

Pensamiento computacional e inteligencia artificial

Luis Miguel Iglesias Albarrán

SERVICIO DE PUBLICACIONES DE LA FEDERACIÓN ESPAÑOLA DE SOCIEDADES DE PROFESORES DE MATEMÁTICAS

Avda. de la Mancha, s/n. (IES Universidad Laboral)

02006 Albacete

publicaciones@fespm.es

<http://www.fespm.es>

@ De esta edición

FEDERACIÓN ESPAÑOLA DE SOCIEDADES DE PROFESORES DE MATEMÁTICAS FESPM

ISBN: 978-84-122154-9-6

España, 2025

Aprender matemáticas es mucho más que hacer operaciones. Es una forma de mirar el mundo, de pensar con lógica y de buscar soluciones. Vivimos rodeados de datos, de algoritmos y de máquinas que aprenden. Usamos asistentes virtuales que responden a nuestras preguntas, programas que traducen textos o calculadoras que no solo hacen cuentas sino que también dibujan gráficos y corrigen errores. Detrás de todo esto hay matemáticas que analizan, modelan y predicen. Matemáticas que piensan.

Este cuadernillo pretende acercarnos a dos ideas clave del presente y futuro: el pensamiento computacional y la inteligencia artificial. No para convertirnos en programadores, sino para usar estas formas de pensar y así aprender matemáticas con más sentido y de manera más significativa.

El pensamiento computacional enseña a organizar ideas, dividir problemas difíciles en partes más sencillas, buscar patrones y crear algoritmos, que son pasos ordenados para resolver un problema. La inteligencia artificial invita a pensar cómo aprenden las máquinas, toman decisiones y mejoran con la experiencia, pero para entenderlas primero hay que saber cómo piensa una persona al resolver problemas.

Aquí se trabajará con un modelo que une tres ideas:

- Resolver un problema que motive y haga pensar.
- Usar el pensamiento computacional para organizar y buscar soluciones.
- Compartir y dialogar en un Círculo Matemático Computacional para aprender en equipo.

Este proceso no solo permite mejorar las habilidades matemáticas, sino también la capacidad para explicar ideas, razonar, colaborar y pensar críticamente. Se aprenderá a explicar por qué se hace cada cosa y cómo mejorar. Se encontrarán actividades, retos y juegos para observar, preguntar, probar, representar y decidir usando herramientas digitales que se presentarán más adelante, como LearningML, Scratch y simuladores. Así, las matemáticas dejan de ser ejercicios sueltos y se convierten en un lenguaje para entender y transformar el mundo. Porque las matemáticas que piensan no solo buscan respuestas correctas, sino que enseñan a razonar bien, comunicar con claridad y actuar con responsabilidad. Y hoy, aprender matemáticas también significa aprender a convivir con las máquinas..., sin dejar de ser humanos.

1. ¿Cómo se aprende resolviendo problemas?

La resolución de problemas como motor del pensamiento y el aprendizaje

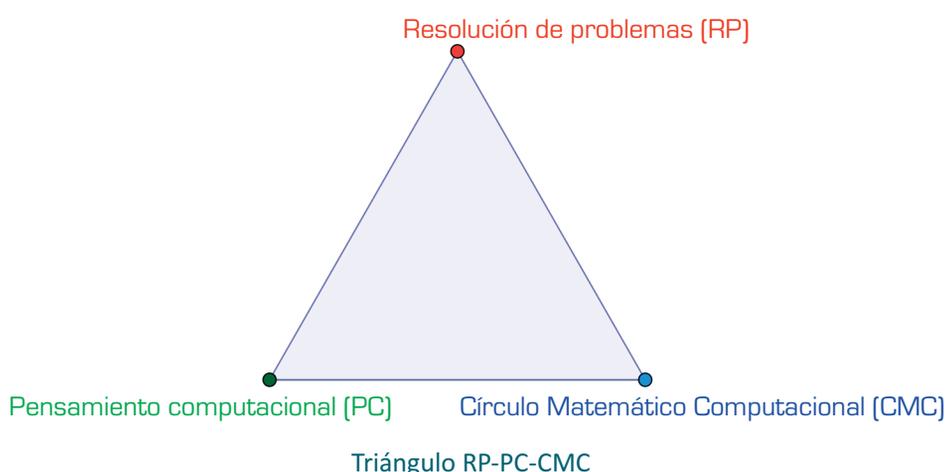
Resolver un problema no es usar un truco sino aprender a mirar las cosas de otra manera. Pensar computacionalmente es aprender a mirar con lógica, con orden y también con imaginación. Aprender matemáticas es mucho más que seguir reglas o hacer cálculos rutinarios. Se trata de entender situaciones, tomar decisiones con sentido, buscar conexiones y construir un significado propio. Cuando uno se enfrenta a un problema interesante y abierto, se pone en marcha algo más profundo que cuando únicamente se hacen cuentas: empieza el pensamiento. En ese momento, no solo se usa lo que ya se sabe, sino que también se descubre lo que falta por aprender.

Resolver problemas es mucho más que encontrar una solución numérica. Es hacer preguntas, entender bien lo que se pide, organizar la información, hacer hipótesis, probar diferentes caminos, comprobar si funcionan y aprender de los errores. Aunque no siempre sigas un orden fijo, hay una lógica detrás: una serie de pasos que puedes analizar, explicar y mejorar. Aquí es donde el pensamiento computacional ayuda mucho.

Pensar computacionalmente significa pensar de forma organizada, ordenada y con un propósito. Es dividir un problema complicado en partes más pequeñas, reconocer patrones, enfocarte en lo importante y diseñar un algoritmo, que es una secuencia clara de pasos para llegar a una solución. El pensamiento computacional no está separado de las matemáticas; es una manera de hacerlas visibles, entendibles y comunicables.

En este cuadernillo trabajamos con un modelo que une tres ideas clave:

- RP (Resolución de Problemas). Se empieza con un reto motivador y que dé sentido a lo que se está aprendiendo.
- PC (Pensamiento Computacional). Más adelante usar estrategias como secuencias, condicionales, bucles, tablas o pseudocódigo para organizar nuestras ideas.
- CMC (Círculo Matemático Computacional). Finalmente convertir el aula en un espacio para dialogar, compartir ideas, generalizar soluciones, hacer nuevas preguntas y aprender juntos.



Un Círculo Matemático Computacional no es una clase normal ni un conjunto de ejercicios para hacer de manera individual. Es una forma diferente de aprender matemáticas: más abierta, más comunicativa y más conectada con el pensamiento real. En ellos, se trabaja en grupo, compartiendo

dudas, probando ideas distintas y discutiendo razonamientos. En un Círculo Matemático Computacional no es importante únicamente el resultado sino la conversación matemática que surge, el aprendizaje que subyace del intercambio de ideas y la reflexión que nos hace ver cosas nuevas.

Por ejemplo, si durante el transcurso de una clase pedimos que se elija cuál de varios depósitos se llenará primero, surgirán ideas como comparar cantidades, representar la situación con gráficos o tablas, estimar y razonar proporcionalmente. Si luego se tiene que programar una aplicación para simularlo, escribir las instrucciones paso a paso o crear un algoritmo para automatizar la decisión, estaremos usando pensamiento computacional y reforzando el sentido matemático. Y si después se explican y mejoran los trabajos con los compañeros, el aprendizaje es mucho más profundo y significativo. Este modo de trabajar no solo mejora las habilidades matemáticas sino que también nos acerca a cómo las máquinas «aprenden» a resolver problemas. Al entender cómo organizamos nuestras soluciones, nos será más fácil comprender cómo una inteligencia artificial crea las suyas, usando datos, reglas y simulaciones. Así, la tecnología no reemplaza al pensamiento humano: lo provoca, lo desafía y lo complementa.

Porque aprender matemáticas resolviendo problemas y usando el pensamiento computacional no es solo aprender más matemáticas. Es aprender a pensar para la vida. Ayuda a entender mejor el mundo, a enfrentarse a él con estrategias y a actuar con sentido. Las matemáticas dejan de ser solo ejercicios rutinarios aislados descontextualizados y se convierten en una herramienta para pensar mejor.

2. Pensar como matemáticos... y también como máquinas

2.1 ¿Qué es el pensamiento computacional?

El pensamiento computacional no es hacer lo que haría una máquina. Es aprender a pensar como un matemático ordenado, curioso y creativo. Esta forma de pensar nos ayuda a resolver problemas difíciles con orden, lógica y claridad. Aunque el término viene del mundo de la informática, no solo sirve para usar ordenadores. Como iremos viendo, es una manera de razonar, útil y práctica, especialmente valiosa para aprender matemáticas.

Pensar computacionalmente significa descomponer problemas grandes en partes más pequeñas, encontrar lo importante, buscar patrones, generalizar ideas y crear procedimientos o algoritmos que podamos usar en muchas situaciones. No es pensar como una máquina, sino pensar para di-



Esquema visual con las cuatro dimensiones del pensamiento computacional (CATEDU)

señalar soluciones claras y reutilizables. Este tipo de pensamiento aparece cuando uno se enfrenta a un problema que no es rutinario. Por ejemplo, si tienes que organizar turnos en clase sin repetir combinaciones, analizas, pruebas, corriges, simplificas y mejoras. Eso es pensar como un programador, aunque sin programar. Lo bueno de este enfoque es que se puede trabajar con papel y lápiz, juegos, lenguaje sencillo o con bloques visuales como Scratch. Lo importante no es la herramienta, sino aprender a pensar paso a paso, prever errores, planificar y explicar bien lo que haces.

En matemáticas, el pensamiento computacional está relacionado con secuencias, funciones, proporciones, coordenadas, lógica, geometría, probabilidad... Además, ayuda a organizar ideas, comunicarte mejor y evaluar tus estrategias. En resumen, es pensar sobre cómo piensas a la hora de hacer matemáticas.

Incorporar esta forma de pensar nos prepara también para entender cómo piensan las máquinas. Si se sabe crear un algoritmo para resolver un problema, se podrá comprender cómo una inteligencia artificial aprende a hacerlo a partir de datos y ejemplos. Esta conexión es muy importante porque conecta el pensamiento humano con la lógica de la inteligencia artificial. El pensamiento computacional no es una moda ni un complemento. Es una manera actual, potente y útil, de aprender matemáticas y de aprender a pensar con rigor, de manera razonada y lógica, y con sentido.

2.2 ¿Qué es un algoritmo y para qué sirve? Expresión de un algoritmo

Un algoritmo es una secuencia ordenada de pasos que seguimos para resolver un problema o hacer una tarea. Aunque la palabra no suene y no se esté familiarizado con ella, lo cierto es que usamos algoritmos cada día, casi sin darnos cuenta: al preparar una receta, montar un mueble, resolver una ecuación o encontrar el camino más corto en un mapa. En matemáticas, los algoritmos son muy importantes. Nos ayudan a resolver operaciones sencillas y combinadas, ecuaciones, calcular raíces cuadradas o decidir si un número es primo. Los algoritmos nos ayudan a organizar el pensamiento, a comprobar la validez de nuestras soluciones y a explicar claramente cómo las hemos encontrado.

Un buen algoritmo debe ser claro, eficiente, fácil de entender y reutilizable. Por eso, crear algoritmos en clase no es solo resolver bien, sino aprender a pensar con orden y precisión. No hace falta saber programar para trabajar con algoritmos. Podemos escribirlos en pseudocódigo (un lenguaje parecido al nuestro), usar diagramas de flujo (dibujos), o programarlos en el lenguaje por bloques que usa Scratch (este u otro lenguaje de programación, como Python, entre otros). Lo importante es aprender a describir paso a paso lo que hacemos, con una lógica clara y pensando en todas las posibilidades: ¿qué pasa si un número es negativo? ¿Cuándo debe terminar el proceso? ¿Cómo corregir errores?

Por ejemplo, diseñar un algoritmo (para saber si un número es par o impar, para sumar dos números, para resolver una ecuación de segundo grado, para clasificar triángulos según sus lados o para calcular una media aritmética) implica usar matemáticas, lógica y aprender a comunicarnos de forma ordenada. Si compartimos, probamos y mejoramos estos algoritmos, el aprendizaje es mucho más profundo.

Además, entender y construir algoritmos nos prepara para entender la inteligencia artificial. Detrás de cada decisión automática hay uno o varios algoritmos. Así, al aprender a crear algoritmos simples, también aprendemos cómo una inteligencia artificial toma decisiones basadas en datos y reglas. Aprender a construir algoritmos en matemáticas es aprender a pensar con rigor, resolver

problemas con claridad y comunicar con lógica. Un buen algoritmo no solo resuelve, sino explica, organiza y nos ayuda a pensar mejor.

A continuación mostramos, mediante un ejemplo, cómo expresar un algoritmo de tres formas:

- Pseudocódigo.
- Diagrama de flujo.
- Programado mediante el lenguaje de bloques Scratch.

Actividad 1

Diseñar un algoritmo para sumar dos números enteros.
Presentarlo de forma sencilla para que se pueda entender y usar.

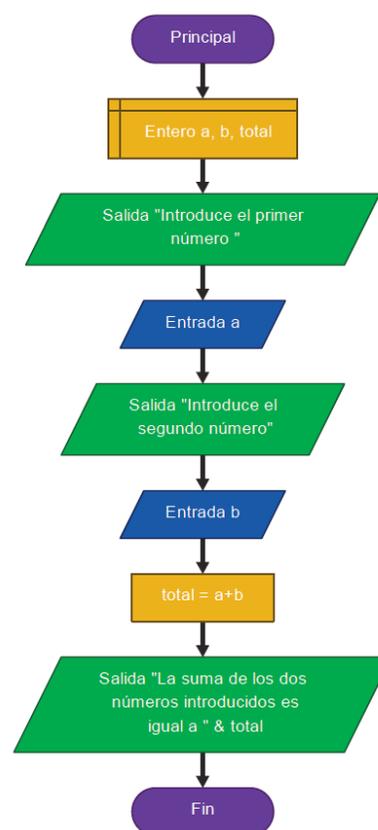
Veamos su resolución de la tres formas:

1. Escrita en pseudocódigo:

```
Función Principal
  Declarar Entero a, b, total

  Salida "Introduce el primer número "
  Entrada a
  Salida "Introduce el segundo número"
  Entrada b
  Asignar total = a+b
  Salida "La suma de los dos números introducidos es igual a " & total
Fin
```

2. Representada con un diagrama de flujo:



3. Programada en el lenguaje de programación por bloques Scratch:



La imagen de la izquierda muestra la pantalla del algoritmo programado en Scratch por Luis Miguel Iglesias. Se puede probar en este enlace <<https://scratch.mit.edu/projects/845191887>>.

2.3 ¿Cómo se relaciona con las matemáticas?

El pensamiento computacional no es un contenido ajeno a las matemáticas, ni una moda pasajera importada desde la informática. Es, en realidad, una manera natural de pensar que ya está presente en muchas de las actividades matemáticas que hacemos en el aula. Lo que aporta es una forma de hacerlas más visibles, más estructuradas y más comunicables.

Cuando se descompone una figura para calcular su área, se clasifican números según una propiedad, se generaliza una sucesión numérica o se explica paso a paso cómo se ha resuelto una ecuación, se está aplicando pensamiento computacional. Lo hacemos sin necesidad de un ordenador: estamos descomponiendo, abstrayendo, detectando patrones y construyendo algoritmos, que son los pilares de este tipo de razonamiento.

Esta conexión se refuerza aún más cuando trabajamos con contextos de simulación, representación o toma de decisiones. Por ejemplo, si se entrena a una inteligencia artificial sencilla para clasificar números como primos o no primos, se está aplicando conocimientos de divisibilidad, razonamiento lógico, uso de variables y construcción de criterios. Estamos usando matemáticas para que una máquina «entienda» una propiedad numérica.

Del mismo modo, si estamos diseñando una tabla de valores para representar un proceso que cambia en el tiempo, como por ejemplo la temperatura de una bebida que se enfría o el crecimiento de una planta, se está desarrollando pensamiento computacional y trabajando saberes básicos relacionados con la medida, funciones y gráficas, proporcionalidad...

El pensamiento computacional nos invita a ver las matemáticas como procesos, no únicamente como productos. Nos permite analizar cómo se llega a una solución, no solo cuál es el resultado. Nos enseña a pensar en términos de entrada, proceso y salida; de hipótesis, prueba y revisión; de decisiones condicionales y de repetición controlada.

Además, trabajar con pensamiento computacional fortalece muchas de las competencias específicas del currículo de matemáticas: ayuda a formular conjeturas, a justificar razonamientos, a representar con claridad, a comunicar estrategias, a utilizar herramientas tecnológicas con criterio, y a reflexionar sobre lo aprendido.

En resumen, el pensamiento computacional no compite con las matemáticas, sino que las acompaña y las mejora. Les aporta orden, estructura, claridad y estrategia. Y, al mismo tiempo, las matemáticas proporcionan al pensamiento computacional el contenido, el lenguaje y el rigor necesarios para construir soluciones con sentido.

2.4 Pensamiento computacional y resolución de problemas

A medida que la tecnología desempeña un papel cada vez más relevante en nuestras vidas, ya no a largo plazo, sino a medio o corto plazo, la enseñanza y el aprendizaje de las matemáticas, encaminada a la alfabetización matemática de toda la ciudadanía, debería también englobar la relación recíproca y sinérgica entre el pensamiento matemático y el pensamiento computacional. Wing (2006) definió este último como «la manera en que piensan los científicos de las ciencias de la computación», visto como «un proceso de pensamiento que supone formular problemas y diseñar sus soluciones de una manera que puedan ser ejecutadas por un ordenador, un humano o una combinación de ambos» (Wing, 2011).

Las funciones que cumple el pensamiento computacional en las matemáticas incluyen la manera en que determinados saberes matemáticos específicos interactúan con conceptos computacionales concretos y sobre cómo el razonamiento matemático complementa el pensamiento computacional. El pensamiento computacional es, en esencia, un enfoque de resolución de problemas que puede ser utilizado por todo el mundo, en una amplia variedad de áreas de contenido y contextos. De manera resumida, podemos presentarlo como un enfoque en el que se descomponen los problemas en partes diferenciadas, se buscan similitudes/analogías con otros, se identifica la información relevante y las oportunidades de simplificación, y se diseña/crea un plan para su solución. Este amplio marco de resolución de problemas incluye cuatro elementos: descomposición, reconocimiento de patrones, abstracción y algoritmos.

Según la Sociedad Internacional para la Tecnología en la Educación (ISTE), «el pensamiento computacional no es más que un proceso de resolución de problemas» que incluye las siguientes características:

- Formular problemas de forma que se permita el uso de un ordenador y otras herramientas para ayudar a resolverlos.
- Organizar y analizar lógicamente la información.
- Representar la información a través de abstracciones como los modelos y las simulaciones.
- Automatizar soluciones haciendo uso del pensamiento algorítmico (estableciendo una serie de pasos ordenados para llegar a la solución).
- Identificar, analizar e implementar posibles soluciones con el objetivo de lograr la combinación más efectiva y eficiente de pasos y recursos.
- Generalizar y transferir este proceso de resolución de problemas para ser capaz de resolver una gran variedad de familias de problemas.

Las habilidades de pensamiento computacional incluyen el reconocimiento de patrones, el diseño y uso de abstracciones, la descomposición de patrones, determinar qué herramientas de cálculo pueden ser empleadas para analizar o solucionar problemas y definir algoritmos como parte de una solución detallada. A continuación se presentan algunas preguntas facilitadoras para cada una de ellas:

<i>Descomposición</i>
¿Cuáles son las diferentes partes del problema que intentas resolver? Describe las secciones/partes principales del problema que estás resolviendo. ¿Cómo podría dividirse este problema en partes más pequeñas?
<i>Reconocimiento de patrones</i>
¿Observas algún patrón? ¿Notas alguna similitud entre este problema y otro que ya hayas resuelto? ¿Alguna de las partes de este problema comparten algunas características? ¿Hay algo que se repita?
<i>Abstracción</i>
¿Qué intentas resolver? ¿Qué detalles son importantes para resolver este problema? ¿Qué puedes omitir? ¿Qué información es innecesaria? ¿Puedes describir este problema como algo más básico, de manera más sencilla?
<i>Algoritmos</i>
¿Cuál es el primer paso que puedes dar para resolver este problema? ¿Cuáles son los pasos que tienes que dar para resolver este problema? ¿En qué orden debes completar estos pasos?

2.5 Resolvemos problemas con ayuda del pensamiento computacional

Resolver un problema empieza por entenderlo bien. Un ordenador no puede hacer magia si no le damos instrucciones claras. Lo mismo pasa con nosotros: antes de resolver, hay que pensar.

Cuando aplicamos pensamiento computacional para resolver un problema matemático, podemos seguir una secuencia parecida a la siguiente.

1. Leer y comprender el problema:
 - Leer el problema varias veces.
 - Establecer los datos del problema.
 - Aclarar lo que se va a resolver (¿cuál es la pregunta?).
 - Precisar el resultado que se desea lograr.
 - Determinar la o las incógnitas del problema.
 - Organizar la información.
 - Agrupar los datos en categorías.
 - Realizar una representación (figura, diagrama...).

2. Diseñar un plan:
 - Escoger y decidir las operaciones que efectuar.
 - Eliminar los datos superfluos, simplificar el problema, quedándonos únicamente con la información relevante y útil.
 - Descomponer el problema en otros más pequeños.
3. Ejecutar el plan. Resolver el problema:
 - Ejecutar en detalle cada operación.
 - Simplificar antes de calcular.
 - Realizar una representación que nos ayude a visualizar el plan.
4. Analizar la validez de la solución:
 - Dar una respuesta completa.
 - Hallar el mismo resultado de otra manera.
 - Verificar por apreciación que la respuesta es adecuada.
5. Diseñar el algoritmo:
 - Cuando se ha realizado un análisis a fondo del problema, se puede proceder a elaborar el algoritmo (diagrama de flujo). Este consiste en la representación gráfica, mediante símbolos geométricos, de la secuencia lógica de las instrucciones (plan), que, posteriormente, serán traducidas a pseudocódigo, y de ahí a un lenguaje de programación, como por ejemplo Scratch, para ejecutarlas y probarlas en un ordenador.

Veamos a continuación todo el proceso mediante el desarrollo completo de un ejemplo resuelto.

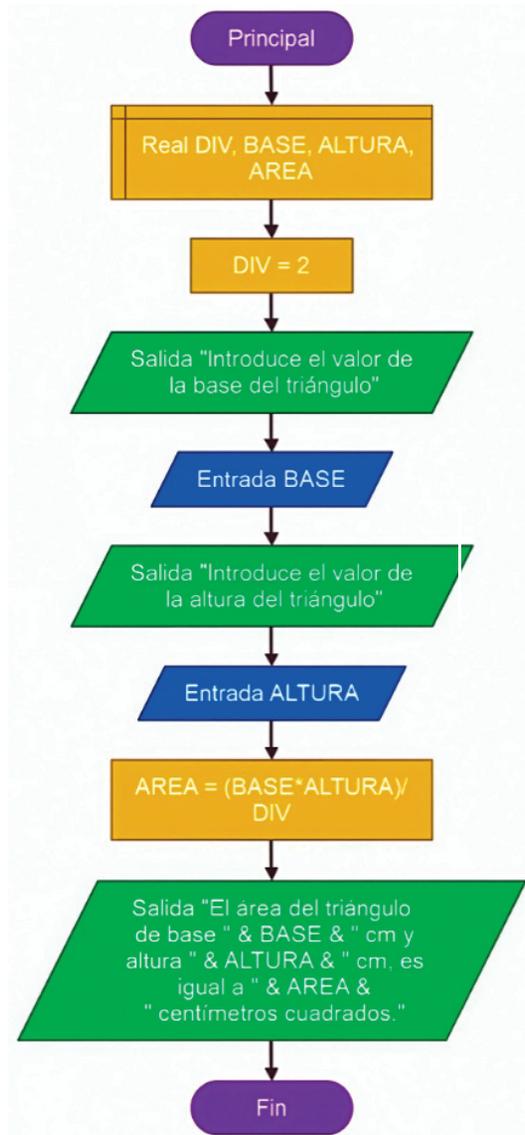
Actividad 2

Diseña un algoritmo (pseudocódigo y diagrama de flujo) para hallar el área de un triángulo rectángulo cuya base mide 3 cm, su altura mide 4 cm y su hipotenusa mide 5 cm.

1. Análisis del problema:

- Formular el problema.
Ya se encuentra claramente planteado.
- Resultados esperados.
El área de un triángulo rectángulo.
- Datos disponibles.
Base, altura, hipotenusa y tipo de triángulo. La incógnita es el área y todos los valores son constantes. El valor de la hipotenusa se puede obviar, no es necesario. El estudiante debe preguntarse si sus conocimientos actuales de matemáticas le permiten resolver este problema. De no ser así, debe plantear una estrategia para obtener, previamente, los conocimientos requeridos.
- Determinar las restricciones.
Utilizar las medidas dadas.
- Pasos necesarios.
Guardar en dos variables (BASE y ALTURA) los valores correspondientes, respectivamente, a la base y la altura dadas en el enunciado.
Guardar en una constante (DIV) el valor 2.
Aplicar la fórmula $(BASE * ALTURA) / DIV$ y guardar el resultado en la variable AREA.
Comunicar el resultado (AREA), mostrarlo en pantalla.

2. Diagrama de flujo:



3. Algoritmo en pseudocódigo:

- Paso 1: Inicio
- Paso 2: Asignar el número 2 a la constante "DIV"
- Paso 3: Asignar el número 3 a la constante "BASE"
- Paso 4: Asignar el número 4 a la constante "ALTURA"
- Paso 5: Guardar en la variable "AREA" el resultado de (BASE*ALTURA)/DIV
- Paso 6: Imprimir, mostrar en pantalla, el valor de la variable "AREA"
- Paso 7: Fin

Función Principal

Declarar Real DIV, BASE, ALTURA, AREA

Asignar DIV = 2

Salida "Introduce el valor de la base del triángulo"

Entrada BASE

Salida "Introduce el valor de la altura del triángulo"

Entrada ALTURA

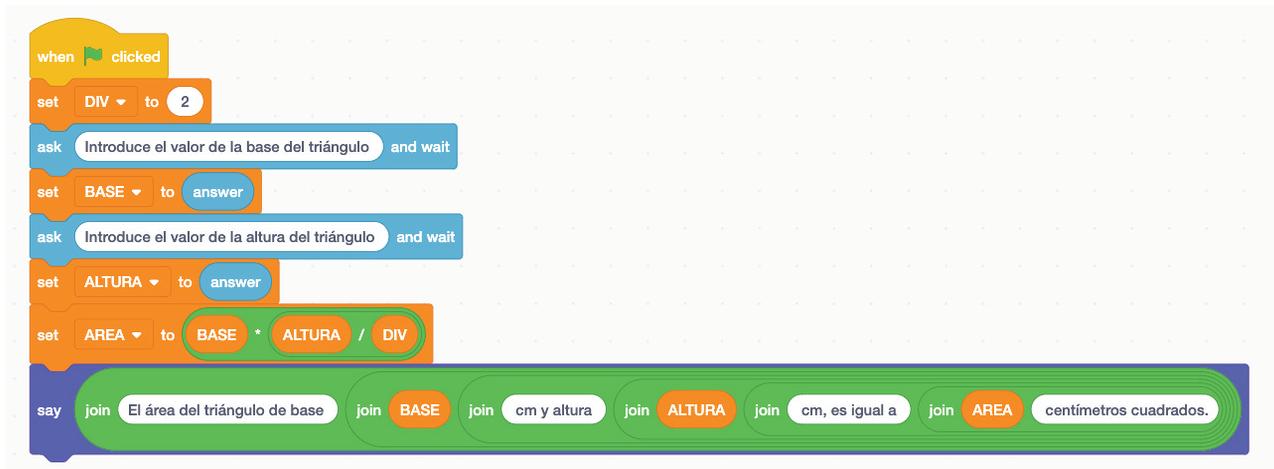
Asignar AREA = (BASE * ALTURA)/DIV

Salida "El área del triángulo de base " & BASE & " cm y altura " & ALTURA & " cm, es igual a " & AREA & " cm²"

Fin

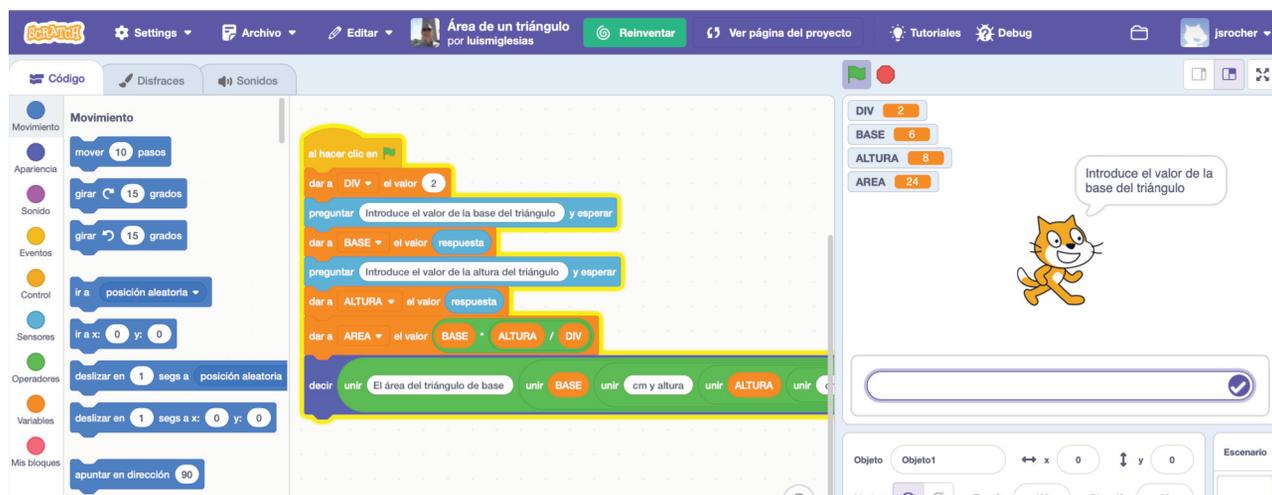
4. Traducir el algoritmos. Ejecutar el plan:

Una vez que el algoritmo está diseñado y representado gráficamente, se pasa a la etapa de traducción a un lenguaje de programación determinado (en nuestro caso será un lenguaje por bloques, Scratch). Cada lenguaje posee sus propias reglas gramaticales; por lo tanto, es fundamental que se conozca de antemano la sintaxis de los comandos que se deben utilizar para resolver el problema. A mayor dominio del lenguaje de programación, mayor posibilidad de llegar rápidamente a una solución satisfactoria. A esta fase de traducción se le conoce comúnmente como «codificación».



5. Depurar el programa. Revisar:

Después de traducir el algoritmo a un lenguaje de programación por bloques como Scratch, el programa resultante debe ser probado para validar su resultado. A este proceso se le conoce como «depuración». Depurar programas contribuye a mejorar la capacidad, como alumnos, de resolver problemas. La depuración es una destreza importante.



6. Representación del algoritmo(diagrama de flujo y pseudocódigo) y ejecución del programa (bloques de Scratch)



Representación de algoritmos: Área de un triángulo · Diagrama de flujo y pseudocódigo - Flowgorithm (Iglesias, 2023).
<<https://www.youtube.com/watch?v=xn0DnRV8AU4>>



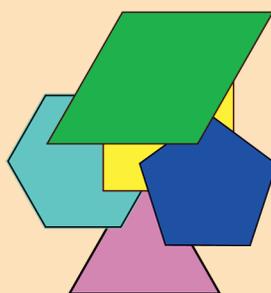
Programa sencillo con bloques Scratch: Área de un triángulo (Perfil Scratch de L. M. Iglesias)
<<https://scratch.mit.edu/projects/852842194/>>

2.6 Propuesta de actividades matemáticas para pensar y resolver con ayuda del pensamiento computacional

Actividad 3: Ordenando polígonos. ¿El orden importa?



Hipatia recortó los cinco polígonos anteriores y los colocó sobre una mesa. A continuación tomó una foto desde arriba:



¿En qué orden colocó los polígonos sobre la mesa?

Como se puede comprobar, se trata de resolver un pequeño rompecabezas sobre polígonos. ¡A disfrutar!

Actividad 3: Ampliación

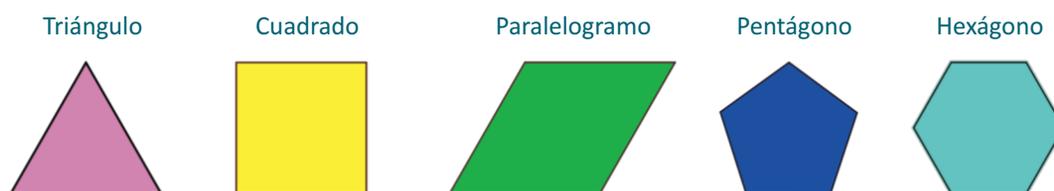
Justifica de manera razonada la respuesta a las siguientes preguntas:

1. ¿Crees que es importante el orden en que se hayan colocado los polígonos o piensas que no afecta?
2. ¿La elaboración de una receta de cocina tendría algo que ver con este problema?
3. ¿Podrías poner un ejemplo de la vida cotidiana, de tu día a día, donde se puede aplicar algo parecido?
4. ¿Serías capaz de dar un par de ejemplos del ámbito de las matemáticas donde haya que seguir una secuencia ordenada de acciones?
5. Coloca las figuras en el orden que consideres y reta a tus compañeros a que averigüen el orden en el que las colocaste.

Para saber más:

6. Busca en un diccionario o en internet la palabra «algoritmo».
7. Visita la página [Algoritmo, ¿qué es eso?](https://descargas.intef.es/recursos_educativos/ODES_SGOA/ESO/Matematicas/6C1_SA_Circulo_Matematico_Computacional/algoritmo_qu_es_eso.html) (<https://descargas.intef.es/recursos_educativos/ODES_SGOA/ESO/Matematicas/6C1_SA_Circulo_Matematico_Computacional/algoritmo_qu_es_eso.html>), léela con atención y realiza las actividades que se proponen.

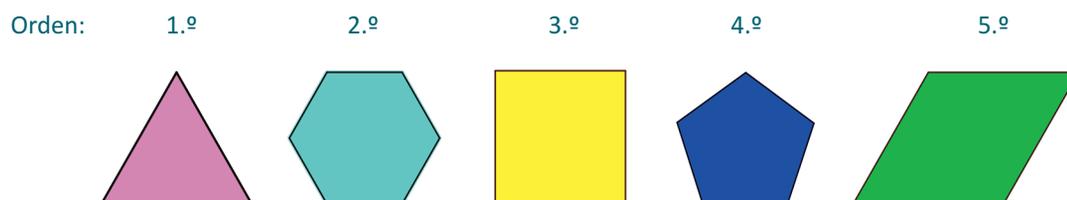
Para resolver la actividad, conviene recordar los nombres de los cinco polígonos.



Hacemos las siguientes consideraciones:

- Como el paralelogramo no tiene ninguna forma que cubra parte de él, debe haber sido el último polígono colocado sobre la mesa.
- El pentágono tiene el paralelogramo cubriendo parte de él, por lo que el pentágono debe haber sido colocado antes que el paralelogramo.
- El cuadrado tiene el pentágono y el paralelogramo cubriendo parte de él, por lo que el cuadrado debe haber sido colocado antes que el pentágono y el paralelogramo.
- El hexágono tiene el cuadrado y el paralelogramo cubriendo parte de él, por lo que el hexágono debe haber sido colocado antes que el cuadrado y el paralelogramo.
- El triángulo tiene el hexágono, el cuadrado y el pentágono cubriendo parte de él, por lo que el triángulo debe haber sido colocado antes que el hexágono, el cuadrado y el pentágono.

Por lo tanto, Hipatia debe haber colocado primero, el triángulo, luego, el hexágono, después, el cuadrado, a continuación, el pentágono y, finalmente, el paralelogramo.

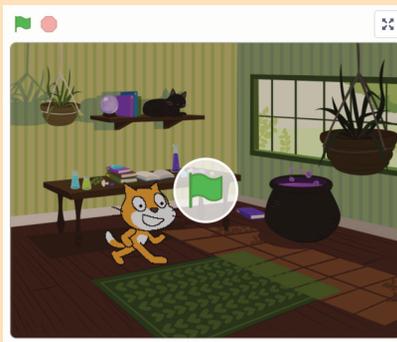


Este problema está estrechamente relacionado con el algoritmo, concepto fundamental del Pensamiento Computacional. El problema y las actividades de ampliación están planteadas con la finalidad de que los alumnos conecten con los algoritmos en su día, en su ámbito cercano, y detecten la presencia y la utilidad de los algoritmos en la vida cotidiana.

Finalmente, con la realización de las actividades de Algoritmo, ¿qué es eso? (<https://descargas.intef.es/recursos_educativos/ODES_SGOA/ESO/Matematicas/6C1_SACirculo_Matematico_Computacional/algoritmo_qu_es_eso.html>), una de las secciones pertenecientes a una Situación de Aprendizaje (SdA) del proyecto homónimo del INTEF – Ministerio de Educación, Formación Profesional y Deportes, se pretende realizar una primera aproximación más formal e histórica al concepto de algoritmo.

Actividad 4: Algoritmos de búsqueda. Adivina el número

En esta actividad tenéis que intentar descubrir el número que ha pensado el gato del programa realizado en Scratch. Pulsa el enlace <<https://scratch.mit.edu/projects/807157052/>> para iniciar una partida.



Para resolver esta actividad se propone seguir la siguiente secuencia:

1. Respuestas aleatorias, al azar.
Este método consiste en adivinar el número probando números al azar.
 - Inicia una nueva partida pulsando la bandera verde.
 - Prueba con números al azar hasta que encuentres la respuesta. Anota cuántos intentos habéis necesitado.
2. Búsqueda lineal:
El algoritmo comienza examinando el primer elemento de la lista de los posibles números, es decir, el 1. Si al introducir el 1 coincide con el valor que ha pensado el gato, hemos terminado; si no, continuamos con el siguiente valor de la lista, el 2. Si el valor coincide, hemos terminado; si no pasamos al siguiente número, y así sucesivamente.
 - Inicia una nueva partida pulsando la bandera verde.
 - Prueba a usar el algoritmo de búsqueda lineal para adivinar el número. Anota cuántos intentos habéis necesitado.
3. Búsqueda binaria:
El algoritmo comienza comparando el elemento medio de la lista de posibles valores. Si el elemento medio es igual al valor buscado, se devuelve la posición del elemento medio. Si el valor buscado es menor que el elemento medio, la búsqueda se limita a la mitad inferior de la lista,

y se repite el proceso con esa mitad. Si el valor buscado es mayor que el elemento medio, la búsqueda se limita a la mitad superior de la lista y se repite el proceso.

- Inicia una nueva partida pulsando la bandera verde.
- Prueba a usar el algoritmo de búsqueda binaria para adivinar el número. Anota cuántos intentos habéis necesitado.

4. Reflexionamos y analizamos la efectividad de cada algoritmo:

Para concluir esta actividad, reflexionaremos y analizaremos la efectividad de cada uno de los algoritmos. Llevaremos a cabo este proceso, primeramente de manera individual y, más tarde, lo pondremos en común con nuestro grupo.

Algunas preguntas que pueden ayudar a llevar a cabo este proceso de reflexión y valoración de cada algoritmo son las siguientes:

- ¿Cuál de los tres algoritmos de búsqueda es más eficiente? ¿Por qué?
- ¿Cuántos intentos, como máximo, necesitaréis realizar para localizar un número menor que 32? ¿Y para un número menor que 64?
- ¿Podrías indicar el número máximo de intentos necesarios para localizar un número menor que 128? ¿Y para localizar uno menor que 256, 512 y 1024?

Trabajad en equipo para localizar una respuesta razonada. Anotad las respuestas en una tabla como la siguiente o en una hoja de cálculo. ¿Observáis algún patrón?

Número menor que...	N.º de intentos máximos necesarios con el algoritmo		
	Búsqueda al azar	Búsqueda lineal	Búsqueda binaria
32			
64			
128			
256			
512			
1024			

Actividad 5: Figuras que esconden números. Tres números consecutivos

Las siguientes representaciones corresponden a tres números naturales consecutivos de tres cifras, escritos de menor a mayor, las cuales han sido reemplazadas por figuras geométricas:



1. ¿Qué números son?
2. ¿Cuál de las siguientes representaciones correspondería al siguiente número de la serie?



3. Dibuja la representación del número de la serie que continuaría al del apartado anterior, adoptando las decisiones que sean necesarias y justificándolas de manera razonada.
4. ¡Pon a prueba tu creatividad tu creatividad! Inventa un problema similar a partir de lo aprendido en la resolución de los apartados anteriores.

Esta actividad es la adaptación de un problema perteneciente a las pruebas de 2023 del Canguro Matemático organizadas por la FESPM.

En el enlace <<https://polypad.amplify.com/p/x4Oy7ahSpA8VZQ>> se puede acceder a la versión digital del problema elaborado en Polypad.

Las siguientes representaciones corresponden a tres números naturales consecutivos, escritos de menor a mayor, de tres cifras cada uno, las cuales han sido reemplazadas por figuras geométricas.



1 ¿Qué números son?

, ,

2 ¿Qué representación correspondería al siguiente número de la serie? Introduce el valor 1 en la correcta y 0 en el resto.



3 Dibuja la representación del número de la serie que continuaría al del apartado anterior, adoptando las decisiones que sean necesarias, justificando de manera razonada las mismas.

4 ¡Pon a prueba tu creatividad! Inventa un problema similar a partir de lo aprendido en la resolución de los apartados anteriores.

3. ¿Qué pasa cuando una máquina aprende?

Las máquinas no nacen sabiendo. No razonan, no entienden ni tienen intuición. Sin embargo, pueden aprender de ejemplos, reconocer patrones y tomar decisiones si se les entrena adecuadamente. A esto lo llamamos aprendizaje automático, *machine learning* en inglés, es una de las ramas más fascinantes de la inteligencia artificial y también una de las más accesibles para trabajar en el aula con sentido matemático.

Una máquina no aprende como una persona, pero sigue un proceso que nos resulta familiar: recoge datos, detecta relaciones, genera una regla y la aplica a nuevos casos. Si acierta, mantiene su modelo. Si se equivoca, lo ajusta. En otras palabras, ensaya, se equivoca y mejora. ¿No es eso lo que hacemos nosotros cuando resolvemos un problema matemático y revisamos nuestra estrategia?

El punto de partida para que una máquina aprenda es un conjunto de ejemplos. Por ejemplo, si queremos que una IA clasifique números como pares o impares, necesitamos mostrarle muchos casos con sus respuestas correctas. La máquina no sabe qué significa «par», pero puede identificar una regularidad: todos los números que terminan en 0, 2, 4, 6 u 8 están etiquetados como «par». A partir de ahí, construye una regla interna que puede usar para clasificar nuevos números.

Este proceso puede trasladarse al aula a través de herramientas como LearningML, que permiten entrenar modelos con texto, imágenes o datos numéricos. Lo interesante es que cada decisión automática se basa en operaciones matemáticas: cálculos, comparaciones, distancias, medias, transformaciones gráficas. Por tanto, el aprendizaje automático ofrece una oportunidad perfecta para trabajar contenidos como proporcionalidad, estadística, funciones o geometría analítica, pero en un contexto actual y motivador.

Además, los errores que comete una inteligencia artificial también se pueden analizar matemáticamente. ¿Por qué ha fallado? ¿Faltaban ejemplos? ¿Había demasiadas excepciones? ¿Las variables

elegidas no eran significativas? Estas preguntas nos llevan a reflexionar sobre la calidad de los datos, la construcción de modelos y el margen de error, es decir, a aplicar pensamiento crítico y competencia matemática en profundidad.

Trabajar con una inteligencia artificial no implica dejar de pensar. Todo lo contrario: exige planificar, representar, probar y justificar. Y cuando los alumnos son capaces de entrenar una máquina para que resuelva un problema sencillo, comprenden mejor cómo piensa la máquina..., y cómo piensan ellos mismos.

3.1 ¿Y si hablamos con una máquina?

Las máquinas pueden predecir muchas cosas, pero (al menos de momento...) no entienden realmente lo que hacen. Por eso, hoy más que nunca, necesitamos aprender a pensar por nosotros mismos.

Hasta hace poco, que una máquina pudiera mantener una conversación, escribir una historia, resolver un problema o crear una imagen parecía cosa de ciencia ficción. Hoy, la inteligencia artificial generativa (IAG) está cambiando cómo usamos la información y cómo interactuamos con la tecnología. Pero lo más importante no es lo que puede hacer, sino cómo nos ayuda a pensar mejor sobre nuestro propio pensamiento.

Una inteligencia artificial generativa no responde porque entienda. Lo que hace es predecir la palabra, el número o la imagen más probable según millones de ejemplos anteriores. Aprende patrones, estructuras y relaciones estadísticas para generar frases, ideas o soluciones que parecen creativas. Pero detrás de todo eso hay matemáticas: cálculo, probabilidad y optimización.

Esto nos abre una puerta para usar estos modelos y entrenar nuestro pensamiento crítico y matemático. Cuando recibimos una respuesta de una inteligencia artificial, debemos preguntarnos: ¿es fiable? ¿Qué pasos matemáticos se han usado? ¿Falta algo? Evaluar las respuestas de una inteligencia artificial es igual que evaluar un procedimiento humano: hay que comprender, argumentar y revisar. Así, la IAG se convierte en un recurso para pensar más, no para pensar menos.

Además, estos modelos nos permiten reformular problemas, explorar nuevas preguntas o automatizar tareas sencillas, siempre desde el control humano. No se trata de dejar de pensar, sino de enriquecer nuestra reflexión. Por ejemplo, al interactuar con un simulador creado con IAG, como el «Adivino Matemático» que presentamos a continuación, no solo se ve la respuesta, sino que se puede analizar el algoritmo oculto, intentar generalizarlo y discutirlo con los compañeros.

Trabajar con IAG también nos lleva a hablar de ética, sesgos, originalidad y verificación. ¿Puede una inteligencia artificial equivocarse? ¿Qué pasa si generaliza mal? ¿Cómo influye el tipo de datos con los que ha sido entrenada? Estas preguntas, lejos de ser solo técnicas, son fundamentales para pensar críticamente y pueden formar parte de un círculo matemático computacional.

Hablar con una máquina no es dejar que piense por ti. Es una oportunidad para explicar mejor lo que haces, cuestionar lo que recibes y aprender a distinguir lo correcto. Aprender matemáticas no es solo calcular mecánicamente, sino razonar con sentido en un mundo lleno de respuestas automáticas.

3.2 Simulador «Adivino Matemático».

El resultado de una conversación real con una máquina

Como ejemplo de esta idea de hablar e interactuar con una máquina, presentamos el «Adivino Matemático», un simulador elaborado con la inteligencia artificial generativa Claude. La conversación con Claude comenzó así:

La idea  es que conviertas en simulador este problema para que hagas de «Adivino». Hay 24 fichas en una mesa. En una primera jugada, cada uno de tres jugadores toma 1, 2 o 3 fichas, todos distintos. En una segunda jugada, el jugador 1 toma tantas fichas como las que tomó en la primera jugada, el jugador 2 toma el doble de las que tomó en la primera jugada y el jugador 3 toma el cuádruple de las que tomó en la primera jugada. Tu reto es averiguar cuántas fichas tomó cada uno de los tres jugadores en la jugada inicial, sabiendo cuántas fichas han quedado sin coger encima de la mesa. Para hacerlo más real, místico y contextualizado, pide que se introduzcan los nombres de cada uno de los tres jugadores. Explica al final tu razonamiento matemático justificativo para llegar a dicha conclusión.

Como se puede comprobar, este simulador es el fruto de una colaboración, un diálogo entre una persona y una máquina, donde se fueron detallando las condiciones del problema, ajustando respuestas y creando un artefacto educativo que permite aprender matemáticas y pensamiento computacional de forma interactiva y motivadora.

Actividad 6: Adivino matemático

Ahora, tú también puedes jugar con tus compañeros, introducir nombres, probar diferentes escenarios y desafiar a la IA a adivinar correctamente. El simulador te mostrará cómo se llega a la solución con una explicación paso a paso.

Puedes probarlo *online* en el enlace <<https://claude.ai/public/artifacts/e642a969-75dd-4161-903a-e49407d3e446>>.

Adivino matemático • MatemÁTICAS: 1,1,2,3,5,8,13,...
Elaborado por el profesor Luis Miguel Iglesias Albarrán

En este juego hay 24 fichas en una mesa. En una primera jugada, tres jugadores toman 1, 2 o 3 fichas cada uno (todos diferente cantidad). En una segunda jugada:

- El primer jugador toma tantas fichas como tomó en la primera jugada
- El segundo jugador toma el doble de fichas que tomó en la primera jugada
- El tercer jugador toma el cuádruple de fichas que tomó en la primera jugada

¡Adivinaré cuántas fichas tomó cada jugador en la primera jugada basándome en las fichas restantes!

Jugador 1:

Jugador 2:

Jugador 3:

[Continuar](#)

Si te gustó, puedes compartirlo en redes sociales. Puedes encontrar más recursos y actividades matemáticas en la web MatemÁTICAS: 1,1,2,3,5,8,13,...
<https://matematicas11235813.luismiglesias.es> y en el canal de Youtube del profesor Luis M. Iglesias @luismiglesias.

Simulador Adivino Matemático IAG

4. Aprendizaje automático, inmersión en la IA, para resolver problemas

4.1 ¿Qué es LearningML y cómo se usa?

Matemáticas y pensamiento computacional están en el corazón de la inteligencia artificial: aprender con herramientas de *machine learning* es aprender a pensar y a resolver problemas con sentido.

En este cuadernillo para el Día Escolar de las Matemáticas queremos mostrar cómo las matemáticas pueden ayudar a entender y usar la inteligencia artificial. LearningML es una herramienta educativa que permite entrenar modelos de IA de forma visual y sencilla, sin necesidad de saber programar.

1. LearningML como herramienta para introducir el Aprendizaje Automático (Machine Learning) en el aula de matemáticas.

LearningML es un entorno para trabajar con datos, clasificaciones y relaciones matemáticas que nos permite:

- Organizar datos: agrupando y etiquetando ejemplos.
- Observar patrones: reconociendo similitudes y diferencias.
- Construir modelos: que representan matemáticamente esas regularidades.
- Evaluar resultados: interpretando aciertos, errores y probabilidades.

Así, conceptos como proporciones, estadísticas, variables o funciones se trabajan de forma práctica y significativa.

2. ¿Cómo funciona LearningML?

Incluye las siguientes fases:

- Introducir ejemplos. Por ejemplo: clasificar números como pares o impares, positivos o negativos,...
- Entrenar el modelo. El programa busca patrones y crea reglas a partir de los ejemplos.
- Hacer predicciones. Se prueban nuevos datos y se observa si el modelo acierta o falla.
- Corregir y mejorar. Se ajustan los datos o etiquetas y se vuelve a entrenar para mejorar la precisión.



Esquema del proceso de aprendizaje automático.
Construcción de modelos de IA en LearningML

3. ¿Por qué usar LearningML en matemáticas?

Entre las razones para ello podemos incluir:

- Ayuda a ver con claridad conceptos abstractos como funciones, modelos o patrones.
- Desarrolla el pensamiento computacional aplicado a contextos matemáticos reales.
- Fomenta la reflexión, la curiosidad y la experimentación.
- Conecta las matemáticas con el mundo tecnológico actual.

En resumen: LearningML transforma las matemáticas en una experiencia activa, donde se puede experimentar, probar, mejorar y descubrir cómo funciona la inteligencia artificial.

Puedes usar esta herramienta educativa gratuita, desarrollada por el informático español Juan David Rodríguez, accediendo a través del enlace <<https://web.learningml.org>>.

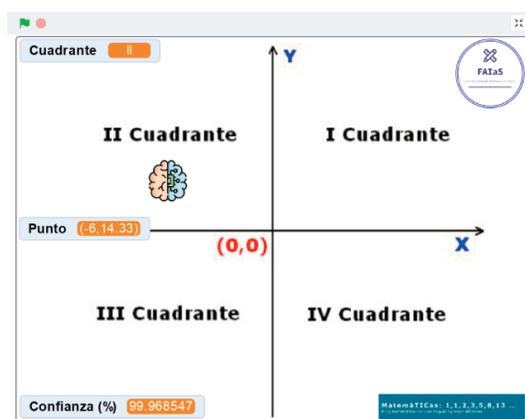
4.2 Cuando las máquinas aprenden a resolver problemas a partir de los datos. Proyecto coordenadas y cuadrantes. Modelización envío de paquetes

Vamos a presentar un ejemplo real de cómo usar IA en matemáticas. Esta propuesta muestra cómo enseñar a una máquina a ubicar puntos del plano en su cuadrante correcto usando LearningML.

Actividad 7: ¿A qué cuadrante pertenece?

1. Crea un modelo en LearningML con dos columnas: una para la coordenada X y otra para la Y.
2. Define cuatro categorías: los cuatro cuadrantes del plano cartesiano.
3. Alimenta el modelo con ejemplos de puntos reales en cada cuadrante.
4. Entrena el modelo.
5. Comprueba si «ha aprendido» y predice bien en nuevos casos.

Más tarde, ese mismo modelo se conecta a un programa en Scratch para hacerlo aún más visual. Se puede ver la propuesta completa con vídeo explicativo y recursos en el enlace <<https://matematicas11235813.luismiglesias.es/2022/06/19/propuesta-didactica-inteligencia-artificial-con-learningml-modelo-numerico-matematicas-puntos-coordenadas-y-cuadrantes/>>.



Clasificación de puntos en el plano usando datos numéricos
(Ejemplo: coordenadas → predicción de cuadrante)

Podemos hacer una extensión contextualizada de esta situación como se refleja en la siguiente actividad de modelización.

Actividad 8: ¿A qué tipo de envío pertenece?

Ahora imaginemos que no estamos trabajando con puntos en un plano... sino con envíos de paquetes. Supongamos que una empresa quiere clasificar los paquetes que se deben entregar según dos características:

1. Si son urgentes o no urgentes.
2. Si son nacionales o internacionales.

Podemos representarlo, como refleja la siguiente tabla, igual que si fuera un plano:

	Nacional	No nacional
Urgente	Cuadrante I	Cuadrante III
No urgente	Cuadrante II	Cuadrante IV

Entonces:

- Un paquete urgente y nacional → Cuadrante I.
- Un paquete no urgente y nacional → Cuadrante II.
- Un paquete urgente y no nacional → Cuadrante III.
- Un paquete no urgente y no nacional → Cuadrante IV.

Usando esta tabla podemos alimentar a LearningML con ejemplos de envíos y su clasificación. A partir de ahí:

1. El modelo aprende a clasificar los paquetes en los cuadrantes.
2. Podemos probar con nuevos datos para ver si predice bien.
3. Comparamos estrategias: ¿lo resolvemos con reglas fijas o con datos y ejemplos?

En definitiva, tenemos dos formas de resolver el mismo problema:

Forma 1. Solución basada en reglas (deductiva)

- Si es urgente y nacional → Cuadrante I.
- Si no es urgente y nacional → Cuadrante II.
- Si es urgente y no nacional → Cuadrante III.
- Si no es urgente y no nacional → Cuadrante IV.

Forma 2. Solución basada en datos (inductiva)

- Le damos al programa muchos ejemplos etiquetados, y él descubre solo las reglas.
- Esto es aprendizaje automático: la máquina aprende a partir de los datos.

Así se puede ver cómo una misma situación puede resolverse de dos formas diferentes: con reglas o con ejemplos. Y aprender cómo lo hace una IA nos ayuda también a mejorar nuestra forma de razonar. Vamos entrenar nuestro propio modelo y ponerlo a prueba. ¡Las matemáticas y la inteligencia artificial nos están esperando!

4.3 Propuesta de actividad. Entrenar, probar, mejorar: tu primer modelo

Entrenar una máquina es como entrenar a un matemático: requiere ejemplos, pruebas y paciencia.

Ahora que sabemos qué es LearningML y cómo se conecta con las matemáticas, vamos a crear nuestro primer modelo de inteligencia artificial en cuatro sencillos pasos.

Actividad 9

Paso 1: Recoger ejemplos

Elegimos un conjunto de datos que se puedan clasificar fácilmente. Por ejemplo: números pares (Par) e impares (Impar). Crearemos una tabla con los ejemplos y asignaremos una etiqueta a cada uno. En LearningML, eligeremos modelo numérico, crearemos dos categorías: Par e Impar, y añadiremos unos 10 elementos de cada una.

Ejemplo	Etiqueta
2	Par
3	Impar
4	Par
7	Impar
...	...

Paso 2: Entrenar el modelo

Subiremos los datos a LearningML para que el sistema «aprenda» a distinguir las categorías basándose en los ejemplos. Este proceso se llama entrenamiento y es como enseñar a la máquina a reconocer patrones.

Paso 3: Probar el modelo

Probaremos con nuevos ejemplos que no se hayan usado antes y observaremos si el modelo clasifica correctamente. Si falla, habrá que pensar en qué ejemplos se podrían añadir o corregir para que mejore.

Paso 4: Mejorar el modelo

Añadiremos más ejemplos o ajustaremos las etiquetas para perfeccionar el aprendizaje. Hay que recordar que la calidad y cantidad de datos afectan la precisión.

5. Mirando hacia el futuro: pensamiento crítico y matemáticas en tiempos de inteligencia artificial

Finalizamos este cuadernillo con una invitación a seguir aprendiendo y pensando con rigor. A lo largo de estas páginas hemos vinculado la resolución de problemas con el pensamiento computacional y, este, con el uso educativo y matemático de la inteligencia artificial. Frente a la fascinación ingenua o al recelo infundado que a veces despiertan estas tecnologías, como docente de matemáticas con dos décadas de experiencia promoviendo el uso de calculadoras y herramientas digitales para enseñar y para aprender matemáticas, defiendo un enfoque crítico, reflexivo y didáctico.

Aprender con IA significa, sobre todo, razonar más y mejor, no sustituir el pensamiento propio. Supone una oportunidad para analizar, argumentar, compartir ideas y crecer junto a tus compañeros y profesores.

Por ello, esperando que el cuadernillo haya sido útil y servido para aprender, comparto a modo de cierre de este, un análisis DAFO, para reflexionar sobre la incorporación de la IA en el aula, así como un decálogo, dirigido al alumnado, que resume principios esenciales para un uso responsable, riguroso y provechoso de estas herramientas en el aprendizaje de las matemáticas.

5.1 Análisis DAFO. Inteligencia Artificial y Enseñanza-Aprendizaje de las matemáticas

DEBILIDADES	AMENAZAS
<ul style="list-style-type: none"> — Riesgo de dependencia tecnológica. — Posibilidad de recibir respuestas imprecisas o incorrectas. — Falta de formación específica del profesorado. — Limitaciones en el desarrollo de habilidades matemáticas profundas. 	<ul style="list-style-type: none"> — Sesgos en la información generada. — «Alucinaciones» de la IA. — Reemplazo indebido de la interacción humana. — Desigualdad/brecha de acceso a la tecnología.
FORTALEZAS	OPORTUNIDADES
<ul style="list-style-type: none"> — Uso innovador de la tecnología. — Acceso a múltiples métodos de resolución. — Apoyo activo del docente. — Estímulo del pensamiento crítico y computacional. 	<ul style="list-style-type: none"> — Aprendizaje personalizado. — Desarrollo del pensamiento y del juicio crítico. — Integración de herramientas digitales y tecnologías emergentes como la IA, y nuevos enfoques metodológicos en el aula. — Colaboración más rica en el aula.

5.2 Decálogo para el uso responsable y provechoso de la IA en matemáticas

1. La IA es una aliada, no un atajo.
Utiliza herramientas de IA para explorar ideas, no para copiar soluciones.
2. Revisa y verifica siempre la información.
Sé crítico y comprueba las respuestas porque la IA puede «alucinar» con datos erróneos.
3. Aprende a reconocer sesgos.
La IA refleja los sesgos de sus datos de entrenamiento. Revisa y cuestiona siempre la validez de las respuestas comprobando si tiene sentido en el contexto en el que estás trabajando.
4. Explora múltiples enfoques.
La IA permite descubrir distintas maneras de resolver un mismo problema. Aprovéchalo.
5. Participa activamente.
Usa tus habilidades lingüísticas y matemáticas para dialogar con la IA, plantear buenas preguntas y obtener respuestas útiles y razonadas.
6. Desarrolla tu pensamiento matemático.
Usa la IA como apoyo, pero asegúrate de entender por ti mismo los conceptos y procesos.

7. Refuerza tu pensamiento crítico.
Analiza los resultados que te ofrezca la IA y pregunta «¿por qué?», «¿cómo?», «¿qué pasaría si...?».
8. Interactúa con la IA, pero también con tus compañeros y compañeras de clase.
Usa la IA como herramienta de apoyo, pero no como sustituto del trabajo colaborativo en equipo.
9. Controla la procedencia de las fuentes.
La IA no siempre cita sus fuentes. Controla, pregunta, revisa y busca referencias confiables y explícitas.
10. Reconoce y valora el papel de tu profesor/a.
La IA puede ser útil, pero es el docente quien te guía, te enseña a razonar y te acompaña en tu crecimiento matemático y personal.

6. Recursos y referencias

6.1 Plataformas y herramientas digitales

- AMPLIFY (2025), *Polypad*, página web, consultada el 1 de septiembre de 2025 en <https://polypad.amplify.com/>.
- ANTHROPIC (2025), *Claude*, página web, consultada el 26 de agosto de 2025 en <https://claude.ai>.
- FUNDACIÓN SCRATCH (2025), *Scratch*, página web, consultada el 1 de septiembre de 2025 en <https://scratch.mit.edu/>.
- OPENAI (2025), *ChatGPT*, página web, consultada el 30 de agosto de 2025 en <https://chatgpt.com>.
- RODRÍGUEZ, J. D. (2025), *LearningML - AI made easy*, página web, consultada en 30 de agosto de 2025 en <https://learningml.org/>.

6.2 Blogs y recursos didácticos

- IGLESIAS, L. M. (2025a), «Pensamiento Computacional», *MatemáTICas: 1,1,2,3,5,8,13,...*, blog, consultado el 28 de agosto 2025 en <https://matematicas11235813.luismiglesias.es/category/pensamiento-computacional/>.
- (2025b), «Inteligencia artificial», *MatemáTICas: 1,1,2,3,5,8,13,...*, blog, consultado el 28 de agosto 2025 en <https://matematicas11235813.luismiglesias.es/category/inteligencia-artificial/>.

6.3 Para saber más

- DOMINGOS, P. (2015), *The Master Algorithm*, Basic Books.
- IGLESIAS, L. M. (2022, 8 de junio), «Participación en ‘Learning, Training & Teaching Activity (LTTA) FAIaS Braga · Fomentando la Inteligencia Artificial en las Escuelas’», *MatemáTICas: 1,1,2,3,5,8,13,...*, blog, consultado el 28 de agosto 2025 en <https://matematicas11235813.luismiglesias.es/2022/06/08/participacion-en-ltta-faias-braga-fomentando-inteligencia-artificial-escuela/>.

- (2023, 15 de diciembre), «Inteligencia Artificial aplicada al contexto educativo y consideraciones éticas», vídeo, *YouTube*, recuperado el 2 de septiembre de 2025 en <<https://www.youtube.com/live/Akg4RvLpN0o?t=451s>>.
 - (2025), «Integración de la Inteligencia Artificial en la Enseñanza y el Aprendizaje de las matemáticas en contextos multiculturales, enfocada en la transformación educativa sostenible»; en Actas del XIX Congreso de Enseñanza y Aprendizaje de las Matemáticas (CEAM), publicación en curso.
 - (2025, 15 de junio), «ChatGPT ya hace los deberes, ahora toca rediseñar la educación», *Huelva información*, <https://www.huelvainformacion.es/huelva/chatgpt-deberes-ahora-toca-redisenar-educacion_0_2004143669.html>.
 - (s. f.), «Creamos nuestro Círculo Matemático Computacional (CMC)», *Intef, Situaciones de Aprendizaje*, página web, consultado el 28 de agosto de 2025 en <https://descargas.intef.es/recursos_educativos/ODES_SGOA/ESO/Matematicas/6C1_SA_Circulo_Matematico_Computacional/index.html>.
- INTERNATIONAL SOCIETY FOR TECHNOLOGY IN EDUCATION (2025), *ISTE*, página web, consultada el 28 de agosto de 2025 en <<https://info.iste.org/recursos-2018>>.
- «La IA en las aulas, una nueva ecuación para modelar el futuro educativo» (2024, 25 de abril), *Periódico Magisterio*, <<https://www.magisnet.com/2024/04/la-ia-en-las-aulas-es-una-nueva-ecuacion-para-modelar-el-futuro-educativo/>>.
- MORENO, J. (2014), «¿Qué es el pensamiento computacional?», *Programamos*, página web, consultada el 30 de agosto de 2025 en <<https://programamos.es/que-es-el-pensamiento-computacional/>>.
- WING, J. (2006), «Computational Thinking», *Communications of the ACM*, 49(3), 33-35. recuperado el 1 de septiembre de 2025 en <<https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf>>.
- (2011), «Computational Thinking – What and Why», *The Link Magazine* 6, 20-23, recuperado el 1 de septiembre de 2025 en <<https://people.cs.vt.edu/~kafura/CS6604/Papers/CT-What-And-Why.pdf>>.

6.4 Vídeos

- IGLESIAS, L. M. (2023), «Representación de algoritmos: Área de un triángulo · Diagrama de flujo y pseudocódigo - Flowgorithm», vídeo, *YouTube*, consultado el 28 de agosto de 2025 en <<https://www.youtube.com/watch?v=xn0DnRV8AU4>>.
- (v. f.), «Matemáticas: 1,1,1,2,3,5,8,13, ...», canal de *YouTube*, consultado el 28 de agosto de 2015 en <<https://www.youtube.com/playlist?list=PLD87Aneh2Axp51PUAGKySSsNoBOPZUji>>.

Listado de títulos de años anteriores

2000	Pon un poliedro en tu centro
2001	Relojes de sol
2002	Las matemáticas de Alicia y Gullivert. Lo grande y lo pequeño
2003	La rosa de los vientos. El rumbo y la navegación
2004	Frutas y matemáticas
2005	El Quijote y las matemáticas
2006	Mirar el arte con ojos matemáticos
2007	Matemáticas y educación para la paz
2008	Música y matemáticas. La armonía de los números
2009	La ciudad y las matemáticas
2010	Prensa y matemáticas
2011	Las matemáticas de la química
2012	Matemáticas y economía
2013	Hydria-Matemáticas
2014	Matemáticas y computación
2015	Matemáticas jugando
2016	Matemáticas en el deporte
2017	Matemáticas para el turismo
2018	Matemáticas y el poder de los votos
2019	Las matemáticas de los mapas
2020	Con mi móvil aprendo matemáticas
2021	Matemáticas en la sanidad vegetal
2022	Las matemáticas de los caminos de Santiago
2023	Las matemáticas de los volcanes
2024	Las matemáticas de la agricultura
2025	Magia y Matemáticas

Para mas información puede consultarse la página web de la FESPM.

