

Algoritmos y pseudocódigo

Tecnología e Informática | Informática

Descripción del Curso

Este curso de Informática está dirigido a estudiantes de 15 a 16 años y se estructura en varias unidades que desarrollan el razonamiento lógico, el diseño de algoritmos y la capacidad de comunicar ideas técnicas de forma clara. Enfocado en la resolución de problemas computacionales, el curso busca que el alumnado sea capaz de analizar, comparar y justificar diferentes soluciones algorítmicas y, así, seleccionar la opción más adecuada para un objetivo específico. Aunque cada unidad aborda aspectos distintos del diseño y la evaluación de algoritmos, todas comparten el propósito de fomentar la claridad en la expresión de ideas, la precisión en la verificación de soluciones y la capacidad de razonar críticamente ante distintas aproximaciones. En particular, la Unidad 4, titulada Evaluación y comparación de soluciones algorítmicas, se centra en comparar distintas soluciones para un mismo problema con criterios explícitos: claridad, corrección y simplicidad del pseudocódigo. Esta unidad complementa el aprendizaje previo al exigir a los estudiantes analizar al menos dos enfoques, evaluar sus ventajas y limitaciones y justificar, de forma documentada, la selección de la solución más adecuada. Las actividades combinan lectura de ejemplos, análisis guiado, diseño de soluciones alternativas, uso de rúbricas de evaluación y presentaciones que permitan comunicar razonadamente las decisiones tomadas. Al finalizar la unidad, el alumnado debe ser capaz de explicar por qué una solución es preferible a otras y de documentar el razonamiento detrás de su elección, aplicando estos principios tanto en contextos académicos como en situaciones prácticas, como mejoras de procesos o resolución de problemas cotidianos. El curso busca además desarrollar habilidades transversales: trabajo colaborativo, comunicación oral y escrita, pensamiento crítico y transferencia de conocimiento a contextos reales. Se contemplan prácticas en las que los estudiantes comparan enfoques, argumentan con evidencia y aprenden a justificar decisiones de diseño con criterios técnicos y conceptuales claros. En resumen, el curso ofrece una base sólida en diseño, evaluación y documentación de algoritmos, preparando a los estudiantes para enfrentar problemas computacionales con rigor, claridad y responsabilidad.

Competencias

- Analizar y comparar soluciones algorítmicas para un mismo problema, identificando ventajas y limitaciones de cada enfoque.
- Explicar con claridad los criterios de evaluación (claridad, corrección y simplicidad) y justificar la elección de la solución más adecuada.
- Aplicar principios de diseño de algoritmos y razonar de forma objetiva, documentando el razonamiento detrás de cada decisión.
- Comunicar ideas técnicas de forma oral y escrita, y colaborar para analizar alternativas y compartir resultados.
- Transferir el pensamiento algorítmico a contextos reales y a situaciones prácticas, considerando aspectos de eficiencia, legibilidad y mantenibilidad.

Requerimientos

- Conocimientos básicos de lógica, estructuras de control y pseudocódigo.
- Familiaridad con la lectura y escritura de pseudocódigo o lenguajes de programación simples.
- Acceso a una computadora con herramientas de edición de texto y conexión a Internet para consultar recursos y documentar razonamientos.
- Disponibilidad de tiempo para realizar análisis comparativos, trabajar en equipo y preparar presentaciones breves de resultados.
- Actitud de pensamiento crítico, colaboración y capacidad para justificar decisiones de diseño.

Unidades del Curso

Unidad 1: Unidad 1: Introducción a Algoritmos y resolución de problemas

Objetivos de Aprendizaje

- Identificar problemas simples que pueden resolverse mediante un algoritmo.
- Describir, en lenguaje natural, el proceso para resolver un problema y convertirlo en un esquema básico de pseudocódigo.
- Distinguir entre entradas, procesos y salidas en un procedimiento sencillo.

Contenidos Temáticos

1. **Tema 1:** Qué es un algoritmo y su relación con la resolución de problemas. Descripción de un algoritmo como un conjunto de pasos finitos que conducen a una solución.
2. **Tema 2:** Del lenguaje natural al pseudocódigo. Cómo convertir una descripción verbal en un esquema lógico básico.
3. **Tema 3:** Identificación de entradas, procesos y salidas. Componentes de un procedimiento sencillo.

Actividades

- **Actividad 1: Exploración de un problema cotidiano** – En parejas, identificar un problema de la vida diaria y describirlo en lenguaje natural y en pseudocódigo muy básico. Puntos clave: claridad de pasos, terminología, entradas y salidas. Aprendizajes: comprensión de qué es un algoritmo y cómo se describe.
- **Actividad 2: Conversión a pseudocódigo** – Tomar un proceso simple (por ejemplo, hacer una taza de té) y escribir su pseudocódigo con entradas, salidas y pasos. Aprendizajes: estructura de un pseudocódigo básico.
- **Actividad 3: Debate sobre lenguajes** – Comparar lenguaje natural y pseudocódigo, discutir cuándo usar cada uno y por qué. Aprendizajes: diferencias, utilidad de la abstracción.

Evaluación

Se evaluará la comprensión de la relación entre algoritmo y resolución de problemas, la capacidad de describir procesos en lenguaje natural y en pseudocódigo, y la identificación de entradas, procesos y salidas.

- Prueba corta de conceptos: definición de algoritmo y su relación con la resolución de problemas.
- Ejercicio práctico: convertir un proceso en lenguaje natural a pseudocódigo básico.
- Participación y calidad de las descripciones en las actividades de clase.

Unidad 2: Unidad 2: Construcción de pseudocódigo para problemas simples

Objetivos de Aprendizaje

- Identificar y describir las entradas, salidas y el proceso en problemas simples.
- Traducir descripciones de problemas en pseudocódigo básico.
- Emplear estructuras básicas de control y secuencias en pseudocódigo para resolver tareas simples.

Contenidos Temáticos

1. **Tema 1:** Definiciones de entradas, salidas y procesos. Cómo distinguir cada componente en un problema.
2. **Tema 2:** Estructuras básicas de pseudocódigo: asignaciones, lectura, escritura y secuencias simples.
3. **Tema 3:** Escribir pseudocódigo para problemas simples con ejemplos prácticos.

Actividades

- **Actividad 1: Esquemas simples** – Identificar entradas, salidas y procesos en problemas cotidianos y escribir pseudocódigo para resolverlos (p. ej., calcular el área de un rectángulo). Puntos clave: entradas/outputs, pasos lógicos.
- **Actividad 2: Pseudocódigo básico** – Escribir pseudocódigo para calcular la suma de dos números y para obtener el promedio de dos números, especificando entradas y salidas.
- **Actividad 3: Revisión entre pares** – Intercambiar pseudocódigos entre compañeros para identificar mejoras en claridad y estructura.
- **Actividad 4: Mini taller de escritura** – Resolver un conjunto de problemas simples mediante pseudocódigo y discutir las decisiones de diseño.

Evaluación

Se evaluará la capacidad para identificar entradas, salidas y procesos, y la habilidad para redactar pseudocódigo correcto y claro para problemas simples.

- Rubrica de escritura de pseudocódigo: claridad, corrección y completitud (entradas, salidas y procesos).
- Ejercicios prácticos individuales: convertir descripciones en pseudocódigo básico.

Unidad 3: Unidad 3: Redacción de soluciones en pseudocódigo para problemas dados

Objetivos de Aprendizaje

- Generar soluciones en pseudocódigo para problemas dados, declarando entradas, salidas y procesos.
- Aplicar estructuras de control: secuencias, condicionales y bucles en pseudocódigo.
- Mantener claridad y corrección en el pseudocódigo, seleccionando estructuras adecuadas según el problema.

Contenidos Temáticos

1. **Tema 1:** Estructuras de control en pseudocódigo: secuencias y condicionales. Cómo organizar pasos y tomar decisiones.
2. **Tema 2:** Bucles y repeticiones para automatizar tareas simples. Conceptos de iteración y condiciones de terminación.
3. **Tema 3:** Organización de entradas, procesos y salidas en soluciones completas. Legibilidad y estructura.

Actividades

- **Actividad 1: Cálculo del factorial** – Escribe pseudocódigo que calcule el factorial de un número usando un bucle, definiendo entradas y salidas. Aprendizajes: uso correcto de bucles para iterar y acumular resultados.
- **Actividad 2: Año bisiesto** – Escribe pseudocódigo que determine si un año es bisiesto usando estructuras condicionales. Aprendizajes: evaluación de condiciones y ramificación.
- **Actividad 3: Máximos en una lista** – Dado una lista de números, escribe pseudocódigo para encontrar el mayor valor usando un bucle. Aprendizajes: manejo de colecciones y variables de control.
- **Actividad 4: Revisión entre pares** – Intercambiar soluciones pseudocódigo con un compañero y proponer mejoras para claridad y corrección.

Evaluación

Se evaluará la capacidad de redactar soluciones en pseudocódigo para problemas dados, empleando estructuras adecuadas y manteniendo claridad y corrección.

- Ejercicios prácticos: escritura de pseudocódigo para tres problemas dados, con entradas, procesos y salidas bien identificados.
- Rúbrica de estructura: uso correcto de secuencias, condicionales y bucles; legibilidad y consistencia de la nomenclatura.

Unidad 4: Evaluación y comparación de soluciones algorítmicas

Objetivos de Aprendizaje

- Analizar dos o más enfoques para resolver el mismo problema.
- Comparar en función de claridad, corrección y simplicidad del pseudocódigo.
- Justificar la selección de la solución más adecuada y documentar el razonamiento detrás de la elección.

Contenidos Temáticos

1. **Tema 1:** Criterios para comparar soluciones. Definir qué hace que una solución sea clara y correcta.
2. **Tema 2:** Análisis de claridad, corrección y simplicidad. Cómo evaluar cada criterio y registrar resultados.
3. **Tema 3:** Presentación y defensa de la mejor solución. Cómo justificar elecciones y comunicar argumentos de manera razonable.

Actividades

- **Actividad 1: Dos enfoques, un problema** – Resolver el mismo problema con dos o tres enfoques en pseudocódigo y compararlos. Aprendizajes: identificar trade-offs entre soluciones.
- **Actividad 2: Rúbrica de evaluación** – Completar una rúbrica para evaluar las soluciones presentadas y registrar hallazgos.
- **Actividad 3: Presentación y defensa** – Presentar la mejor solución ante la clase y defender la elección con argumentos claros y sólidos. Aprendizajes: comunicación y justificación técnica.
- **Actividad 4: Retroalimentación entre pares** – Realizar feedback constructivo sobre las soluciones vistas y destrezas de razonamiento.

Evaluación

Se evaluará la capacidad de comparar y justificar la elección de una solución entre varias opciones, valorando claridad, corrección y simplicidad del pseudocódigo.

- Proyecto de comparación de soluciones: entrega de un informe con análisis y elección justificada.
- Presentación oral de la solución elegida, con defensa de argumentos y respuestas a preguntas.