

Lección 7: Programar en Laberinto

Algoritmos | Debugging | Programa | Programar

Reseña

En esta serie de desafíos online, los/as estudiantes desarrollarán la comprensión de los algoritmos, debugging y la alfabetización general para computación. Con la aparición de los personajes de Angry Birds, los/as estudiantes desarrollarán algoritmos secuenciales para hacer que el pájaro llegue al cerdo, sin estrellarse con los muros o el TNT. La aplicación de debugging a los desafíos, también se ha mezclado en esta etapa para adicionar práctica con la solución de problemas y el pensamiento crítico.

Propósito

En esta lección, los/as estudiantes practicarán sus habilidades para hacer debugging y de programación en una plataforma computacional. Cuando alguien comienza a programar ellos/as juntan instrucciones en un orden específico usando algo que una máquina pueda leer. Por medio del uso de la programación, los/as estudiantes desarrollarán la comprensión de cómo un computador navega las instrucciones y en el orden indicado. Hacer debugging es un concepto que es muy importante en la programación computacional. Los/as científicos/as en computación tienen que prepararse para enfrentar todos los bugs en sus propios programas. Hacer debugging fuerza a los/as estudiantes a reconocer los problemas y superarlos mientras construyen el pensamiento crítico y las habilidades de resolución de problemas.

Orden de las Actividades

Actividad Previa (5 min.)

Introducción

Actividad Puente - Programar (10 min.)

Actividad Sin Conexión usando bloques de papel Revisión de los desafíos online como clase

Actividad Principal (30 min.)

Sitio web Curso B Desafíos Online

Actividad de cierre (5 - 10min.)

Escribir en el diario

Aprendizaje Ampliado

Objetivos

Los/as estudiantes serán capaces de:

- Ordenar comandos de movimiento, como pasos secuenciales en un programa.
- Representar un algoritmo como un programa computacional.
- Desarrollar las habilidades de resolución de problemas y pensamiento crítico, mediante la revisión de la práctica del debugging.

Preparación

- Usar el sitio web del Curso B para realizar los desafíos en la etapa 6 para encontrar cualquier área de problema potencial para su clase.
- (Opcional) Elija un par de desafíos para hacer como grupo con su clase.
- Asegúrese de que cada estudiante tenga un Diario Think Spot – Diario de Reflexión.

Links

¡Atención! Haga una copia de cada documento que planea compartir con los/as estudiantes.

Para el/la Profesor/a

- Sitio web - Desafíos online Curso B.
- Recomendaciones para las lecciones – Sugerencias para la actividad Principal del curso fundamentos.

Para los/as estudiantes

- Bloques Blockly Sin Conexión (Curso 1º y 2º Básico) – Manipulativos.
- Diario Think Spot – Diario de Reflexión.

Vocabulario

- **Algoritmo.** Una lista de pasos para terminar una tarea.
- **Bug.** Parte de un programa que no funciona de forma correcta.
- **Debugging.** Encontrar y reparar los problemas de un algoritmo o programa.
- **Programa.** Un algoritmo que ha sido codificado en algo que puede ser ejecutado por una máquina.
- **Programar.** El arte de crear un programa.

Guía Didáctica

Actividad Previa (5 min.)

Introducción

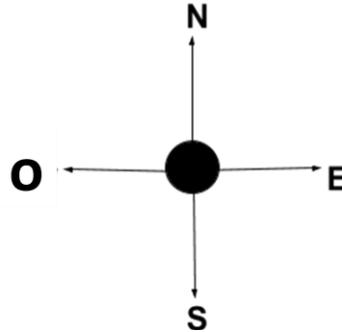
Pregunte a los/as estudiantes si están familiarizados/as con el juego o la película "Angry Birds". Explíqueles que escribirán programas para ayudar a Red, de Angry Birds, a ubicar al Cerdo.

Actividad Puente - Programar (10 min.)

Esta actividad ayudará a traer los conceptos de la sección sin conexión de Mapas Felices al mundo online al que los/as estudiantes están ingresando. Escoja uno de los siguientes para realizar con su clase:

Sugerencia para el/la Profesor/a:

Revise los puntos cardinales con su clase.



Cuéntele a los/as estudiantes que verán esas letras en sus programas, al costado de las flechas de dirección. Recomendamos dibujar las instrucciones en algún lugar que los/as estudiantes puedan mirar para revisar.

Actividad Sin Conexión usando Bloques de Papel.

Seleccione un mapa de Flurb vacío de las tarjetas de las Hojas de Trabajo – y entregue a los/as estudiantes los Manipulativos-Bloques Blockly sin conexión de los Mapas Felices (Cursos 1° - 2°) pre llenados con los comandos cardinales como E → (Este) y O ← (Oeste). Ahora, pida a los/as estudiantes que programen al Flurb desde sus puestos, usando bloques Blockly de papel para que los Flurbs lleguen a la fruta. Asegúrese que entiendan que los bloques necesitan ir de arriba a abajo y que todos ¡Tienen que tocarse!

Pre visualización de los Desafíos Online como Clase

Escoja una lección de los desafíos correspondiente a esta lección. Recomendamos el desafío 7. Usando las flechas de los Manipulativos – piezas de Juegos de los Mapas Felices, pida a los/as estudiantes que preparen un patrón que crean que llevará al pájaro hasta el cerdo. Pida a los/a estudiantes compartir. ¡Ve cuántos otros/as estudiantes tienen la misma respuesta!

Actividad Principal (30 min.)

Demostración del/la Profesor/a

Hemos incluido algunos niveles de predicción con selección múltiple, que son difíciles para los que no leen. Al igual que los desafíos en la Actividad Puente, estos niveles son opcionales para que revise con su clase y ayudar a preparar los desafíos que vienen. Alternativamente, estos podrían ser usados luego de terminar la etapa, como revisión para la clase.

Niveles de predicción:

- Curso B, Programar en Laberinto #1.
- Curso B, Programar en Laberinto #2.

Curso B desafíos online – Sitio Web

Circular: Los/as profesores/as juegan un rol vital en la educación de Ciencias de la Computación y en apoyar un ambiente en la sala de clases que sea colaborativo y motivante. Durante las clases online, el rol del/la docente es principalmente el de motivar y apoyar.

Las clases online están pensadas para ser centradas en los/as estudiantes, por lo que los/as docentes deben evitar intervenir cuando los/as estudiantes se vean estancados/as. Algunas ideas de cómo hacer esto son las siguientes:

- Usar el Video del/la Estudiante sobre Programación en Parejas, cuando sea posible.
- Motivar a los/las estudiantes con preguntas/desafíos que pueden comenzar preguntando a sus compañeros/as.
- Preguntas sin respuesta se pueden compartir con otros grupos, quienes posiblemente ya tengan la solución.
- Recordar a sus estudiantes sobre el proceso de debugging antes de acercarse.
- Haga que sus estudiantes describan el problema que están viendo. ¿Qué se supone que haga? ¿Qué está haciendo? ¿Qué te dice eso?
- Recuerde a aquellos/as estudiantes que se sienten frustrados/as, que la frustración es un paso en el camino del aprendizaje, y que la perseverancia valdrá la pena.
- Si un/a estudiante aún está estancado/a después de todo esto, realice preguntas que lleven a detectar el error por sí mismos/as.

Sugerencia para el/la Profesor/a

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

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

Actividad de Cierre (5 - 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:

- Dibuja una cara en la esquina de tu Diario Think Spot, que demuestre como te sentiste en la clase de hoy.
- Dibuja un pájaro y un cerdo en alguna parte de la hoja. ¿Puedes escribir un programa que haga llegar al pájaro al cerdo?

Aprendizaje Ampliado

En grupos pequeños, deje que los/as estudiantes diseñen sus propios laberintos en papel y que desafíen a otros/as estudiantes o grupos a escribir programas para resolverlos. Para diversión adicional, haga un laberinto de tamaño real con los/as estudiantes, actuando como el pájaro y el cerdo.



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

Si está interesado en licencias de Code.org para propósitos comerciales, **contáctenos**.