

Descubriendo el Poder de los Algoritmos: ¡Crea Soluciones con Lógica!

Tecnología e Informática | Informática | Aprendizaje Basado en Proyectos

Descripción

Este plan de clase está diseñado para que los estudiantes de secundaria comprendan qué son los algoritmos, cómo funcionan y por qué son esenciales en el mundo digital y cotidiano. A través de un enfoque práctico y colaborativo basado en proyectos, los alumnos aprenderán a diseñar y representar algoritmos para resolver problemas reales, desarrollando pensamiento lógico y habilidades computacionales desde una edad temprana. El aprendizaje no se limitará a la teoría, sino que los estudiantes aplicarán conceptos creando sus propios algoritmos para situaciones que ellos mismos identifiquen, fomentando así la autonomía y la creatividad.

El estudio de los algoritmos es fundamental porque están detrás de las tecnologías que los estudiantes usan a diario, desde videojuegos hasta redes sociales y aplicaciones móviles. Además, el desarrollo de habilidades algorítmicas fortalece capacidades de razonamiento crítico y la resolución de problemas, competencias valiosas para su vida académica y personal. Con este plan, se busca que los estudiantes no solo entiendan la importancia de los algoritmos, sino que se conviertan en creadores activos, capaces de diseñar soluciones innovadoras y eficientes a través del pensamiento lógico estructurado.

Objetivos de Aprendizaje

- Definir y explicar qué es un algoritmo y su importancia en la vida cotidiana y la informática.
- Diseñar algoritmos simples para resolver problemas concretos utilizando diagramas de flujo y pseudocódigo.
- Analizar y mejorar algoritmos para optimizar su eficiencia y claridad.
- Trabajar colaborativamente para crear un proyecto que aplique algoritmos en una situación real.
- Reflexionar sobre el proceso de creación de algoritmos y su aplicación práctica.

Recursos Necesarios

- Computadoras o tabletas con software para crear diagramas de flujo (por ejemplo, draw.io, Lucidchart, o software offline similar) - al menos una por cada 2 estudiantes.
- Hojas impresas con ejemplos de algoritmos y plantillas para diagramas de flujo y pseudocódigo (una por estudiante).
- Proyector y computadora para presentaciones.
- Pizarras blancas y marcadores para trabajo en grupo.
- Materiales para trabajo manual: hojas, colores, reglas, lápices.

- Videos cortos sobre algoritmos (2-3 minutos) para introducción.

Requisitos Previos

- Conocimiento básico del uso de computadora y software simple de dibujo.
- Habilidades básicas de lectura y escritura para comprender instrucciones y redactar pasos.
- Experiencia previa con secuencias lógicas en matemáticas o ciencias (por ejemplo, instrucciones paso a paso o resolución de problemas).
- Capacidad para trabajar en equipo y comunicarse con compañeros.

Actividades

Sesión 1: Introducción a los Algoritmos y su Importancia

Fase de Inicio

Tiempo estimado: 15 minutos

Propósito de la sesión: Comprender qué son los algoritmos y por qué son importantes en la tecnología y en la vida diaria.

Activación de conocimientos previos:

- **Docente:** Muestra un video corto (2 minutos) que explica qué es un algoritmo y ejemplo sencillo (como una receta de cocina).
- **Estudiantes:** Responden a la pregunta: “¿Alguna vez han seguido una receta o una lista de pasos para hacer algo? ¿Qué pasó si no siguieron los pasos correctamente?”

Motivación y enganche:

- **Docente:** Presenta un dato curioso: “¿Sabían que los algoritmos están en casi todos los dispositivos que usan, desde sus teléfonos hasta videojuegos? Sin ellos, nada funcionaría.”
- **Estudiantes:** Expresan sus ideas y se generan expectativas sobre el tema.

Contextualización:

- **Docente:** Explica cómo los algoritmos nos ayudan a resolver problemas diarios, desde organizar horarios hasta navegar en internet.
- **Estudiantes:** Comparten ejemplos de situaciones cotidianas donde creen que usan algoritmos sin saberlo.

Fase de Desarrollo

Tiempo estimado: 95 minutos

Presentación del contenido: Introducción participativa a conceptos clave: definición de algoritmo, pasos, secuencia, entrada, salida. Uso de ejemplos sencillos para ilustrar.

• **Actividad 1: “Descompón la tarea”**

- **Objetivo:** Identificar pasos secuenciales en un problema real.
- **Instrucciones:** En grupos de 3-4, elige una tarea cotidiana (como preparar un sándwich o lavarse las manos). Divídanla en pasos claros y ordenados. Escriban cada paso en una hoja.
- **Producto:** Lista ordenada de pasos o pseudocódigo básico.
- **Tiempo:** 30 minutos.
- **Rol docente:** Circular entre grupos, hacer preguntas como “¿Qué pasa si no haces este paso?”, “¿Se puede hacer otro paso antes?”

• **Actividad 2: “Transforma tu tarea en diagrama de flujo”**

- **Objetivo:** Representar un algoritmo visualmente mediante diagramas de flujo.
- **Instrucciones:** Cada grupo usa papel o software para crear un diagrama de flujo que represente los pasos de su tarea.
- **Producto:** Diagrama de flujo completo.
- **Tiempo:** 40 minutos.
- **Rol docente:** Guiar el uso de símbolos (inicio, proceso, decisión, fin), corregir errores y fomentar claridad.

• **Actividad 3: “Presenta y compara”**

- **Objetivo:** Compartir y analizar diferentes algoritmos para la misma tarea.
- **Instrucciones:** Cada grupo presenta su diagrama al resto y explica su lógica. Se discuten similitudes y diferencias.
- **Producto:** Retroalimentación oral y reflexión grupal.
- **Tiempo:** 25 minutos.
- **Rol docente:** Facilitar preguntas para comparación y reflexión, promover respeto y escucha activa.

Diferenciación:

- Estudiantes que terminan antes pueden diseñar un algoritmo para una tarea más compleja o innovadora.
- Para estudiantes con dificultades, el docente ofrece guías de pasos o plantillas pre-dibujadas para facilitar la creación del diagrama.

Transición:

El docente explica que en la próxima sesión se profundizará en cómo escribir algoritmos usando pseudocódigo y se comenzará un proyecto colaborativo para crear algoritmos que resuelvan problemas reales.

Fase de Cierre

Tiempo estimado: 10 minutos

- **Síntesis:** En plenaria, cada estudiante dice una idea clave que aprendió hoy sobre algoritmos.
- **Reflexión metacognitiva:**

- ¿Qué es un algoritmo en tus propias palabras?
- ¿Para qué crees que sirven los algoritmos en la vida diaria?
- ¿Cuál fue el paso que más te costó entender y por qué?
- **Retroalimentación:** El docente brinda comentarios positivos sobre la participación y claridad de ideas.
- **Transferencia:** Anuncia que en la siguiente sesión comenzarán a escribir algoritmos en pseudocódigo, una forma de describirlos con palabras que las computadoras entienden.
- **Tarea:** Piensa en una actividad diaria que hagas y escribe al menos 5 pasos que sigues para realizarla.

Sesión 2: Escritura de Algoritmos usando Pseudocódigo

Fase de Inicio

Tiempo estimado: 15 minutos

Propósito de la sesión: Comprender y practicar la escritura de algoritmos en pseudocódigo para mejorar la comunicación lógica.

Activación de conocimientos previos:

- **Docente:** Revisa brevemente la tarea de pasos que hicieron y pregunta: “¿Cómo podemos escribir estos pasos para que alguien más los entienda fácilmente?”
- **Estudiantes:** Comparten respuestas y ejemplos de tareas diarias.

Motivación y enganche:

- **Docente:** Presenta un reto: “Vamos a crear instrucciones muy claras, casi como un lenguaje secreto para que solo las computadoras lo entiendan”.
- **Estudiantes:** Se muestran curiosos y motivados para probar el pseudocódigo.

Contextualización:

- **Docente:** Explica que el pseudocódigo es un lenguaje intermedio entre el humano y la computadora, usado para planear programas.
- **Estudiantes:** Relacionan el pseudocódigo con sus conocimientos previos de pasos y diagramas.

Fase de Desarrollo

Tiempo estimado: 95 minutos

Presentación del contenido: Introducción guiada al pseudocódigo con ejemplos claros y sencillos.

- **Actividad 1: “Aprendiendo las reglas básicas”**
 - **Objetivo:** Identificar y usar la estructura básica del pseudocódigo (inicio, fin, instrucciones, decisiones).
 - **Instrucciones:**
 - El docente presenta ejemplos en la pizarra y explica sintaxis básica.
 - Los estudiantes copian y completan ejercicios cortos en equipo.

- **Producto:** Ejercicios escritos de pseudocódigo con correcciones iniciales.
- **Tiempo:** 40 minutos.
- **Rol docente:** Explica, corrige y propone preguntas para clarificar conceptos.

• **Actividad 2: “Escribe tu algoritmo en pseudocódigo”**

- **Objetivo:** Traducir una tarea cotidiana en pseudocódigo.
- **Instrucciones:** En grupos, escriban el pseudocódigo para la tarea que descompusieron en la sesión anterior.
- **Producto:** Documento o papel con pseudocódigo claro y ordenado.
- **Tiempo:** 40 minutos.
- **Rol docente:** Supervisar, orientar y estimular la discusión sobre claridad y uso correcto del lenguaje.

• **Actividad 3: “Intercambio y mejora”**

- **Objetivo:** Revisar y mejorar el pseudocódigo con la retroalimentación de compañeros.
- **Instrucciones:** Los grupos intercambian su trabajo con otro grupo, leen el pseudocódigo y sugieren mejoras o preguntas.
- **Producto:** Retroalimentación escrita o verbal para mejorar el algoritmo.
- **Tiempo:** 15 minutos.
- **Rol docente:** Facilitar la crítica constructiva y promover respeto en los comentarios.

Diferenciación:

- Para estudiantes avanzados, proponer la inclusión de estructuras condicionales más complejas o bucles sencillos.
- Para quienes requieren más apoyo, proporcionar plantillas con lenguaje guiado para completar.

Transición:

Se explica que en la siguiente sesión iniciarán el proyecto grupal para crear un algoritmo que resuelva un problema real usando pseudocódigo y diagramas.

Fase de Cierre

Tiempo estimado: 10 minutos

- **Síntesis:** En grupo, elaboran un breve resumen con las reglas básicas del pseudocódigo en un papel para pegar en el aula.
- **Reflexión metacognitiva:**
 - ¿Qué diferencias encontraste entre escribir pasos en lenguaje común y en pseudocódigo?
 - ¿Qué parte del pseudocódigo te pareció más fácil y cuál más difícil de usar?
 - ¿Cómo te ayudó trabajar en grupo para mejorar tu algoritmo?
- **Retroalimentación:** El docente comenta los avances y destaca buenas prácticas en el uso del pseudocódigo.
- **Transferencia:** Se anticipa que en próximas sesiones se desarrollarán proyectos para aplicar todo lo aprendido.
- **Tarea:** Redacta en pseudocódigo un algoritmo para ordenar tus útiles escolares en tu mochila.

Sesión 3: Inicio del Proyecto: Identificación de Problemas y Diseño de Algoritmos

Fase de Inicio

Tiempo estimado: 15 minutos

Propósito de la sesión: Identificar problemas cotidianos que pueden ser resueltos con algoritmos y formar equipos de trabajo para el proyecto colaborativo.

Activación de conocimientos previos:

- **Docente:** Revisa brevemente los conceptos de algoritmo y pseudocódigo, pregunta: “¿Dónde ven problemas o tareas complicadas en su día a día que podrían simplificarse con un algoritmo?”
- **Estudiantes:** Comparten ideas y ejemplos.

Motivación y enganche:

- **Docente:** Presenta un reto: “Vamos a crear un algoritmo para resolver un problema real que ustedes mismos elijan y que ayude a otros.”
- **Estudiantes:** Se entusiasman con la idea de crear soluciones útiles.

Contextualización:

- **Docente:** Explica cómo los proyectos reales funcionan cuando se diseñan soluciones paso a paso.
- **Estudiantes:** Reflexionan sobre la conexión entre la informática y su vida diaria.

Fase de Desarrollo

Tiempo estimado: 95 minutos

Presentación del contenido: Guía para identificar problemas y plantear soluciones algorítmicas.

- **Actividad 1: “Tormenta de ideas y selección de problema”**
 - **Objetivo:** Elegir un problema real para desarrollar un algoritmo.
 - **Instrucciones:** En grupos de 3-4, hagan lluvia de ideas de problemas cotidianos en casa, escuela o comunidad. Luego elijan uno para trabajar.
 - **Producto:** Lista de problemas y elección de uno con justificación.
 - **Tiempo:** 40 minutos.
 - **Rol docente:** Facilitar ideas, ayudar a enfocar problemas claros y factibles.
- **Actividad 2: “Diseño inicial del algoritmo”**
 - **Objetivo:** Crear un borrador de algoritmo en pseudocódigo para el problema elegido.
 - **Instrucciones:** Escriban los pasos para resolver el problema usando pseudocódigo, apoyándose en plantillas y ejemplos previos.
 - **Producto:** Primer borrador de pseudocódigo.
 - **Tiempo:** 40 minutos.
 - **Rol docente:** Supervisar, corregir y guiar en el orden lógico de los pasos.

• **Actividad 3: “Comparte tu algoritmo y recibe sugerencias”**

- **Objetivo:** Mejorar el algoritmo con retroalimentación.
- **Instrucciones:** Cada grupo presenta su borrador a otro grupo, escucha sugerencias y anota mejoras.
- **Producto:** Lista de sugerencias para mejorar el algoritmo.
- **Tiempo:** 15 minutos.
- **Rol docente:** Promover respeto y enfoque constructivo.

Diferenciación:

- Para estudiantes que avanzan rápido, sugerir incluir decisiones condicionales o variables en su pseudocódigo.
- Para quienes necesitan apoyo, proporcionar ejemplos detallados y acompañamiento individualizado.

Transición:

El docente anuncia que en la próxima sesión continuarán refinando sus algoritmos y comenzarán a crear diagramas de flujo digitales.

Fase de Cierre

Tiempo estimado: 10 minutos

- **Síntesis:** Cada grupo comparte en una frase el problema elegido y el objetivo de su algoritmo.
- **Reflexión metacognitiva:**
 - ¿Por qué es importante elegir un problema claro para diseñar un algoritmo?
 - ¿Qué aprendiste al escuchar las sugerencias de otros grupos?
 - ¿Qué parte del diseño del algoritmo te parece más difícil?
- **Retroalimentación:** Comentarios del docente sobre la pertinencia y claridad de los problemas seleccionados.
- **Transferencia:** Se anticipa el uso de herramientas digitales para mejorar sus algoritmos en la próxima sesión.
- **Tarea:** Investigar y traer ejemplos de algoritmos en aplicaciones que usen diariamente.

Sesión 4: Representación Digital de Algoritmos con Diagramas de Flujo

Fase de Inicio

Tiempo estimado: 15 minutos

Propósito de la sesión: Aprender a usar herramientas digitales para crear diagramas de flujo que representen algoritmos de manera clara y profesional.

Activación de conocimientos previos:

- **Docente:** Revisa los diagramas hechos en papel y pregunta: “¿Qué ventajas creen que tiene hacer diagramas en computadora?”
- **Estudiantes:** Discuten ventajas y comparten ideas.

Motivación y enganche:

- **Docente:** Presenta un tutorial breve de una herramienta digital (por ejemplo, draw.io) y destaca lo útil que será para sus proyectos.
- **Estudiantes:** Se muestran interesados en usar tecnología para crear sus diagramas.

Contextualización:

- **Docente:** Explica cómo los profesionales usan estas herramientas para crear diagramas claros y compartirlos fácilmente.
- **Estudiantes:** Relacionan con experiencias previas de dibujo en computadora o presentación de trabajos.

Fase de Desarrollo

Tiempo estimado: 95 minutos

Presentación del contenido: Demostración guiada y práctica en computadora para crear diagramas de flujo digitales.

• **Actividad 1: “Tutorial guiado”**

- **Objetivo:** Aprender a usar la herramienta digital para diagramas de flujo.
- **Instrucciones:** El docente proyecta y explica paso a paso cómo crear un diagrama simple (inicio, procesos, decisiones, fin).
- **Producto:** Diagrama de ejemplo en pantalla.
- **Tiempo:** 30 minutos.
- **Rol docente:** Mostrar y responder dudas en vivo.

• **Actividad 2: “Crea tu diagrama de flujo digital”**

- **Objetivo:** Representar digitalmente el algoritmo de su proyecto.
- **Instrucciones:** En equipos, cada grupo crea el diagrama digital de su algoritmo en la computadora o tableta.
- **Producto:** Diagrama de flujo digital completo.
- **Tiempo:** 50 minutos.
- **Rol docente:** Supervisar, orientar diseño y corregir errores técnicos o lógicos.

• **Actividad 3: “Presentación rápida y ajustes”**

- **Objetivo:** Compartir avances y recibir retroalimentación para mejoras.
- **Instrucciones:** Algunos grupos presentan su diagrama digital y reciben comentarios.
- **Producto:** Notas para mejorar el diagrama.
- **Tiempo:** 15 minutos.
- **Rol docente:** Facilitar preguntas y promover autoevaluación.

Diferenciación:

- Para estudiantes avanzados, sugerir agregar símbolos de bucles o subprocesos.
- Para estudiantes con dificultades, ofrecer plantillas digitales con símbolos predefinidos.

Transición:

Se explica que en la siguiente sesión comenzarán a ejecutar y probar sus algoritmos, verificando que funcionen correctamente.

Fase de Cierre

Tiempo estimado: 10 minutos

- **Síntesis:** Cada estudiante escribe en una nota adhesiva qué aprendió hoy sobre diagramas digitales y la comparte en un mural.
- **Reflexión metacognitiva:**
 - ¿Qué fue lo más sencillo y lo más difícil al usar la herramienta digital?
 - ¿Cómo te ayudó el diagrama digital a entender mejor tu algoritmo?
 - ¿Qué mejorarías en tu diagrama para que sea más claro?
- **Retroalimentación:** Comentarios orales del docente sobre el uso de la tecnología y claridad de diagramas.
- **Transferencia:** Preparación para la prueba y mejora de sus algoritmos en la siguiente sesión.
- **Tarea:** Revisar en casa el diagrama creado y pensar en posibles mejoras o preguntas para la próxima sesión.

Sesión 5: Prueba y Mejora de Algoritmos

Fase de Inicio

Tiempo estimado: 15 minutos

Propósito de la sesión: Entender la importancia de probar y mejorar los algoritmos para asegurar que funcionen correctamente.

Activación de conocimientos previos:

- **Docente:** Pregunta: “¿Alguna vez tuvieron que corregir una receta o instrucciones porque algo no salió bien? ¿Qué hicieron?”
- **Estudiantes:** Comparten experiencias de prueba y error.

Motivación y enganche:

- **Docente:** Explica que los algoritmos también deben probarse y ajustarse para funcionar bien.
- **Estudiantes:** Se preparan para validar sus proyectos.

Contextualización:

- **Docente:** Muestra un ejemplo de algoritmo con errores y cómo se mejora tras la prueba.
- **Estudiantes:** Reflexionan sobre la importancia de la revisión.

Fase de Desarrollo

Tiempo estimado: 95 minutos

Presentación del contenido: Introducción a la prueba de algoritmos mediante simulaciones y análisis de errores.

- **Actividad 1: “Simulación manual del algoritmo”**

- **Objetivo:** Identificar errores o pasos confusos en el algoritmo al simular su ejecución.
- **Instrucciones:** En grupos, simulan paso a paso la ejecución del algoritmo usando datos de ejemplo y anotan problemas o dudas.
- **Producto:** Lista de observaciones y posibles correcciones.
- **Tiempo:** 40 minutos.
- **Rol docente:** Guiar la simulación y fomentar la detección crítica de errores.

- **Actividad 2: “Revisión y mejora del algoritmo”**

- **Objetivo:** Modificar el pseudocódigo y diagrama para corregir errores y optimizar.
- **Instrucciones:** Aplicar las correcciones y mejorar la claridad y eficiencia del algoritmo.
- **Producto:** Algoritmo corregido y actualizado.
- **Tiempo:** 45 minutos.
- **Rol docente:** Acompañar en la mejora y sugerir optimizaciones.

- **Actividad 3: “Presentación de mejoras”**

- **Objetivo:** Compartir y explicar los cambios realizados y su impacto.
- **Instrucciones:** Cada grupo presenta brevemente las mejoras y recibe comentarios.
- **Producto:** Presentación oral y discusión.
- **Tiempo:** 10 minutos.
- **Rol docente:** Facilitar y valorar la capacidad de mejora y reflexión.

Diferenciación:

- Para estudiantes avanzados, proponer que expliquen las mejoras en términos de eficiencia o legibilidad.
- Para estudiantes que necesitan apoyo, dar ejemplos concretos de errores comunes y cómo corregirlos.

Transición:

El docente comenta que en la próxima sesión presentarán su proyecto final y reflexionarán sobre todo el proceso.

Fase de Cierre

Tiempo estimado: 10 minutos

- **Síntesis:** Cada estudiante escribe en una tarjeta una mejora importante que aprendió a hacer en su algoritmo.
- **Reflexión metacognitiva:**
 - ¿Por qué es importante probar y mejorar un algoritmo?
 - ¿Qué fue lo más difícil al corregir tu algoritmo?
 - ¿Cómo te ayudó trabajar en equipo en esta etapa?
- **Retroalimentación:** Comentarios alentadores y recomendaciones personalizadas.
- **Transferencia:** Preparación para la presentación y reflexión final en la próxima sesión.

- **Tarea:** Practicar explicar su algoritmo y mejoras a un familiar o amigo.

Sesión 6: Presentación Final y Reflexión sobre el Aprendizaje

Fase de Inicio

Tiempo estimado: 15 minutos

Propósito de la sesión: Preparar la presentación final del proyecto y reflexionar sobre el aprendizaje logrado durante el plan.

Activación de conocimientos previos:

- **Docente:** Revisa con preguntas: “¿Cuál fue la parte más importante de tu algoritmo?”, “¿Qué aprendiste sobre trabajar en equipo?”
- **Estudiantes:** Comparten ideas y expectativas para la presentación.

Motivación y enganche:

- **Docente:** Anima a los estudiantes a mostrar con orgullo su trabajo y a ser creativos en la presentación.
- **Estudiantes:** Se muestran motivados y seguros.

Contextualización:

- **Docente:** Explica que esta presentación es una oportunidad para demostrar habilidades y recibir reconocimiento.
- **Estudiantes:** Se preparan para comunicar claramente sus ideas.

Fase de Desarrollo

Tiempo estimado: 95 minutos

Presentación del contenido: Ejecución de presentaciones y discusión final.

- **Actividad 1: “Presentación grupal del proyecto”**

- **Objetivo:** Comunicar claramente el problema, algoritmo y soluciones desarrolladas.
- **Instrucciones:** Cada grupo presenta su algoritmo con su diagrama de flujo digital y explica las decisiones tomadas.
- **Producto:** Presentación oral y visual.
- **Tiempo:** 60 minutos (10 minutos por grupo aprox.).
- **Rol docente:** Escuchar, tomar notas para retroalimentación y generar preguntas para el público.

- **Actividad 2: “Evaluación y retroalimentación entre pares”**

- **Objetivo:** Valorar el trabajo de otros grupos y ofrecer comentarios constructivos.
- **Instrucciones:** Cada estudiante completa una ficha con aspectos positivos y sugerencias para cada presentación.
- **Producto:** Fichas de retroalimentación.
- **Tiempo:** 20 minutos.

- **Rol docente:** Supervisar y moderar comentarios.

- **Actividad 3: “Reflexión grupal y cierre”**

- **Objetivo:** Reflexionar sobre el aprendizaje y la experiencia del proyecto.
- **Instrucciones:** En plenaria, responder a las preguntas finales y compartir aprendizajes personales.
- **Producto:** Discusión y conclusiones escritas en la pizarra.
- **Tiempo:** 15 minutos.
- **Rol docente:** Facilitar la reflexión y destacar logros.

Diferenciación:

- Ofrecer apoyo para preparar presentaciones para estudiantes que lo necesiten.
- Incentivar a estudiantes con mayor confianza a ayudar a compañeros menos seguros.

Transición:

El docente invita a aplicar lo aprendido en otras áreas y proyectos futuros.

Fase de Cierre

Tiempo estimado: 10 minutos

- **Síntesis:** Crear un mapa mental colectivo en la pizarra con las palabras clave aprendidas: algoritmo, pseudocódigo, diagrama de flujo, proyecto, mejora.
- **Reflexión metacognitiva:**
 - ¿Qué habilidad crees que mejorarás gracias a este plan?
 - ¿Cómo puedes usar lo aprendido en otras materias o en tu vida diaria?
 - ¿Qué te gustaría aprender a hacer con algoritmos en el futuro?
- **Retroalimentación:** Comentarios finales del docente y reconocimiento del esfuerzo y trabajo colaborativo.
- **Transferencia:** Se invita a los estudiantes a explorar programación básica por su cuenta o en futuras clases.
- **Tarea:** Reflexionar en un breve escrito sobre cómo los algoritmos pueden ayudar en su vida diaria y futura carrera.

Evaluación

Tipo de evaluación:

- **Diagnóstica:** En la sesión 1, a través de preguntas iniciales y la tarea sobre pasos de una actividad cotidiana.
- **Formativa:** Durante todo el desarrollo del proyecto, con observación directa, revisión de actividades escritas, diagramas y presentaciones intermedias.
- **Sumativa:** Al final del plan, con la presentación grupal del proyecto y evaluación de la calidad del algoritmo diseñado y presentado.

Criterios de evaluación:

- Claridad y correcta definición del concepto de algoritmo (Objetivo 1).
- Capacidad para diseñar algoritmos usando pseudocódigo y diagramas de flujo (Objetivos 2 y 3).
- Colaboración efectiva y participación en el proyecto grupal (Objetivo 4).
- Capacidad para reflexionar y comunicar aprendizajes sobre algoritmos y su aplicación (Objetivo 5).

Instrumentos sugeridos:

- Lista de cotejo para evaluar la presentación oral y visual del proyecto.
- Rúbrica para valorar diagramas de flujo y pseudocódigo (claridad, secuencia, corrección).
- Observación directa y anotaciones del docente durante las actividades.
- Autoevaluación y coevaluación mediante fichas de retroalimentación entre pares.
- Portafolio digital o físico con los productos generados (diagramas, pseudocódigos, reflexiones).

Evidencias de aprendizaje:

- Listas y diagramas de flujo creados en sesiones iniciales.
- Pseudocódigo redactado y mejorado en sesiones intermedias.
- Diagrama de flujo digital finalizado.
- Presentación oral y visual del proyecto final.
- Reflexiones escritas y orales sobre el proceso de aprendizaje.