

Algoritmos Secuenciales: Fundamentos del Pensamiento Computacional

Tecnología e Informática | Pensamiento Computacional | para estudiantes de secundaria (12-15 años) | 4 semanas

Descripción del Curso

Este curso introduce a los estudiantes de secundaria en el fascinante mundo de los algoritmos secuenciales, una piedra angular del pensamiento computacional. A lo largo de cuatro semanas, los alumnos explorarán los conceptos básicos para entender cómo se diseñan y ejecutan instrucciones paso a paso para resolver problemas cotidianos y tecnológicos.

Dirigido a estudiantes de 12 a 15 años, el curso adopta un enfoque práctico y participativo, promoviendo la comprensión activa mediante ejemplos, actividades y ejercicios que refuercen el razonamiento lógico. Se enfatiza la importancia de estructurar secuencias claras de instrucciones que puedan ser interpretadas correctamente por personas y máquinas.

Al finalizar, los estudiantes serán capaces de diseñar, analizar y representar algoritmos secuenciales simples, aplicando técnicas básicas de resolución de problemas y sentando las bases para futuros aprendizajes en programación y ciencias de la computación.

Objetivos Generales

- Identificar y describir las características principales de los algoritmos secuenciales.
- Construir representaciones gráficas y escritas de algoritmos secuenciales para resolver problemas específicos.
- Evaluar y mejorar algoritmos mediante la detección de errores y la optimización de pasos.
- Aplicar técnicas básicas de pensamiento computacional para planificar soluciones mediante secuencias lógicas.

Competencias

- Comprender y explicar el concepto de algoritmo secuencial y su importancia en la resolución de problemas.
- Diseñar algoritmos secuenciales claros y precisos para tareas simples utilizando diagramas de flujo y pseudocódigo.
- Analizar y corregir algoritmos secuenciales para garantizar su correcta ejecución.
- Aplicar el pensamiento lógico para descomponer problemas en pasos secuenciales ordenados.
- Comunicar soluciones algorítmicas de forma estructurada y coherente.

Requerimientos

- Conocimientos básicos de informática y uso de dispositivos digitales (computadoras o tabletas).

- Habilidades elementales en lectura y comprensión de instrucciones escritas.
- Materiales: cuaderno para anotaciones, acceso a software básico para diagramas (opcional).
- Actitud abierta para el trabajo colaborativo y la resolución de problemas.

Unidades del Curso

Unidad 1: Introducción a los algoritmos y el pensamiento computacional

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de definir los conceptos de algoritmo y pensamiento computacional usando sus propias palabras en un resumen escrito.
- Al finalizar la unidad, el estudiante será capaz de identificar y explicar la importancia de los pasos secuenciales en la resolución de problemas mediante ejemplos cotidianos.
- Al finalizar la unidad, el estudiante será capaz de construir diagramas de flujo básicos que representen algoritmos secuenciales para resolver problemas simples.
- Al finalizar la unidad, el estudiante será capaz de analizar un algoritmo secuencial dado para detectar errores lógicos y proponer mejoras en su secuencia.
- Al finalizar la unidad, el estudiante será capaz de aplicar técnicas básicas de pensamiento computacional para planificar y describir soluciones secuenciales a problemas planteados en clase.

Contenidos Temáticos

1. Introducción al pensamiento computacional

- Definición y relevancia del pensamiento computacional en la vida diaria
- Componentes del pensamiento computacional: descomposición, reconocimiento de patrones, abstracción y algoritmos
- Ejemplos simples de pensamiento computacional en actividades cotidianas

2. Concepto de algoritmo

- Definición de algoritmo como conjunto de instrucciones claras, finitas y ordenadas para resolver un problema
- Características fundamentales de un algoritmo: secuencialidad, precisión, y finitud
- Importancia de los algoritmos en la tecnología y en la vida diaria

3. Pasos secuenciales en la resolución de problemas

- La estructura secuencial: ejecución paso a paso
- Ejemplos cotidianos que requieren procesos secuenciales (preparar un sándwich, seguir una receta, rutina de estudio)

- La importancia del orden correcto en los pasos para obtener resultados adecuados

4. Representación de algoritmos mediante diagramas de flujo

- Elementos básicos de un diagrama de flujo: inicio, proceso, decisión, fin
- Símbolos y convenciones para diagramas de flujo secuenciales
- Construcción de diagramas de flujo para problemas simples

5. Análisis y mejora de algoritmos secuenciales

- Identificación de errores lógicos en algoritmos secuenciales
- Técnicas para corregir errores y optimizar la secuencia de pasos
- Ejemplos prácticos de revisión y mejora de algoritmos dados

6. Aplicación del pensamiento computacional para planificar soluciones

- Uso de técnicas de descomposición y secuenciación para planificar soluciones
- Elaboración de descripciones secuenciales para resolver problemas planteados en clase
- Integración de los conceptos aprendidos para diseñar soluciones claras y efectivas

Actividades

Actividad 1: Redactando definiciones propias de algoritmo y pensamiento computacional

Objetivo: Definir los conceptos de algoritmo y pensamiento computacional usando sus propias palabras en un resumen escrito.

Descripción:

- El docente presenta definiciones básicas y ejemplos simples de algoritmo y pensamiento computacional.
- Los estudiantes, de forma individual, redactan un breve resumen explicando con sus propias palabras qué entienden por algoritmo y pensamiento computacional.
- Comparan sus resúmenes con un compañero para discutir y enriquecer sus definiciones.
- Finalmente, cada estudiante ajusta su resumen integrando las ideas discutidas.

Organización: Individual, luego parejas

Producto esperado: Resumen escrito individual sobre algoritmo y pensamiento computacional

Duración estimada: 45 minutos

Actividad 2: Identificando pasos secuenciales en actividades cotidianas

Objetivo: Identificar y explicar la importancia de los pasos secuenciales en la resolución de problemas mediante ejemplos cotidianos.

Descripción:

- El docente propone ejemplos cotidianos (preparar un sándwich, cepillarse los dientes, ordenar el escritorio).

- En grupos pequeños, los estudiantes listan los pasos para realizar una actividad asignada, asegurándose de respetar el orden lógico.
- Discuten por qué el orden de los pasos es importante y qué pasaría si se altera.
- Presentan al grupo clase su lista y explicación.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Lista de pasos secuenciales con explicación del orden lógico

Duración estimada: 50 minutos

Actividad 3: Creación de diagramas de flujo para problemas simples

Objetivo: Construir diagramas de flujo básicos que representen algoritmos secuenciales para resolver problemas simples.

Descripción:

- El docente explica los símbolos básicos de diagramas de flujo y muestra ejemplos sencillos.
- Los estudiantes reciben un problema simple (por ejemplo, calcular el promedio de dos números) y elaboran su diagrama de flujo.
- Se realiza una puesta en común en parejas para revisar y mejorar los diagramas.
- Se presenta un diagrama corregido al grupo para recibir retroalimentación colectiva.

Organización: Individual y luego parejas

Producto esperado: Diagrama de flujo dibujado a mano o digital que represente un algoritmo secuencial

Duración estimada: 1 hora

Actividad 4: Análisis y mejora de un algoritmo secuencial dado

Objetivo: Analizar un algoritmo secuencial dado para detectar errores lógicos y proponer mejoras en su secuencia.

Descripción:

- El docente entrega a cada grupo un algoritmo secuencial con errores lógicos (por ejemplo, pasos desordenados o faltantes).
- Los estudiantes trabajan en equipo para identificar errores y discutir cómo corregirlos.
- Proponen una versión mejorada del algoritmo, ya sea en texto o diagrama de flujo.
- Cada grupo presenta su análisis y solución al resto de la clase.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Informe o diagrama corregido con justificación de las mejoras

Duración estimada: 1 hora

Actividad 5: Planificación y descripción secuencial para resolver un problema planteado

Objetivo: Aplicar técnicas básicas de pensamiento computacional para planificar y describir soluciones secuenciales a problemas planteados en clase.

Descripción:

- El docente plantea un problema cotidiano sencillo que requiere solución paso a paso (por ejemplo, organizar una mochila para la escuela).
- Los estudiantes aplican descomposición para dividir el problema en partes más pequeñas.
- Planifican la solución definiendo la secuencia de pasos necesarios para resolver el problema.
- Redactan una descripción clara y ordenada de la solución secuencial.
- Se realiza una puesta en común para comparar diferentes soluciones y discutir su efectividad.

Organización: Individual y luego discusión en grupo clase

Producto esperado: Planificación escrita con descripción secuencial de la solución

Duración estimada: 1 hora

Evaluación

Evaluación diagnóstica

Qué se evalúa: Conocimientos previos sobre algoritmo, pensamiento computacional y secuencias de pasos.

Cómo se evalúa: Preguntas orales y escritas breves al inicio de la unidad para conocer ideas previas.

Instrumento sugerido: Cuestionario de preguntas abiertas y discusión guiada.

Evaluación formativa

Qué se evalúa: Progreso en la comprensión y aplicación de conceptos, elaboración de diagramas de flujo, análisis de algoritmos y desarrollo de soluciones secuenciales.

Cómo se evalúa: Revisión continua de actividades prácticas, retroalimentación en grupo, observación de participación y corrección de productos parciales.

Instrumento sugerido: Rúbricas para diagramas de flujo y análisis de algoritmos, listas de cotejo para participación y cumplimiento de tareas.

Evaluación sumativa

Qué se evalúa: Capacidad para definir conceptos con sus propias palabras, identificar y explicar pasos secuenciales, construir diagramas de flujo básicos, analizar y corregir algoritmos, y planificar soluciones aplicando pensamiento computacional.

Cómo se evalúa: Examen escrito que incluye redacción de definiciones, resolución de problemas con diagramas de flujo, análisis crítico de un algoritmo dado y planteamiento de una solución secuencial.

Instrumento sugerido: Prueba escrita con preguntas abiertas y ejercicios prácticos; rúbrica para evaluación de diagramas y análisis.

Unidad 2: Elementos y estructura de los algoritmos secuenciales

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar los componentes básicos de un algoritmo secuencial en ejemplos escritos y gráficos con una precisión del 90%.
- Al finalizar la unidad, el estudiante será capaz de clasificar los diferentes tipos de instrucciones que conforman un algoritmo secuencial, justificando su función en la estructura del algoritmo.
- Al finalizar la unidad, el estudiante será capaz de ordenar correctamente los pasos de un algoritmo secuencial para resolver un problema específico, garantizando la claridad y la eficacia en la solución propuesta.
- Al finalizar la unidad, el estudiante será capaz de diseñar un algoritmo secuencial básico que incluya instrucciones ordenadas y claras para una tarea sencilla, utilizando diagramas de flujo o pseudocódigo.

Contenidos Temáticos

1. Introducción a los algoritmos secuenciales

- Definición de algoritmo secuencial: explicación sencilla y ejemplos cotidianos.
- Importancia de los algoritmos en la vida diaria y en la informática.
- Concepto de secuencia: pasos que se ejecutan uno tras otro sin saltos ni decisiones.

2. Componentes básicos de un algoritmo secuencial

- Datos de entrada: qué son y ejemplos (números, textos, valores).
- Procesos: operaciones o cálculos que se realizan con los datos.
- Datos de salida: resultado obtenido tras ejecutar el algoritmo.
- Ejemplos gráficos y escritos: diagramas de flujo y pseudocódigo simples.

3. Tipos de instrucciones en un algoritmo secuencial

- Instrucciones de entrada: cómo se reciben datos (por ejemplo, "Leer número").
- Instrucciones de proceso: operaciones aritméticas, asignaciones, etc.
- Instrucciones de salida: mostrar resultados (por ejemplo, "Mostrar suma").
- Justificación de la función de cada tipo en la estructura del algoritmo.

4. Ordenamiento de pasos en un algoritmo secuencial

- Importancia de la secuencia correcta para la claridad y eficacia.
- Consejos para ordenar pasos: lógica, claridad y evitar redundancias.
- Ejemplos prácticos de ordenamiento de pasos incorrectos y corrección.
- Identificación de errores comunes en el ordenamiento.

5. Diseño de algoritmos secuenciales básicos

- Uso de diagramas de flujo: símbolos básicos y convenciones.
- Uso de pseudocódigo: estructura y lenguaje sencillo.

- Ejemplos completos: desde la descripción del problema hasta el algoritmo.
- Práctica guiada para diseñar algoritmos secuenciales para tareas sencillas.

Actividades

Actividad 1: Identificación de componentes en algoritmos

Objetivo: Identificar los componentes básicos de un algoritmo secuencial en ejemplos escritos y gráficos.

Descripción paso a paso:

- El docente presenta varios ejemplos de algoritmos secuenciales escritos y en diagramas de flujo.
- Los estudiantes, de forma individual, subrayan o señalan las partes de entrada, proceso y salida en cada ejemplo.
- Discusión en grupo sobre las respuestas y aclaración de dudas.

Organización: Individual y posteriormente grupo completo.

Producto esperado: Lista o marcado de componentes en los ejemplos entregados.

Duración estimada: 40 minutos.

Actividad 2: Clasificación de instrucciones en un algoritmo dado

Objetivo: Clasificar diferentes tipos de instrucciones y justificar su función en la estructura del algoritmo.

Descripción paso a paso:

- Se entrega a los estudiantes un algoritmo secuencial escrito con instrucciones mezcladas.
- En parejas, clasifican cada instrucción como entrada, proceso o salida.
- Cada pareja explica ante el grupo por qué ubicaron cada instrucción en su categoría.
- El docente complementa y corrige, enfatizando la función de cada instrucción.

Organización: Parejas y grupo completo.

Producto esperado: Listado clasificado de instrucciones con justificaciones.

Duración estimada: 50 minutos.

Actividad 3: Ordenamiento de pasos para resolver un problema

Objetivo: Ordenar correctamente los pasos de un algoritmo para garantizar claridad y eficacia.

Descripción paso a paso:

- Se plantea un problema sencillo (por ejemplo, calcular el área de un rectángulo).
- Se entrega a cada grupo un listado desordenado de pasos para resolver el problema.
- Los estudiantes ordenan los pasos en secuencia lógica y los escriben en forma de pseudocódigo.
- Cada grupo presenta su solución y se discuten las diferencias.

Organización: Grupos pequeños (3-4 estudiantes).

Producto esperado: Pseudocódigo ordenado y claro para resolver el problema.

Duración estimada: 60 minutos.

Actividad 4: Diseño de un algoritmo secuencial básico

Objetivo: Diseñar un algoritmo secuencial básico que incluya instrucciones ordenadas para una tarea sencilla utilizando diagramas de flujo o pseudocódigo.

Descripción paso a paso:

- Se presenta un problema simple (por ejemplo, convertir grados Celsius a Fahrenheit).
- Individualmente, los estudiantes diseñan un algoritmo secuencial en pseudocódigo o diagrama de flujo.
- Se promueve la revisión entre pares para sugerir mejoras en claridad y orden.
- Finalmente, cada estudiante entrega su diseño corregido.

Organización: Individual con revisión en parejas.

Producto esperado: Algoritmo secuencial claro y ordenado en pseudocódigo o diagrama de flujo.

Duración estimada: 90 minutos.

Evaluación

Evaluación diagnóstica

Qué se evalúa: Conocimientos previos sobre algoritmos y comprensión básica de secuencia de pasos.

Cómo se evalúa: Preguntas cortas y ejercicios simples donde identifiquen pasos en un algoritmo básico.

Instrumento sugerido: Cuestionario escrito con ejemplos breves y preguntas de selección múltiple y respuesta abierta.

Evaluación formativa

Qué se evalúa: Progreso en la identificación de componentes, clasificación y ordenamiento de instrucciones en algoritmos secuenciales.

Cómo se evalúa: Observación de actividades en clase, revisión de productos parciales (listas, clasificaciones, pseudocódigos).

Instrumento sugerido: Rúbrica de observación para actividades prácticas y revisión de trabajos escritos.

Evaluación sumativa

Qué se evalúa: Capacidad para diseñar un algoritmo secuencial básico con instrucciones ordenadas y claras, utilizando diagramas de flujo o pseudocódigo.

Cómo se evalúa: Proyecto final individual donde se presenta el diseño completo de un algoritmo para un problema sencillo.

Instrumento sugerido: Rúbrica de evaluación que incluye criterios de identificación correcta de componentes, clasificación adecuada, orden lógico y claridad en la presentación.

Unidad 3: Representación gráfica y textual de algoritmos

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de identificar los símbolos y estructuras básicas de los diagramas de flujo para representar algoritmos secuenciales correctamente.
- Al finalizar la unidad, el estudiante será capaz de redactar pseudocódigo sencillo que describa secuencias lógicas para resolver problemas específicos.
- Al finalizar la unidad, el estudiante será capaz de construir diagramas de flujo y pseudocódigo que representen algoritmos secuenciales de manera clara y organizada.
- Al finalizar la unidad, el estudiante será capaz de comparar una representación gráfica y una textual de un mismo algoritmo para evaluar su precisión y coherencia.
- Al finalizar la unidad, el estudiante será capaz de corregir errores básicos en diagramas de flujo y pseudocódigo para mejorar la exactitud de los algoritmos representados.

Contenidos Temáticos

1. Introducción a la representación de algoritmos

- Concepto de algoritmo y su importancia en la resolución de problemas.
- Diferencia entre representación gráfica y textual de algoritmos.
- Ventajas de utilizar diagramas de flujo y pseudocódigo.

2. Diagramas de flujo: símbolos y estructuras básicas

- Descripción y función de los símbolos principales:
 - Óvalo: inicio y fin del algoritmo.
 - Paralelogramo: entrada y salida de datos.
 - Rectángulo: proceso o instrucción.
 - Flechas: dirección del flujo.
- Estructuras secuenciales en diagramas de flujo:
 - Representación lineal de pasos.
 - Cómo conectar los símbolos para formar secuencias coherentes.
- Normas básicas para construir diagramas de flujo claros y organizados.

3. Pseudocódigo: conceptos y redacción básica

- Definición y propósito del pseudocódigo.
- Elementos básicos del pseudocódigo:
 - Instrucciones de entrada y salida.
 - Asignación de variables.

- Secuencia lógica de instrucciones.
- Formato y sintaxis sencilla para redactar pseudocódigo.

4. Construcción de algoritmos secuenciales: diagramas de flujo y pseudocódigo

- Análisis de problemas sencillos para diseñar algoritmos secuenciales.
- Pasos para construir un diagrama de flujo que represente el algoritmo.
- Pasos para redactar el pseudocódigo correspondiente.
- Buenas prácticas para asegurar claridad y organización en ambas representaciones.

5. Comparación y evaluación de representaciones de un mismo algoritmo

- Cómo leer y comparar un diagrama de flujo y un pseudocódigo equivalentes.
- Criterios para evaluar precisión, coherencia y completitud.
- Identificación de diferencias y errores comunes en ambas representaciones.

6. Corrección de errores básicos en diagramas de flujo y pseudocódigo

- Análisis de errores frecuentes:
 - Falta de símbolos o instrucciones.
 - Secuencias incorrectas o incongruentes.
 - Errores en la lógica de entrada y salida.
- Estrategias para corregir y mejorar la exactitud de los algoritmos.
- Práctica de revisión y edición de ejemplos con errores.

Actividades

Actividad 1: Identificación y uso de símbolos en diagramas de flujo

Objetivo: Identificar los símbolos y estructuras básicas de los diagramas de flujo para representar algoritmos secuenciales correctamente.

Descripción:

- El docente presenta imágenes con diferentes símbolos de diagramas de flujo y explica su función.
- Los estudiantes reciben una lista de pasos para un problema sencillo y deben asociar cada paso con el símbolo adecuado.
- En parejas, realizan un diagrama de flujo básico con los símbolos correctos para el problema dado.
- Comparten y analizan sus diagramas con el grupo para identificar aciertos y aspectos a mejorar.

Organización: Parejas

Producto esperado: Diagrama de flujo básico con símbolos correctamente aplicados.

Duración estimada: 45 minutos

Actividad 2: Redacción de pseudocódigo sencillo para problemas cotidianos

Objetivo: Redactar pseudocódigo sencillo que describa secuencias lógicas para resolver problemas específicos.

Descripción:

- El docente explica la estructura básica y convenciones del pseudocódigo.
- Los estudiantes reciben un problema cotidiano (por ejemplo, calcular el área de un rectángulo) y redactan el pseudocódigo que resuelve el problema.
- En grupos pequeños, comparten sus pseudocódigos para comparar y sugerir mejoras.
- El docente revisa y retroalimenta para asegurar comprensión y corrección.

Organización: Individual y grupos pequeños

Producto esperado: Pseudocódigo claro y organizado que describa la solución al problema.

Duración estimada: 50 minutos

Actividad 3: Construcción conjunta de algoritmo en diagrama de flujo y pseudocódigo

Objetivo: Construir diagramas de flujo y pseudocódigo que representen algoritmos secuenciales de manera clara y organizada.

Descripción:

- Se presenta un problema de complejidad moderada (por ejemplo, determinar el mayor de tres números).
- En grupos, los estudiantes diseñan primero el diagrama de flujo para resolver el problema.
- Luego, redactan el pseudocódigo que corresponda al diagrama.
- Finalmente, cada grupo presenta ambas representaciones y explica su lógica al resto de la clase.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Diagrama de flujo y pseudocódigo completos y coherentes.

Duración estimada: 1 hora 15 minutos

Actividad 4: Análisis y corrección de errores en algoritmos representados

Objetivo: Comparar una representación gráfica y textual de un mismo algoritmo para evaluar su precisión y coherencia, y corregir errores básicos para mejorar la exactitud.

Descripción:

- Se entregan a los estudiantes ejemplos de diagramas de flujo y pseudocódigos con errores intencionales.
- Individualmente, identifican y anotan los errores encontrados en cada representación.
- En parejas, discuten las correcciones posibles y reformulan el algoritmo corregido tanto en diagrama como en pseudocódigo.
- Se realiza una puesta en común para compartir las correcciones y justificar los cambios.

Organización: Individual y parejas

Producto esperado: Listado de errores identificados y versiones corregidas de diagramas y pseudocódigos.

Duración estimada: 1 hora

Evaluación

Evaluación diagnóstica

¿Qué se evalúa? Conocimientos previos sobre algoritmos, símbolos de diagramas de flujo y nociones básicas de pseudocódigo.

¿Cómo se evalúa? Mediante un cuestionario corto con preguntas de selección múltiple y ejercicios simples para identificar símbolos y escribir un pseudocódigo básico.

Instrumento sugerido: Cuestionario escrito o digital con preguntas tipo test y ejercicios cortos.

Evaluación formativa

¿Qué se evalúa? Progreso en la construcción y redacción de diagramas de flujo y pseudocódigo, capacidad para comparar y corregir errores.

¿Cómo se evalúa? Observación directa durante actividades, revisión de productos parciales (diagramas y pseudocódigos), retroalimentación oral y escrita.

Instrumento sugerido: Listas de cotejo para evaluar claridad, organización y corrección; rúbrica para trabajos grupales.

Evaluación sumativa

¿Qué se evalúa? Dominio integral para construir, comparar y corregir algoritmos secuenciales representados en diagramas de flujo y pseudocódigo.

¿Cómo se evalúa? Presentación individual o en parejas de un algoritmo resuelto en ambas representaciones, con explicación oral y entrega escrita.

Instrumento sugerido: Rúbrica que valore precisión, coherencia, claridad, corrección de errores y capacidad explicativa.

Unidad 4: Aplicación y análisis de algoritmos secuenciales

Objetivos de Aprendizaje

- Al finalizar la unidad, el estudiante será capaz de diseñar algoritmos secuenciales para resolver problemas cotidianos y académicos utilizando diagramas de flujo y pseudocódigo.
- Al finalizar la unidad, el estudiante será capaz de ejecutar y evaluar algoritmos secuenciales para identificar errores lógicos y de secuencia en un contexto dado.
- Al finalizar la unidad, el estudiante será capaz de corregir y optimizar algoritmos secuenciales para mejorar su eficiencia y claridad en la resolución de problemas específicos.
- Al finalizar la unidad, el estudiante será capaz de reflexionar críticamente sobre la eficacia de diferentes algoritmos secuenciales aplicados a un mismo problema, justificando sus mejoras.

Contenidos Temáticos

1. Introducción a la Aplicación de Algoritmos Secuenciales

- Concepto y importancia de los algoritmos secuenciales en la vida diaria y académica: Exploración de situaciones cotidianas donde se aplican procesos secuenciales.
- Revisión rápida de elementos básicos: repaso de instrucciones, secuencia, entrada y salida en algoritmos.

2. Diseño de Algoritmos Secuenciales

- Uso de diagramas de flujo para representar algoritmos: símbolos básicos, estructura y reglas para su elaboración.
- Elaboración de pseudocódigo: convenciones, sintaxis sencilla y ejemplos aplicados a problemas cotidianos.
- Práctica de diseño de algoritmos para problemas cotidianos y académicos: ejercicios guiados con enfoque práctico.

3. Ejecución y Evaluación de Algoritmos Secuenciales

- Ejecutar algoritmos paso a paso: simulación y seguimiento de procesos para entender la lógica.
- Identificación de errores lógicos y de secuencia: tipos comunes de errores y estrategias para detectarlos.
- Estudio de casos con errores para análisis y corrección.

4. Corrección y Optimización de Algoritmos Secuenciales

- Metodologías para corregir errores en algoritmos: técnicas de depuración y revisión sistemática.
- Optimización para mejorar eficiencia: simplificación de pasos, eliminación de redundancias y claridad en la presentación.
- Reescritura de algoritmos mejorados con diagramas de flujo y pseudocódigo.

5. Reflexión Crítica sobre Algoritmos Secuenciales

- Comparación de diferentes soluciones algorítmicas para un mismo problema.
- Análisis de ventajas y desventajas en términos de claridad, eficiencia y aplicabilidad.
- Justificación de mejoras y selección de la mejor solución propuesta.

Actividades

Actividad 1: Diseño de un algoritmo para una tarea cotidiana

Objetivo: Diseñar algoritmos secuenciales para resolver problemas cotidianos utilizando diagramas de flujo y pseudocódigo.

Descripción:

- Los estudiantes seleccionan una tarea diaria (por ejemplo, preparar un sándwich, ordenar útiles escolares, o calcular el cambio al comprar algo).
- En parejas, diseñan el algoritmo que resuelve la tarea usando primero un diagrama de flujo y luego el pseudocódigo.

- Presentan sus propuestas al grupo para discusión y retroalimentación.

Organización: Parejas

Producto esperado: Diagramas de flujo y pseudocódigos claros y completos para la tarea elegida.

Duración estimada: 90 minutos

Actividad 2: Ejecución y detección de errores en algoritmos dados

Objetivo: Ejecutar y evaluar algoritmos secuenciales para identificar errores lógicos y de secuencia.

Descripción:

- Se entregan a los estudiantes algoritmos con errores comunes (por ejemplo, pasos fuera de orden, cálculos incorrectos).
- Individualmente o en pequeños grupos, ejecutan el algoritmo paso a paso para detectar y anotar los errores encontrados.
- Discuten en plenaria las posibles causas y cómo corregirlos.

Organización: Individual o grupos de 3

Producto esperado: Informe de errores detectados con explicación y propuesta de corrección.

Duración estimada: 60 minutos

Actividad 3: Optimización y mejora de un algoritmo secuencial

Objetivo: Corregir y optimizar algoritmos para mejorar su eficiencia y claridad.

Descripción:

- Se presenta un algoritmo funcional pero poco eficiente o ambiguo.
- En grupos, revisan el algoritmo para identificar pasos mejorables.
- Rediseñan el algoritmo optimizado en diagrama de flujo y pseudocódigo.
- Comparan la versión original y la optimizada, destacando las mejoras.

Organización: Grupos de 3-4 estudiantes

Producto esperado: Algoritmo optimizado con documentación que explique las mejoras realizadas.

Duración estimada: 90 minutos

Actividad 4: Debate y reflexión crítica sobre diferentes algoritmos

Objetivo: Reflexionar críticamente sobre la eficacia de diferentes algoritmos aplicados a un mismo problema y justificar sus mejoras.

Descripción:

- Se presentan dos o más algoritmos que resuelven el mismo problema.
- En equipos, analizan cada algoritmo en términos de claridad, eficiencia y aplicabilidad.
- Preparan una argumentación para defender cuál consideran mejor y por qué.

- Realizan un debate guiado donde exponen sus conclusiones y reciben retroalimentación.

Organización: Equipos de 4 estudiantes

Producto esperado: Argumentos escritos y exposición oral con justificación crítica.

Duración estimada: 60 minutos

Evaluación

Evaluación Diagnóstica

Qué se evalúa: Conocimientos previos sobre algoritmos secuenciales, comprensión básica del concepto y habilidades iniciales para representar algoritmos.

Cómo se evalúa: Breve cuestionario escrito y un ejercicio práctico simple para diseñar un algoritmo básico (por ejemplo, pasos para lavarse las manos).

Instrumento sugerido: Cuestionario de opción múltiple y una actividad práctica en clase.

Evaluación Formativa

Qué se evalúa: Progreso en diseño, ejecución, corrección y optimización de algoritmos; participación en actividades; aplicación de conceptos.

Cómo se evalúa: Observación directa durante actividades, revisión de productos parciales (diagramas de flujo, pseudocódigos, informes de errores), autoevaluaciones y coevaluaciones.

Instrumento sugerido: Rúbricas para evaluación de diagramas y pseudocódigo, listas de cotejo para participación y corrección de errores, formatos de autoevaluación.

Evaluación Sumativa

Qué se evalúa: Capacidad para diseñar, ejecutar, evaluar, corregir y reflexionar críticamente sobre algoritmos secuenciales.

Cómo se evalúa: Proyecto final donde el estudiante debe presentar un algoritmo secuencial para un problema definido, incluyendo su diagrama de flujo, pseudocódigo, análisis de errores, versión optimizada y reflexión crítica comparativa.

Instrumento sugerido: Rúbrica de evaluación integral que considere calidad técnica, claridad, corrección, optimización y profundidad de la reflexión.