

Apellidos del profesor: Yildirim	Nombre: Fatma Merve
Título: Atrapa los bichos! Un diseño de juego en Scratch	Duración: 2 horas
Asignatura: TIC	
Objetivos: Competencia general: Conocimiento de conceptos de pensamiento computacional para crear un juego en el que un personaje de rana atrapa insectos, aplicando habilidades básicas de pensamiento computacional Competencia específica: permitir a los estudiantes profundizar su comprensión de una creación sencilla de un juego. Objetivo de la actividad: Cómo crear un juego simple a través del pensamiento computacional	
Elementos clave de CC: Descomposición; Reconocimiento de Patrones; Abstracción; Diseño de Algoritmos	
Grupo de edad: De 6 a 8 años	
Situaciones de Aprendizaje: aula	Tipo de Actividad: extracurricular
Materiales: <ul style="list-style-type: none"> - Ordenadores o tabletas con acceso a Scratch - Proyector o pantalla para demostración - Hojas de trabajo para planificar elementos del juego (Opcional) 	Recursos: <ul style="list-style-type: none"> - Sitios web o vídeos educativos relacionados con Scratch.
Desarrollo del Aprendizaje	
Definición del Problema: Crear los pasos correctos para crear un juego simple utilizando principios de pensamiento computacional.	
Introducción: 1. Introducción al pensamiento computacional Discutir el tema del juego: Explicar que hoy, los estudiantes crearán un juego donde una rana trata de atrapar mariposas, mariquitas y escarabajos. Introducción a los conceptos clave: Breve introducción a las habilidades de pensamiento computacional: Descomposición: Dividir el juego en partes más pequeñas. Reconocimiento de patrones: Identificación de patrones en los movimientos o comportamientos de insectos. Abstracción: Centrarse en detalles esenciales (por ejemplo, movimiento básico en lugar de animaciones complejas). Diseño del algoritmo: planificación de instrucciones paso a paso para que la rana atrape insectos.	

Evaluación Previa (opcional)

2. Planificación de los elementos del juego

1. **Descomposición:** Identificar cada componente del juego:

- Rana (personaje principal)
- Mariposa, mariquita y escarabajo (objetivos)
- Fondo (estanque o hierba)

2. Reconocimiento de patrones.

Reconocer cómo se mueve cada insecto (por ejemplo, las mariposas vuelan, las mariquitas caminan lentamente, los escarabajos gatean).

Pida a los estudiantes que discutan en parejas cómo pueden hacer que cada insecto se mueva de manera diferente para crear patrones.

3. **Abstracción:** Simplificar el diseño centrándose únicamente en los detalles esenciales:

- La rana solo necesita movimientos básicos hacia arriba, abajo, izquierda y derecha.
- Cada insecto requiere patrones de movimiento únicos pero simples para distinguirlos unos de otros.
- Mantener la cuenta de puntuación y "captura" mecánica simple sin animaciones adicionales.

4. **Diseño de algoritmos:** Planificación de la mecánica del juego

Explicar el diseño del algoritmo: Guiar a los estudiantes a pensar paso a paso:

Paso 1: Empieza con un bloque "cuando la bandera verde haga clic" para inicializar el juego.

Paso 2: Programa los movimientos de la rana usando las teclas de flecha.

Paso 3: Establecer los movimientos de insectos utilizando bucles y comandos "mover".

Paso 4: Utilice declaraciones condicionales para detectar cuando la rana toca un insecto:

Aumentar la puntuación en 1.

Hacer que el insecto desaparezca o reaparezca en una ubicación diferente.

Actividad: Pida a los estudiantes que mapeen estos pasos en un diagrama de flujo simple o en papel antes de codificar, asegurando la claridad en sus algoritmos.

Pruebas, depuración y refinamiento

Pruebas: Anime a los estudiantes a ejecutar el juego, observando si los insectos se mueven correctamente y si la puntuación funciona como está planeado.

Depuración: Ayudar a los estudiantes a identificar problemas, como si la rana no se mueve o si los insectos no desaparecen cuando son capturados.

Refinamiento con Abstracción: Pida a los estudiantes que revisen su código y eliminen cualquier bloque innecesario, centrándose en la codificación eficiente para mantener el juego funcional pero simple.

Resumen y reflexión

- **Discusión:** Pida a los estudiantes que compartan lo que aprendieron sobre la planificación de un juego y el uso del pensamiento computacional.
- **Reflexión sobre las habilidades:** ¿Cómo la descomposición, la abstracción, el reconocimiento de patrones y el diseño de algoritmos los ayudaron a crear su juego?

Evaluación:

Observar el compromiso de los estudiantes con cada habilidad de pensamiento computacional durante la lección y su capacidad para completar el juego en Scratch.

Con el diseño de abstracción y algoritmos, los estudiantes se enfocan en lo esencial del juego, creando un flujo de trabajo más suave mientras aprenden los fundamentos del pensamiento computacional. ¡Avísame si quieres añadir más detalles!

Prueba de evaluación posterior (opcional):

Feedback basado en la prueba posterior (opcional):

Resultados esperados:

Al final de la lección, los estudiantes habrán creado un juego funcional de Scratch en el que una rana atrapa varios insectos, demostrando su comprensión de la descomposición, reconocimiento de patrones, abstracción y diseño de algoritmos en pensamiento computacional.

Notas:

Este proyecto también puede vincularse a las clases de TIC para los alumnos de 5,6 años y 10-11 años.