

Descubriendo la Programación: Algoritmos y Soluciones Tecnológicas

Tecnologías Emergentes e Impacto Social | Inteligencia Artificial | Aprendizaje Basado en Problemas

Descripción

Este plan de clase está diseñado para que adultos en educación para el trabajo comprendan los conceptos fundamentales de la programación, qué son los programas y cómo los algoritmos permiten resolver problemas cotidianos y técnicos. A través de actividades prácticas basadas en problemas reales, los estudiantes descubrirán que detrás de toda solución tecnológica hay un conjunto de instrucciones claras que un computador interpreta para funcionar. Este aprendizaje no solo es relevante para entender la tecnología que nos rodea, sino que también potencia habilidades de pensamiento lógico y resolución de problemas, fundamentales para mejorar su desempeño laboral y adaptarse a un mundo cada vez más digital.

Los estudiantes explorarán la programación desde un enfoque sencillo y contextualizado a su realidad, reconociendo la importancia de los algoritmos para automatizar tareas y facilitar procesos. Este conocimiento les brindará una base sólida para continuar su formación en tecnologías emergentes y ampliar sus oportunidades laborales en el futuro.

Objetivos de Aprendizaje

- Identificar qué es la programación y su función en el desarrollo de soluciones tecnológicas.
- Explicar qué es un algoritmo y cómo se utiliza para resolver problemas de manera ordenada.
- Analizar ejemplos prácticos de algoritmos aplicados a situaciones cotidianas.
- Diseñar un algoritmo simple para resolver un problema básico cotidiano.
- Reconocer la importancia de seguir instrucciones claras para que un computador pueda interpretar un programa.

Recursos Necesarios

- Hojas blancas y marcadores o lápices (1 por estudiante).
- Tarjetas con problemas cotidianos impresos (ejemplos: ordenar una lista de compras, preparar una receta simple).
- Pizarra o rotafolio con marcadores.
- Video corto introductorio sobre programación y algoritmos (3-5 minutos).
- Computadora con proyector o pantalla para mostrar video.
- Ejemplos impresos de pseudocódigo y diagramas de flujo simples.
- Reloj o cronómetro para controlar tiempos.

Requisitos Previos

- Habilidades básicas de lectura y escritura.
- Conocimiento general sobre el uso de computadoras (encender/apagar, uso básico).
- Experiencias previas con resolución de problemas cotidianos.

Actividades

Fase de Inicio

Tiempo estimado: 20 minutos

Propósito de la sesión

Docente: Explica que en esta sesión aprenderán qué es la programación y los algoritmos, y cómo estas herramientas ayudan a resolver problemas en su vida diaria y laboral. Destaca la importancia de entender que detrás de cada tecnología hay instrucciones que una computadora debe seguir.

Estudiantes: Escuchan y se preparan para participar activamente.

Activación de conocimientos previos

Docente: Pregunta a los estudiantes: "¿Alguna vez han seguido una receta para cocinar o una lista de pasos para armar algo? ¿Cómo saben qué hacer primero, segundo y así sucesivamente?"

Estudiantes: Responden y comparten ejemplos de instrucciones o pasos que han seguido en su vida cotidiana.

Motivación y enganche

Docente: Presenta un dato curioso: "¿Sabían que para que un robot limpie una casa, primero alguien debe escribirle un conjunto de instrucciones claras? Esas instrucciones se llaman programas y están formadas por algoritmos, que son como recetas para que las máquinas funcionen."

Luego, muestra un video corto que ilustra qué es la programación y los algoritmos con ejemplos simples.

Estudiantes: Observan el video y reflexionan sobre el contenido.

Contextualización

Docente: Conecta el tema con la vida diaria: "Así como ustedes siguen pasos para hacer una tarea, las computadoras también necesitan que les digamos paso a paso qué hacer para ayudarnos en el trabajo o en casa. Hoy aprenderemos cómo se crean esas instrucciones."

Estudiantes: Comprenden la relación entre sus experiencias y el aprendizaje que viene.

Fase de Desarrollo

Tiempo estimado: 80 minutos

Presentación del contenido

Docente: Introduce brevemente qué es un programa (conjunto de instrucciones para que una computadora realice tareas) y un algoritmo (serie ordenada de pasos para resolver un problema). Explica con ejemplos simples, usando lenguaje cotidiano y apoyándose en la pizarra o rotafolio.

Actividad 1: "Identificando algoritmos en la vida diaria"

- **Objetivo:** Explicar qué es un algoritmo y cómo se aplica en situaciones cotidianas.
- **Instrucciones:**
 - **Docente:** Divide a los estudiantes en grupos de 3-4 personas.
 - Entrega a cada grupo una tarjeta con un problema cotidiano (por ejemplo: ordenar una lista de compras, preparar una taza de café, organizar una mochila).
 - Pide que discutan y escriban, en orden, los pasos que seguirían para resolver ese problema, como si fuera un algoritmo.
 - Luego, cada grupo presenta sus pasos en plenaria.
- **Organización:** Grupos de 3-4 estudiantes.
- **Producto:** Lista ordenada de pasos (algoritmo) para el problema asignado.
- **Tiempo:** 30 minutos.
- **Rol docente:** Observa que los pasos estén claros y en orden lógico, formula preguntas guía como: "¿Qué paso viene primero? ¿Qué pasa si no hacemos este paso?"

Actividad 2: "Creando un algoritmo simple"

- **Objetivo:** Diseñar un algoritmo simple para resolver un problema básico.
- **Instrucciones:**
 - **Docente:** Propone un problema sencillo: "Imaginen que quieren dar indicaciones para que alguien apague una computadora."
 - Pide a los estudiantes que, individualmente, escriban los pasos claros y en orden para dar esas indicaciones.
 - Luego, invita a algunos voluntarios a compartir sus algoritmos y discuten en grupo la claridad y orden de los pasos.
- **Organización:** Individual y luego plenaria.
- **Producto:** Algoritmo escrito con pasos para apagar una computadora.
- **Tiempo:** 25 minutos.
- **Rol docente:** Facilita la discusión, señala ejemplos de instrucciones claras y ordenadas, y pregunta: "¿Este algoritmo ayudaría a alguien que nunca ha apagado una computadora?"

Actividad 3: "Relación entre algoritmos y programación"

- **Objetivo:** Reconocer que los algoritmos son la base de los programas que las computadoras ejecutan.
- **Instrucciones:**

- **Docente:** Explica con un ejemplo sencillo (como una receta de cocina) cómo un algoritmo puede transformarse en un programa que una computadora entiende.
- Pide a los estudiantes que en parejas expliquen con sus palabras por qué es importante que las instrucciones sean claras y en orden para que una computadora funcione correctamente.
- Se realiza una puesta en común donde se resumen estas ideas en la pizarra.
- **Organización:** Parejas y plenaria.
- **Producto:** Explicación oral y resumen en pizarra.
- **Tiempo:** 25 minutos.
- **Rol docente:** Escucha las respuestas, corrige conceptos erróneos y refuerza la relación entre algoritmos y programación.

Diferenciación

- **Para estudiantes que terminan antes:** Se les invita a pensar y escribir un algoritmo para un problema un poco más complejo, como organizar una agenda diaria.
- **Para estudiantes que necesitan más apoyo:** Se les ofrece ejemplos más guiados y se trabaja en parejas con apoyo directo del docente, usando preguntas específicas para avanzar paso a paso.

Transiciones

Docente: Conecta las actividades resaltando que cada paso que diseñaron es como una instrucción que un programa debe seguir, y que entender esto es clave para aprender a programar en el futuro.

Fase de Cierre

Tiempo estimado: 20 minutos

Síntesis

Docente: Solicita a los estudiantes que, en forma individual, escriban en una hoja tres ideas clave que aprendieron sobre programación y algoritmos, y las compartan con el grupo para crear un mapa mental colectivo en la pizarra.

Estudiantes: Escriben y participan en la construcción del mapa mental.

Reflexión metacognitiva

- ¿Cómo te ayudó entender qué es un algoritmo para resolver problemas?
- ¿Por qué crees que es importante que las instrucciones para una computadora sean claras y en orden?
- ¿En qué situaciones de tu vida diaria crees que podrías aplicar lo aprendido hoy?

Docente: Invita a compartir respuestas y reflexiona con ellos sobre la utilidad práctica del tema.

Retroalimentación

Docente: Proporciona comentarios verbales positivos y constructivos sobre los productos generados, destacando el esfuerzo y aclarando dudas restantes. Anima a los estudiantes a seguir practicando la creación de algoritmos simples.

Transferencia

Docente: Relaciona lo aprendido con futuros temas en Inteligencia Artificial, explicando que la programación es la base para crear soluciones inteligentes que impactan en muchas áreas laborales.

Tarea o reto

Docente: Propone que cada estudiante observe una tarea cotidiana que realiza en casa o en el trabajo y escriba un algoritmo sencillo que describa los pasos para realizarla, para compartirlo en la próxima sesión o con compañeros.

Evaluación

Tipo de evaluación: Diagnóstica al inicio con preguntas activadoras; formativa durante las actividades de desarrollo mediante observación y retroalimentación; sumativa al cierre con síntesis y reflexión.

Criterios de evaluación:

- Reconoce y describe qué es la programación y su función (Objetivo 1).
- Demuestra comprensión del concepto de algoritmo mediante la identificación y explicación de pasos (Objetivo 2 y 3).
- Elabora un algoritmo simple organizado y claro para un problema dado (Objetivo 4).
- Relaciona correctamente la importancia de instrucciones claras para el funcionamiento de un computador (Objetivo 5).

Instrumentos sugeridos:

- Lista de cotejo para observar participación y calidad de algoritmos diseñados.
- Rúbrica sencilla para evaluar claridad, orden y relevancia de los pasos en los algoritmos.
- Observación directa durante discusiones y presentaciones.
- Autoevaluación en la reflexión metacognitiva.

Evidencias de aprendizaje:

- Listas de pasos (algoritmos) generadas en actividades grupales e individuales.
- Respuestas en reflexiones escritas y orales.
- Mapa mental colectivo que sintetiza conceptos clave.