

# Lección 7: Depuración con Laurel

Lección en línea [Ver en Code Studio](#)

## Propósito

En esta Lección en línea, los estudiantes practicarán cómo depurar en un ambiente de “recolección”, al practicar cómo leer y editar códigos para reparar desafíos con algoritmos simples, bucles y bucles anidados.

El propósito de esta Lección es enseñar a los estudiantes que fallar es normal cuando se estamos aprendiendo nuevas habilidades. A los estudiantes se les darán programas que NO funcionan correctamente y se les pedirá repararlos. Este proceso, llamado “depuración”, les ayudará a desarrollar habilidades de pensamiento crítico y resolución de problemas, habilidades que los acompañarán mientras avanzan a proyectos de programación cada vez más complejos.

## Secuencia para el aprendizaje

- Conocimiento inicial (15 min)
- Lección puente: depuración (15 min)
- Ampliación del conocimiento (30 min)
- Transferencia del conocimiento (15 min)

## Objetivos

Los estudiantes serán capaces de:

- Leer y comprender un código dado.
- Identificar un bug y los problemas que causa en un programa.

- Describir e implementar un plan para depurar un programa.

## Preparación

- Realice los desafíos para encontrar cualquier área potencialmente problemática para su clase.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

## Recursos

¡Atención!

Por favor, haga una copia de cada documento que planea compartir con los estudiantes.

Para los estudiantes:

- Actividad Previa: [Paquete de lecciones – Programación de relevos](#)
- [Manipulativos – Bloques sin conexión \(cursos C - F\)](#)

## Vocabulario

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

# Estrategia de aprendizaje

## Conocimiento inicial (15 min)

### Introducción

Una de las partes más importantes en el aprendizaje de la programación es aprender a depurar. Pregunte a la clase si alguna vez han aprendido una habilidad nueva y si alguna vez han fallado.

Por ejemplo:

- Aprender a andar en bicicleta y caerse.
- Aprender a hornear y quemar la comida.
- Aprender algún deporte y no ganar ningún juego.

Equivocarse es muy común cuando se aprenden cosas nuevas. Pida a los estudiantes que discutan sobre situaciones en que se hayan equivocado y cómo las superaron.

En la programación, los programadores normalmente encuentran “bugs” en sus códigos.

- Bug (error): parte de un programa que no funciona correctamente.
- Un bug puede realmente arruinar un programa, por lo que es importante aprender a “depurar” el código.
- Depurar (debugging): encontrar y solucionar los problemas en un algoritmo o programa.

Continúe la conversación si cree que su clase necesita una mayor introducción, pero no olvide dejar tiempo para una de las lecciones puente.

## Lección puente: depuración (15 min)

Estas lecciones ayudarán a sus estudiantes a traer los conceptos sin conexión de “depuración sin conexión: programación de relevos” al mundo digital al cual se están sumergiendo. Elija una de las siguientes lecciones:

### Lección sin conexión: bloques de papel

Divida a la clase en equipos de 3 a 5 estudiantes y encuentre un lugar amplio, puede ser un gimnasio o un área verde. Ordene los equipos en fila, similar a como se hizo en la Lección “programación de relevos”. Tome un diseño mediadamente difícil del Paquete de lecciones – Programación de relevos. A una distancia considerable, coloque una imagen por cada equipo. Además, entrégueles suficientes bloques de papel de los Manipulativos – Bloques sin conexión (cursos C - F). Cada equipo necesitará muchos bloques “llena 1” y “mover \_\_\_\_”. Los bloques “mover \_\_\_\_” pueden ser rellenos con anticipación o durante el mismo juego. De cualquier modo, asegúrese de que los bloques estén bien definidos mientras se desarrolle el juego.

Una vez que todos los equipos estén formados, muestre o lea en voz alta las reglas:

- El primer estudiante en la fila debe correr hacia la imagen, revisarla, y colocar el primer bloque de código en el programa para reproducir esa imagen.
- El primer estudiante debe volver y tocar al siguiente de la fila. Luego, debe ponerse al final de la fila.
- La siguiente persona en la fila debe correr hasta la imagen y revisarla. Luego, debe revisar el programa que se ha escrito y hacer una de dos: depurar el programa quitando los bloques de código incorrectos o agregar un bloque nuevo. Luego, el estudiante debe correr de vuelta y toca al siguiente en la fila para que sea su turno.
- Repetir el punto anterior hasta que un equipo termine el programa.

Asegúrese de que los estudiantes usen sólo los bloques “llena 1” y “mover \_\_\_\_” y que sólo coloquen un bloque por turno. ¡El primer equipo que escriba correctamente el código de su imagen, gana!

### Revisión de desafíos en línea en conjunto

Agrupe a los estudiantes en grupos de 3. Elija un desafío de la Lección, recomendamos el número 5. Pídales que se sienten frente a un computador con el desafío en pantalla. Cada equipo puede tener sólo un computador y un estudiante mirando la pantalla. Muestre o lea las reglas:

- Sólo un estudiante por equipo puede estar mirando la pantalla.
- Esa persona sólo puede añadir o borrar un bloque a la vez. Cuando esa persona haya añadido o eliminado un bloque, puede tocar el hombro de un compañero para que sea su turno.
- El siguiente estudiante puede ponerse frente a la pantalla y jugar su turno.
- Los turnos no pueden repetirse ni saltarse, todos deben jugar la misma cantidad de veces.

¡El primer equipo en resolver el desafío correctamente, gana!

## Ampliación del conocimiento (30 min)

### Desafíos en línea

Podría ser muy útil que los integrantes de los equipos de la Lección puente se sienten cerca unos de otros. Cada estudiante debe realizar los desafíos de forma individual o en parejas; sin embargo, contar con un grupo de trabajo conocido para hacer y responder preguntas puede ayudarlos a sentir más confianza y a comprender mejor el tema.

### Lección en Code Studio (link)

Acceder a Code Studio para realizar los desafíos de depuración.

## Transferencia del conocimiento (15 min)

### Escribir en la bitácora

El acto de escribir en su bitácora sobre lo aprendido, respecto de si les pareció útil y de lo que sintieron, ayuda a sus estudiantes a fortalecer cualquier conocimiento que hayan obtenido hoy y servir como un resumen al que puedan recurrir en el futuro.

### Sugerencias para la bitácora

- ¿Sobre qué se trataba 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 “depurar” un código? ¿cómo depuras un programa?

## Sugerencia para evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Comprenden y analizan un problema.
- Piensan en la solución.
- Buscan alternativas; es decir, preparan los elementos para que desarrollen el programa y resuelvan el problema.