

Lección 3

Programación de relevos

Propósito

Esta Lección comenzará con una breve Lección de depuración y perseverancia. Luego, rápidamente se transformará en una carrera contra el tiempo, mientras los estudiantes se dividen en grupos y trabajan en equipo para escribir un programa, una instrucción a la vez.

El trabajo en equipo es muy importante en la computación. Los equipos escriben y depuran códigos en conjunto, en lugar de trabajar de forma individual. Los estudiantes aprenderán a trabajar en equipo intentando ser lo más eficientes posible.

Esta Lección también involucra el concepto de urgencia, lo cual enseñará a los estudiantes que deben administrar su tiempo cuidadosamente, evitando cometer errores sin comprometer mucho tiempo. Esta experiencia puede ser estresante (¡y lo será!). Asegúrese de compartir con los estudiantes las herramientas para lidiar con la frustración.

Secuencia para el aprendizaje

Conocimiento inicial (15 min)

Ampliación del conocimiento (20 min)

Transferencia del conocimiento (15 min)

Objetivos

Los estudiantes serán capaces de:

- Definir ideas usando código y símbolos.
- Verificar el trabajo hecho por compañeros.
- Identificar las señales de la frustración.

Lección sin conexión

Recursos

¡Atención!

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

Para los profesores:

- Guía del profesor – [Programación de relevos.](#)
- Guía de depuración para el profesor – [Programación de relevos.](#)
- Video – [Programación de relevos.](#)

Para los estudiantes:

- Guía de trabajo – [Programación de relevos.](#)
- Video – [Programación de relevos.](#)
- Paquete de actividades – [Programación de relevos.](#)

Vocabulario

- **Algoritmo:** lista de pasos para realizar una tarea.
- **Bug (error):** parte de un programa que no funciona correctamente.

Preparación

- Vea el Video – Programación de relevos.
- Encuentre un espacio abierto para esta Lección, como el gimnasio o un área verde.
- Imprima un Paquete de actividades – Programación de relevos para cada grupo.
- Provea a cada grupo hojas y lápices.
- Imprima una Guía de trabajo – Programación de relevos para cada estudiante.
- Asegúrese de que cada estudiante tenga su [Bitácora](#).

- **Depurar (debugging):** encontrar y solucionar los problemas en un algoritmo o programa.
- **Frustración:** sentirse molesto o enojado porque algo no funciona como quieres.
- **Perseverancia:** intentarlo una y otra vez, incluso cuando algo es muy difícil.
- **Programa:** algoritmo que ha sido codificado de forma que pueda ser ejecutado por una máquina.

Estrategia de aprendizaje

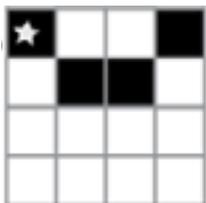
Conocimiento inicial (15 min)

Recordemos que en “programación con papel cuadriculado” guiamos el Sistema de Ejecución Automática (SEA) de nuestros compañeros usando flechas. Esta sección trae de vuelta esas ideas, las cuales serán necesarias en la Ampliación del conocimiento.

¿Dónde me equivoqué?

Objetivo: en esta Lección, queremos ayudar a los estudiantes a identificar y resolver bugs en sus programas. La forma más sencilla de hacerlo es trabajar en un programa que ya contenga bugs, pero que no sean culpa de los estudiantes. Una vez que le hayan ayudado a reparar “su” programa, comparta con ellos lo frustrante que puede llegar a ser cometer errores, y ayúdeles a ver que esos sentimientos son absolutamente normales, no deben sentirse avergonzados por ellos.

Mostrar: Muestre la imagen de la Guía de depuración para el profesor – Programación de relevos.



Análisis: Presente la imagen e indique que tiene un bug ¿pueden identificarlo?

Tómese un momento para repasar las reglas: comenzar en la estrella, seguir las instrucciones paso a paso, terminar cuando se hayan ejecutado todos los cuadrados de la derecha.

Reflexión: ¿pueden descifrar por qué el programa no funciona?

Interacción: pida a los estudiantes que trabajen en equipo para ver si pueden descifrar cómo debería estar escrito el programa.

Exposición: pregunte si alguien encontró una manera de resolver el problema. Cuando le den una respuesta correcta, recuerde la definición de depuración

Sugerencias para lidiar con la frustración:

- Contar hasta 10
- Respirar hondo
- Escribir acerca del problema
- Hablar con algún amigo sobre el problema
- Pedir ayuda

Discusión: En la programación muchas veces nos vamos a ver enfrentados a la frustración, donde quizás te veas tentado a rendirte; sin embargo, la frustración es un sentimiento normal y es un gran indicio de que estás a punto de aprender algo. En lugar de rendirte, intenta ser perseverante. Sigue intentándolo una y otra vez. Después de algunos intentos, ¡empezarás a entender cómo depurar tus problemas!

Distribuir: para asegurarse de que los estudiantes entienden la idea de encontrar y solucionar problemas (depurar), distribuya la Guía de trabajo – programación de relevos y que trabajen en parejas.

Opcional: Si no quiere invertir mucho tiempo en esta Lección, pueden realizarla en conjunto como clase.

Transición: ¡llegó la hora de la acción!

Ampliación del conocimiento (20 min)

Programación de relevos

Con los conceptos de “programación con papel cuadriculado” en mente, ¡es momento de dividirse en equipos y prepararse para la Lección en relevos!

Preparación: prepárese imprimiendo el Paquete de actividades – Programación de relevos para cada equipo, de 4 a 5 estudiantes. Corte o doble cada página por la línea punteada.

Revise las reglas del juego con su clase:

- Dividirse en equipos de 4-5 estudiantes.
- Que cada grupo haga una fila, uno detrás de otro.
- Colocar una de las imágenes guía al otro lado de la sala/gimnasio/área para cada equipo (use la misma imagen para todos los equipos).
- El primer estudiante de la fila debe correr hasta la imagen, revisarla, y escribir el primer símbolo del 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 realizar sólo una de estas dos opciones: depurar el programa

Sugerencias para ser perseverante:

- Llevar registro de lo que ya has intentado
- Describir qué es lo que pasa
- Describir qué debería hacer
- ¿Qué te dice eso?
- Haz un cambio y vuelve a intentar

Algunas cosas que deben ser aclaradas y recordadas de vez en cuando:

- Sólo una persona de cada grupo puede estar frente a la imagen a la vez.
- Está permitido discutir el algoritmo con el resto del grupo mientras están en la fila, incluso pueden planificar quién y qué van a escribir en el programa.
- Cuando un estudiante depura el programa tachando una instrucción incorrecta (o un conjunto de instrucciones incorrectas), cuenta como su turno. El siguiente estudiante debe descifrar cómo corregir las instrucciones eliminadas.

tachando los símbolos incorrectos o agregar un símbolo nuevo. El estudiante luego corre 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.

¡El primer grupo que consiga un programa que calce con la imagen es el ganador! Repita esta Lección varias veces, aumentando la dificultad.

Repita el juego tantas veces como pueda, hasta que se agote el tiempo o hasta que los estudiantes se sientan fatigados.

Transición: una vez que el juego acabe, reúna a todos los estudiantes en un círculo para compartir lo aprendido.

Discusión: ¿qué aprendimos hoy?

- ¿Qué tal si cada persona en la fila pudiese hacer 5 flechas a la vez?
- ¿Qué tan importante puede ser depurar nuestro propio trabajo y el trabajo del programador anterior a nosotros?
- ¿Y si pudiesen hacerse 10 flechas?
- ¿Y si fuesen 10.000? ¿Sería más o menos importante?
- ¿Crees que un programa es mejor o peor cuando más de una persona trabaja en él?
- ¿Crees que las personas comenten más o menos errores cuando están apurados?
- Si encuentras un error ¿debes deshacer todo el programa y comenzar desde cero?

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?
- ¿Cómo es que el trabajo en equipo influyó en el éxito de escribir el programa de hoy?
- ¿Te sentiste frustrado en algún punto? ¿qué hiciste respecto a eso?

Experiencias de aprendizaje de profundización

Use estos contenidos para ampliar el aprendizaje de los estudiantes. Se pueden usar como Contenidos extras fuera del aula.

Pasa la hoja

- Si no tiene tiempo o espacio para una Lección de relevos al aire libre, puede reunir a los estudiantes en grupos y que pasen la hoja con el programa de un puesto a otro. Que cada estudiante haga una flecha antes de pasar la hoja a un compañero.

Llévalo, muévelo

- Como profesor, dibuje una imagen con tantos cuadrados pintados como niños haya en cada grupo.
- Pida a los estudiantes que anoten tantas flechas en el programa como sea necesario para llegar a un cuadrado pintado, incluyendo la acción de pintar ese cuadrado, antes de pasar la hoja al siguiente estudiante.

Depurando juntos

Dibuje una imagen en la pizarra. Pídale a cada estudiante crear un programa para esa imagen. Luego, que intercambien sus programas con sus compañeros de al lado y depuren sus códigos.

- Encierra en un círculo el primer paso incorrecto, luego devuelve el programa.
- Deles a los estudiantes otra oportunidad para revisar y depurar sus propios programas.
- Pida que un voluntario comparta su programa.
- Pregunte:
- ¿Cuántos tienen el mismo programa?
- ¿Alguien tiene algo diferente?

Sugerencias de evaluación

Se sugiere el siguiente indicador para evaluar formativamente los aprendizajes:

- Identifican las partes que componen una situación.
- Elaboran instrucciones.

Lección 4

Depuración con Laurel

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](#).

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

Recursos

¡Atención!

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

Para los estudiantes:

- Act. Previa. Paquete de actividades – [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.
- Entretenga la conversación si cree que su clase necesita una mayor introducción, pero no olvide dejar tiempo para una de las actividades puente.

Lección puente: depuración (15 min)

Estas actividades 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 actividades:

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 actividades – 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.