as estudiantes vean el sin número de posibilidades de codificar cuando se usan en juntos. Si los/as estudiantes tienen problemas para comprender las similitudes o las diferencias entre las afirmaciones “si/si no”, los “loops mientras” o los “loops hasta”, pídale que vuelvan a practicar en los desafíos previos que solo usan uno de los tres.

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?
- ¿Cuál es la diferencia entre un loop "hasta" y un loop "mientras"?
- ¿Qué te gusta usar más: afirmaciones "si/si no", loops "hasta" o loops "mientras"? ¿Por qué?

Aprendizaje Extendido

Simón manda Hasta

Salga al patio y pida a los/as estudiantes que se paren en línea frente a usted. Asegúrese que todos/as puedan verlo/a. Declare una pareja de “loops hasta” como:

- Hasta que levante mi mano derecha, pueden caminar hacia mí. • Hasta que diga “tomate” pueden caminar retrocediendo hacia mí.
- Hasta que gire mi cabeza hacia la derecha, tienen que caminar como cangrejos.

El/la primer/a estudiante en llegar a usted, gana. Si hay tiempo, permita que otros/as sean “Simón” frente a la clase.

Normas de aprendizaje

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

► **AP** - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 13: Condicionales & Loops en Cosechadora

Condicional | Bucle | Cosechadora

Descripción

Los/as estudiantes practicarán bucles `mientras`, bucles `hasta`, y las declaraciones `si / si no`. 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)

Course D Online Puzzles 2018 - Sitio web

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

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** para encontrar áreas de potenciales problemas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

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

Vocabulario

- **Condición** - una sentencia que un programa verifica si es verdadera o falsa.

Si es verdadera, se realiza una acción. De lo contrario, se ignora.

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

Course D Online Puzzles 2018 - Sitio web

Los/as estudiantes continuarán trabajando con declaraciones 'si/si no', loops 'mientras,' y loops 'hasta'. Estos ejercicios son un poco más desafiantes, por eso, anime a los/as estudiantes a seguir con ellos hasta que sean capaces de describir lo que debe suceder con cada programa.

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?

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 14: Ciudadanía Digital

Common Sense Education | Sin conexión

Descripción

En colaboración con el Sitio Web de Common Sense Education, esta lección ayuda a los/as estudiantes a aprender a pensar, en forma crítica, sobre la información de usuario/a que algunos sitios webs solicitan o requieren. Los/as estudiantes aprenden la diferencia entre información privada e información personal, distinguiendo lo que es seguro e inseguro de compartir online. Los/as estudiantes también exploran lo que significa ser responsables y respetuosos con sus comunidades offline y online, como un paso hacia el aprendizaje de cómo ser buenos ciudadanos digitales.

Propósito

A medida que los/as estudiantes pasan más tiempo en los computadores, deberían estar conscientes que el internet no siempre es un espacio seguro. En esta lección, a los/as estudiantes se les enseña cuál información es segura compartir y qué información debería permanecer privada. Los/as estudiantes crearán “superhéroes” y aprenderán lo que significa ser un Ciudadano Digital en internet.

Agenda

Actividad previa (15 mín)

Vocabulario

Online Personal vs. Privado

Actividad Principal (35 - 40 mín)

Plantillas De Superhéroe de Cubecraft - Manipuladores

Actividad de Cierre (15 mín)

Conversación Rápida: ¿Qué aprendimos?

Escribir en el Diario

Evaluación (5 mín)

Ciudadanía Digital - Evaluación

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Comparar y contrastar las responsabilidades con las comunidades conectadas y sin conexión.
- Comprender qué tipo de información puede ponerlos ante el riesgo de robo de identidad y otras estafas.
- Reflexionar sobre las características que hacen a un ciudadano honrado.
- Diseñar resoluciones para dilemas digitales.

Preparación

- ☐Vea **Ciudadanía Digital - Video para el docente**.
- ☐Imprima una buena selección de las hojas de las plantillas de superhéroes hombre y mujer **Plantillas De Superhéroe de Cubecraft - Manipuladores** para toda la clase.
- ☐Imprima una **Ciudadanía Digital - Evaluación** para cada estudiante.
- ☐Revise **Ciudadanía Digital de Fundamentos de Ciencias de la Computación - Lista de recursos** para más contenido seguro en línea.

Enlaces

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

Para los docentes

- **Ciudadanía Digital** - Guía de respuestas de la evaluación [Hacer una copia](#)
- **Common Sense Education** - Sitio web
- **Ciudadanía Digital de Fundamentos de Ciencias de la Computación** - Lista de recursos

Para los estudiantes

- **Plantillas De Superhéroe de Cubecraft** - Manipuladores
- **Ciudadanía Digital** - Evaluación
- **Diario Think Spot** - Diario de reflexiones

Vocabulario

- **Ciudadano digital** - alguien que actúa de forma segura, responsable y respetuosa en Internet.

Guía docente

Actividad previa (15 mín)

Vocabulario

Esta lección tiene una frase nueva importante:

Ciudadano Digital- Díganlo conmigo: Ciu—da-da-no Di-gi-tal.

Alguien que actúa online con seguridad, responsablemente y respetuosamente.

Online Personal vs. Privado

[Lesson Tip][0]

Si tiene acceso a un computador, vaya a un sitio que requiera este tipo de información, como Gmail o Facebook.

- Pregunte “¿Qué tipos de información crees que está bien compartir públicamente online, o en un perfil que otros podrán ver?”
- ¿Cuáles son algunos ejemplos de sitios webs donde debes registrarte para poder participar?
- Escriba los nombres de los sitios web en la pizarra.
- ¿Qué información se solicita y por qué crees que la piden?
- La información puede ser solicitada para ayudar a distinguir a una persona de otra.
- El sitio web puede mantener un registro de quién lo usa.
- Explique que es importante saber que compartir algún tipo de información como usuario/a, puede poner tu privacidad y la de tu familia en riesgo.
- Señale que no tienen que completar los campos en los sitios webs si no son requeridos.
- Los campos requeridos están usualmente marcados con un asterisco (*) o destacados en rojo.
- Los/as estudiantes de escuelas básicas nunca deberían registrarse en sitios que requieran información privada, sin la aprobación y guía de uno de los padres o tutor.
- Aquí hay un ejemplo de información Pública vs. Privada:

I SEGURO – Información Personal | INSEGURO – Información Privada Información | \ | -----
----- \ | -----
----- \ | | Tu comida favorita
Tu opinión (aunque debe ser dada con respeto)
Nombre de pila (con permiso) | Nombre de tu mamá
Tu fecha de nacimiento
Información de la tarjeta de crédito de tus padres
Tu número de teléfono |.

Actividad Principal (35 - 40 min)



Plantillas De Superhéroe de Cubecraft - Manipuladores

- El Hombre Araña dice: “Con gran poder, viene una gran responsabilidad”. Esto es verdad también cuando trabajamos o jugamos en Internet.
- Lo que leemos, vemos y escuchamos online, puede guiar a la gente a tener todo tipo de sentimientos (ej. feliz, herido/a, emocionado/a enojado/a, curioso/a).
- Lo que hacemos o decimos puede ser poderoso.
- Internet nos permite aprender acerca de todo, hablar con gente a cualquier hora (no importa en qué parte del mundo estén) y compartir nuestro conocimiento y proyectos creativos con otra gente.

- Esto también significa que los comentarios negativos se pueden extender muy rápido a amigos/as de todas las edades.
- Cree un cuadro de tres columnas con los términos “Seguro”, “Responsable”, y “Respetuoso/a”, escritos al inicio de cada columna. Invite a los/s estudiantes a gritar palabras o frases que describan cómo la gente puede actuar de forma segura, responsable y respetuosa online, y luego escríbalos en la columna apropiada.

Seguro	Responsable	Respetuoso	

Actividad de Cierre (15 mín)

Conversación Rápida: ¿Qué aprendimos?

- ¿Cuál es una forma correcta de actuar responsablemente online?
- ¿Qué tipos de información personal puedes compartir de ti mismo sin mostrar tu identidad?
- ¿Qué tipos de súper poderes o cualidades, tienen en común sus superhéroes digitales?
- ¿Qué significa el lema del Hombre Araña “Con gran poder, viene una gran responsabilidad” para ti, como alguien que usa internet?

💡 Lesson Tip

Las preguntas de la Charla Exprés tienen como objetivo incentivar el pensamiento a gran escala, pensando en cómo la lección se relaciona con el gran mundo de los/as estudiantes y su incluso más grande futuro. Utilice el conocimiento de su aula para decidir si quiere discutir las como clase, en grupos pequeños o compañeros/as de banco.

Escribir en el Diario

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

Sugerencias para el diario:

- ¿De qué trató la lección de hoy?
- ¿Cómo te sentiste durante la lección de hoy?
- ¿Qué es un Ciudadano Digital?
- ¿Qué necesitas hacer para ser un Ciudadano Digital?

Evaluación (5 min)

Ciudadanía Digital - Evaluación

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

Aprendizaje Extendido

Usa estas actividades para mejorar el aprendizaje del alumno. Pueden ser usadas como actividades al aire libre u otros enriquecimientos.

- Visita **Common Sense Education - Sitio web** para aprender más acerca de cómo mantener a sus alumnos seguros en esta era digital.

Normas de aprendizaje

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

► **NI** - Redes y la Internet



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

Español (Latinoamérica) ▼

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

Lección 15: Construir un Juego de Play Lab

Laboratorio de juegos | Evento

Descripción

En esta actividad online, los/as estudiantes tendrán la oportunidad de aprender a usar eventos en Play Lab y a aplicar todas las habilidades de programación que han aprendido para crear un juego animado. ¡Es hora de ponerse creativos y hacer un juego en Play Lab!

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)

Introducción

Revisión de "Ciudadanía Digital"

Actividad Principal (30 min)

Course D Online Puzzles 2018 - Sitio web

Actividad de Cierre (15 min)

Escribir en el Diario

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

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

Preparación

- ☐ Juegue a través de **Course D Online Puzzles 2018 - Sitio web** para encontrar áreas de potenciales problemas para su clase.
- ☐ Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones**.
- ☐ Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones**.

Enlaces

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

Para los docentes

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

Para los estudiantes

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

Vocabulario

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

Guía docente

Actividad previa (10 min)

Introducción

Hoy los/as estudiantes serán introducidos a los *eventos* en programación.

Event: Una acción que causa que algo suceda.

Pida a los/as estudiantes que levanten sus manos.

Lo que usted hizo fue declarar un evento. Cuando dice “levanten sus manos” los/as estudiantes responden levantando sus manos. En codificación, usted declararía esto diciendo “cuando diga “levanten sus manos” ustedes levantarán sus manos. Puede también pensar en ciudades declarando eventos. Hay leyes que dicen “cuando hay una luz verde, los autos pueden pasar la intersección”. Pregunte a los/as estudiantes por qué creen que esto es un evento.

Hoy, los/as estudiantes jugarán en el Play Lab, pero los eventos con los que trabajarán serán más como videojuegos que están acostumbrados a jugar. Los eventos tomarán la forma de acciones, tales como presionar la flecha hacia arriba, o dos personajes corriendo hacia ellos.

Revisión de "Ciudadanía Digital"

Recuerde a los/as estudiantes acerca de la información que es segura de compartir online y la que es estrictamente privada.

SEGURA - Información Personal INSEGURA	- Información Privada
Tu comida favorita Tu opinión (siempre entregada respetuosamente) Nombre (con autorización)	Nombre de la madre Número de tarjeta personal social Fecha de nacimiento Número tarjeta de crédito Número de teléfono

Discuta otros ejemplos como los de las categorías de arriba.

Actividad Principal (30 min)

Recuérdelos solo compartir sus trabajos con amigos cercanos o familia. Para más información muestre el video **Detente y Piensa Online - Video**.

Course D Online Puzzles 2018 - Sitio web

Esta es la actividad online más libre del curso. En la etapa final los/as estudiantes tendrán la libertad de crear un juego propio. Puede entregarles guías estructuradas sobre qué tipo de juego hacer, en forma particular a los/as estudiantes que se sienten sobrepasados por tantas opciones.

💡 Lesson Tip

Los/as estudiantes podrán compartir su producto final con un link. Esta es una buena oportunidad para mostrar a la comunidad escolar las cosas maravillosas que sus estudiantes están haciendo. Recolecte todos los links en el sitio web del curso. ¡Para que todos los lean!

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?
- ¿Cuál es un evento que tu programa utilizó hoy?
- ¿Hay algún evento que te gustaría haber usado en tu juego que no hayas podido usar en Play Lab?

Aprendizaje Extendido

Usa estas actividades para mejorar el aprendizaje del alumno. Se pueden usar como actividades al aire libre u otro tipo de enriquecimiento.

Mirar bajo la capucha

Cuando compartes un enlace a tu historia, también compartes todo el código que va de fondo. Esta es una buena forma para que los alumnos aprendan de los demás.

Publica enlaces para historias completas en línea. - ¡También inventa una historia para compartir!

- Cuando los alumnos suban un enlace, pídeles que hagan clic en el botón "Cómo Funciona" para ver el código que está detrás de la historia.
- Discutan como grupo las formas diferentes en que sus compañeros codificaron sus historias.
 - ¿Qué te sorprendió?
 - ¿Qué te gustaría intentar?
- Escoge la historia de alguien más y haz clic en [Remix](#) para construir desde ahí. (No te preocupes, la historia original será guardada).

Normas de aprendizaje

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

- ▶ **AP** - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 16: Más Allá de a Programación: Binario

Binario | Sin conexión

Descripción

Aunque mucha gente piense en lo binario estrictamente como ceros y unos, se le presentará a los/as estudiantes la idea de que la información puede ser representada en una variedad de opciones binarias. Esta lección toma el concepto de un paso más allá a medida que ilustra cómo un computador puede también almacenar incluso información más compleja (como imágenes y colores) en binario.

Propósito

En esta lección los/as estudiantes aprenderán cómo se representa la información de manera tal que un computador pueda interpretarla y almacenarla. Cuando se aprende binario, los/as estudiantes tendrán la posibilidad de escribir códigos y compartirlos con sus pares para verlos como imágenes. Esto se puede relacionar a cómo los computadores leen un programa, lo traducen a binario, usan la información de alguna manera, luego responden de forma que los humanos puedan entender. Por ejemplo, cuando tipeamos una oración en un documento y luego presionamos “guardar”, un computador traduce la oración a binario, almacena la información, luego envía un mensaje indicando que ha sido guardada.

Agenda

Actividad previa (10 min)

Vocabulario

Introducción a Binario

Actividad Principal (20 min)

Más Allá de a Programación: Binario - Ficha

Actividad de Cierre (10 min)

Conversación Rápida: ¿Qué aprendimos?

Escribir en un diario

Evaluación (10 min.)

Imagen Binaria - Evaluación

Aprendizaje Extendido

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Identificar métodos para codificar imágenes en código binario.
- Relacionar imágenes con un par mediante la codificación binaria.
- Reproducir una imagen, basada en código binario.

Preparación

- ☐Vea **Más Allá de a Programación: Binario - Video para el docente.**
- ☐Imprima **Más Allá de a Programación: Binario - Ficha** por pareja.
- ☐Imprima una **Imagen Binaria - Evaluación** por estudiante.
- ☐Junte grupos de objetos que puedan mostrar opuestos a los/as estudiantes, para usar cuando creen sus propias codificaciones binarias (opcional).
- ☐Asegúrese que cada estudiante tenga un **Diario Think Spot - Diario de reflexiones.**

Enlaces

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

Para los docentes

- **Imágenes Binarias** - Guía de respuestas de la evaluación [Hacer una copia](#)
- **Bolas Binarias** - Lección de Thinkersmith

Para los estudiantes

- **Más Allá de a Programación: Binario** - Video sin conexión ([descargar](#))
- **Más Allá de a Programación: Binario** - Ficha [Hacer una copia](#)
- **Imagen Binaria** - Evaluación [Hacer una copia](#)

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

Hacer una copia ▾

Vocabulario

- **Binario** - una forma de representar la información con solo dos opciones.
- **Alfabeto binario** - las dos opciones usadas en tu código binario.

Guía docente

Actividad previa (10 min)

Vocabulario

Esta lección tiene dos palabras nuevas:

- **Binario** - Díganlo conmigo Bi-na-rio

Las dos opciones usadas en su código binario.

Introducción a Binario

Actividad Principal (20 min)

Más Allá de a Programación: Binario - Ficha

¡Es el turno de los/as estudiantes!

Instrucciones para la actividad:

1. Divida a los/as estudiantes en parejas.
2. Pídales que escojan una imagen junto a su compañero/a.
3. Anímelos a descubrir cuál va a ser su código binario.
4. Pídales que codifiquen su imagen usando su nuevo alfabeto binario.
5. Instrúyalos para que intercambien sus codificaciones con otro equipo y vean si pueden descifrar cuál imagen los otros construyeron.
6. Escojan un nivel
 - Fácil: Cuenten al otro equipo cuál fue su método de codificación.
 - Difícil: Pidan al otro equipo que adivinen su método de codificación.

Actividad de Cierre (10 min)

Conversación Rápida: ¿Qué aprendimos?

- ¿Qué aprendimos hoy?
- ¿Qué clase de alfabeto binario creaste?
- ¿Te imaginas cómo puedes codificar una imagen usando solamente tus dedos?
- ¿Crees que podrías crear un alfabeto binario solamente a partir de sonidos?

Escribir en un diario

Hacer que los estudiantes escriban lo que aprendieron, por qué es útil, y cómo se sienten, puede ayudar a consolidar algún conocimiento obtenido y así construir una hoja de repaso para ellos en el futuro.

Sugerencias para el diario:

- ¿De qué se trató la lección de hoy?
- ¿Cómo te sentiste durante la lección?
- ¿Qué es un alfabeto binario?
- ¿Qué clase de información puedes compartir usando binario?

Evaluación (10 min.)

Imagen Binaria - Evaluación

Distribuya esta evaluación para los/as estudiantes para responder en forma individual. Intente dejar un tiempo al final para revisar las respuestas.

Aprendizaje Extendido

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

****Almacenando Imágenes de Color****

- Si su clase entiende realmente la idea detrás del almacenamiento de imágenes binarias, podrían querer saber cómo hacerlo con imágenes de color. - Primero, va a necesitar discutir cómo funciona el color usando binario (as in **Bolas Binarias - Lección de Thinkersmith**, page 21).
 - Luego, presente algunas imágenes que usen combinaciones de esos colores. Anime a los/as estudiantes a crear formas para codificar estas imágenes de colores.



Hexadecimal

- Lleve la idea del color un paso más allá, para presentar **hexadecimal color codes**.

Normas de aprendizaje

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

- ▶ AP - Algoritmos y programación



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

Español (Latinoamérica) ▼

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

Lección 17: Artista Binario

Binario | Artista

Descripción

Esta serie de lecciones online hará que los/as estudiantes aprendan a crear imágenes usando on y off.

Propósito

Esto ayudará a reforzar el hecho de que los computadores pueden hacer una multitud de cosas con los 0s y 1s.

Agenda

Actividad previa (15)

Actividad Principal (30)

Curso D Desafíos Online - Sitio web

Actividad de Cierre (15)

Escribir en el Diario

Ver en Code Studio

Objetivos

Los estudiantes podrán:

- Crear imágenes mediante combinaciones únicas de encendido y apagado
- Identificar secuencias repetidas y dividir códigos largos en trozos más pequeños que se pueden enlazar
- Utilizar bucles y código binario para recrear imágenes proporcionadas

Preparación

- Juegue a través de **Course D Online Puzzles 2018 - Sitio web** correspondiente a este curso para encontrar áreas de potenciales problemas para su clase.
- Revise **Sugerencias para la Actividad Principal de Fundamentos de Ciencias de la Computación - Recomendaciones de lecciones.**
- Asegúrese que cada estudiante tiene un **Diario Think Spot - Diario de reflexiones.**

Enlaces

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

Para los docentes

- **Curso D Desafíos Online** - Sitio web

Vocabulario

- **Binario** - una forma de representar la información con solo dos opciones.

Guía docente

Actividad previa (15)

Para empezar, puede ser útil revisar la lección anterior, específicamente formas diferentes de usar binario para indicar cómo crear una imagen en una cuadrícula. Esta etapa traducirá la actividad sin conexión en una lección simple e independiente en línea.

Actividad Principal (30)

Curso D Desafíos Online - Sitio web

No todas estas imágenes son intuitivas. Anime a los estudiantes a hacer clic en "Ejecutar" para ver lo que sucede, incluso si su código todavía no está "terminado".

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?
- ¿Cómo te sentiste durante la lección?
- ¿Te gustó más dibujar en la cuadrícula de 8x8 o en la de 16x16? Por qué?
- Los computadores también almacenan sonidos usando binarios. Usa tu imaginación para escribir una posible manera de que eso funcione.

Normas de aprendizaje

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

- ▶ **AP** - Algoritmos y programación



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

Español (Latinoamérica) ▼

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