



Horizonte

HOMOLOGADOS

CURSO HOMOLOGADO · 100 HORAS

Pensamiento computacional, robótica y STEAM en el aula

Del saber básico al aula que tienes, empezando sin enchufar nada

Profesorado en activo y en formación de Educación Infantil, Educación Primaria, Educación Secundaria Obligatoria, Bachillerato y Formación Profesional, con independencia de su especialidad y de que el centro disponga o no de equipamiento; tutores y maestros generalistas, profesorado de Matemáticas, Ciencias, Tecnología y Digitalización, Educación Plástica y Visual y Música implicado en propuestas STEAM; coordinadores de digitalización y equipos directivos que tengan que decidir en qué se gasta el presupuesto de material

Ficha técnica



Denominación	Pensamiento computacional, robótica y STEAM en el aula
Modalidad y duración	En línea · 100 horas
Destinatarios	Profesorado en activo y en formación de Educación Infantil, Educación Primaria, Educación Secundaria Obligatoria, Bachillerato y Formación Profesional, con independencia de su especialidad y de que el centro disponga o no de equipamiento; tutores y maestros generalistas, profesorado de Matemáticas, Ciencias, Tecnología y Digitalización, Educación Plástica y Visual y Música implicado en propuestas STEAM; coordinadores de digitalización y equipos directivos que tengan que decidir en qué se gasta el presupuesto de material
Ámbito	Estatul
Estructura	4 bloques · 5 epígrafes por bloque · 20 figuras · 60 aplicaciones prácticas · 4 pruebas de bloque · 1 tarea final optativa
Certificación	Horizonte Homologados · Universidad Católica de Ávila
Cierre de la edición	Septiembre de 2026

Preliminares	4
Bloque 1. Concepto de pensamiento computacional, prescripción curricular y alcance de sus efectos	
1.1. Delimitación del pensamiento computacional frente a la competencia informática	10
1.2. Presencia del pensamiento computacional en el currículo vigente por etapa educativa	14
1.3. Evidencia sobre la transferencia a otras áreas y afirmaciones sin respaldo	18
1.4. Actividades desconectadas: alcance del trabajo sin dispositivos	21
1.5. Diagnóstico inicial: inventario de recursos, horario y criterios de adquisición	24
Bloque 2. Aprendizaje del pensamiento computacional: operaciones, progresión y recursos	
2.1. Las operaciones del pensamiento computacional y la depuración de errores	29
2.2. Progresión de los aprendizajes por edad y etapa educativa	33
2.3. Aportación diferencial de los recursos desconectados, los lenguajes de bloques y la robótica	36
2.4. Delimitación del enfoque STEAM frente a la actividad manipulativa sin contenido	40
2.5. Factores de fracaso de una sesión y su relación con el diseño didáctico	44
Bloque 3. Aplicación en el aula a partir de los recursos disponibles	
3.1. Diseño de una secuencia desconectada con continuidad didáctica	49
3.2. Transición a la programación con un número limitado de dispositivos	53
3.3. Organización de una sesión de robótica en un grupo ordinario	57
3.4. Diseño de un reto STEAM con contenido científico verificable	60
3.5. Ubicación de la propuesta en la programación anual y asignación horaria	64
Bloque 4. Evaluación del pensamiento computacional y sostenibilidad de la propuesta	
4.1. Objeto de la evaluación: producto, proceso y concepto	68
4.2. Procedimientos de observación de aprendizajes sin registro escrito	71
4.3. Alcance y limitaciones de las pruebas de pensamiento computacional	74
4.4. Criterios de calificación y control del efecto del acceso previo a los recursos	77
4.5. Plan de incorporación de los cambios y calendario de revisión	80
Cierre: la tarea final optativa	83
Bibliografía consolidada	86
Referencias citadas con reserva	98



Qué tienes delante

En el armario del aula de material hay seis robots de suelo con las pilas gastadas. Llegaron hace dos cursos con una subvención, se usaron en la semana cultural, salieron en la foto de la web del centro y ahí siguen. Nadie del claustro sabe muy bien de quién son, ni a qué área pertenecen, ni qué había que hacer con ellos más allá de que el alumnado los mueva por una alfombra de cuadros. Y mientras tanto, en tu programación hay un saber básico que dice literalmente «Estrategias para la interpretación y modificación de algoritmos sencillos» y otro, en un área distinta, que dice «Iniciación en la programación a través de recursos analógicos (actividades desenchufadas) o digitales», y esos dos saberes hay que darlos igual, tengas robots o no los tengas. Fíjate en la conjunción, porque es la que sostiene el curso entero: la norma no obliga a lo digital, ofrece el recurso analógico como vía suficiente y además lo nombra primero.

Ese es el desajuste que trae aquí a la mayoría: lo que el currículo pide no es lo que la tienda vende, y lo que la tienda vende lleva dos años sin pilas. Este manual lo resuelve en el orden correcto. Empieza por situarte —qué es exactamente el pensamiento computacional y qué no es, dónde está escrito en el currículo de tu etapa y con qué palabras exactas, y qué se puede prometer de todo esto y qué no—. Sigue por el fundamento: las operaciones que hay debajo de cualquier actividad de programación, la depuración incluida, que es la que casi ningún material escolar desarrolla; qué puede hacer de verdad un niño de cuatro años, uno de ocho y uno de catorce; y qué añade cada material cuando se sube un escalón. Baja después al aula, y baja empezando por donde se puede empezar siempre: sin enchufar nada. Papel, tiza, tarjetas, cuerpo y suelo. Desde ahí sube al escenario de un solo dispositivo con proyector, al del robot compartido por toda la clase y al del carro de tabletas. Y termina en la evaluación, que es donde se juega la credibilidad de esta materia: cómo se acredita un aprendizaje que ocurre mientras el alumno se equivoca y corrige, y que casi nunca deja rastro en un cuaderno.

Se puede hacer entero sin comprar nada. No es una concesión al centro pobre: es el diseño. Las sesenta aplicaciones prácticas se resuelven con papel, con el currículo de tu comunidad y con el material que ya está en tu centro; las cinco que pueden apoyarse en robot o en tableta llevan siempre su alternativa desenchufada, escrita con el mismo detalle y con el mismo tiempo asignado, para que nadie se quede a mitad de curso por no tener un aparato.

Lo que no vas a encontrar es la promesa de que aprender a programar enseña a pensar. Conviene decirlo en la primera página, porque es la afirmación más repetida del sector y la peor sostenida: el metaanálisis que la sostiene encuentra transferencia, pero la encuentra pequeña, concentrada en lo que se parece a programar, casi nula en rendimiento escolar e indistinguible de cero en competencia lectora, y cuatro veces menor cuando el grupo de comparación hace otra actividad buena en lugar de no hacer nada. Tampoco vas a encontrar que el robot desarrolle por sí solo el pensamiento computacional, ni que la A de STEAM mejore el aprendizaje de las demás materias, ni el dato del 65 % de escolares que trabajarán en empleos que aún no existen, que circula por todas las presentaciones del ramo y no tiene fuente identificable. Ni una marca, ni un modelo, ni un kit recomendado: ni con nombre propio ni por descarte.

Al terminar tendrás una carpeta con sesenta documentos escritos por ti, y ninguno es un trabajo de clase: todos hablan de tu centro, de tu programación y de tu grupo. Del bloque 1 salen los que te sitúan —la auditoría del vocabulario de tu centro, tu definición operativa, el mapa curricular de tu etapa con la letra de la norma, tu decreto autonómico fichado, las fichas de lectura de los metaanálisis, el inventario de cuatro columnas y el filtro de compra aplicado a un kit real—. Del bloque 2, los que te dan criterio: las cinco operaciones localizadas en actividades que ya usas, la escala de progresión de tu etapa, la tabla de los tres escalones, la prueba del algodón sobre un reto ajeno y la previsión de los cinco puntos de rotura. Del bloque 3, lo que entra en clase: la

secuencia desenchufada completa con sus tarjetas imprimibles, el plan de subida para tu escenario real, el protocolo de sesión con su plan de contingencia, el reto STEAM en versión de coste cero y en versión ampliada, y el encaje en el calendario con el reparto de horas. Y del bloque 4, los que lo evalúan y lo dejan montado para el curso que viene: la matriz producto-proceso-concepto, el plan de observación aplicado en una sesión real, la elección justificada de instrumento, la revisión de tu propia calificación en clave de equidad y las tres decisiones con su documento de continuidad.

Ningún documento del curso los reúne todos: eso lo hace la tarea final optativa, descrita al cierre del manual. Esa carpeta es el resultado material del curso, y es tuya.

Objetivo general

Que al terminar el curso seas capaz de distinguir el pensamiento computacional de la programación, de la robótica educativa y de la competencia digital, y de localizar en el currículo vigente de tu etapa y de tu comunidad los saberes y criterios que te obligan a trabajarlo, con la letra exacta de la norma; de exponer con datos qué transfiere y qué no transfiere esta forma de trabajo a otras áreas, y de explicarlo a un claustro o a una familia sin prometer de más; de diseñar y poner en práctica una secuencia desenchufada anclada a un saber básico de tu propia programación; de planificar la subida escalonada hasta la programación por bloques y, si lo hay, hasta el robot, con el equipamiento real de tu centro y no con el del folleto; de construir un reto STEAM en el que las cinco letras hagan trabajo y no decoración; y de evaluar el proceso —descomposición, depuración, decisiones y trabajo en equipo— con criterios explícitos que no premien al alumnado que ya tenía el material en casa.

Competencias

1. **Distinguir** pensamiento computacional, programación, robótica educativa, STEAM y competencia digital, y determinar cuál de ellos describe realmente lo que se hace en tu centro cuando se dice que se hace «robótica».
2. **Identificar** en el currículo vigente de tu etapa —y en el decreto autonómico que lo desarrolla— los saberes básicos, las competencias específicas y los criterios de evaluación que prescriben el pensamiento computacional, citándolos con la letra de la norma, y determinar cuáles son obligatorios y cuáles dependen de la optatividad del centro.
3. **Determinar** qué sostiene y qué no sostiene la evidencia disponible sobre la transferencia del pensamiento computacional a otras áreas, sobre el efecto de la robótica educativa y sobre la eficacia de las actividades desenchufadas, y reconocer el papel del sesgo de publicación y de la condición de comparación en los resultados que se citan.
4. **Diseñar** una secuencia desenchufada de tres a cinco sesiones anclada a un saber básico de tu área, con su producto observable, sin usar ningún dispositivo.
5. **Planificar** la escalada de esa secuencia hasta la programación por bloques y hasta el robot partiendo del inventario real de tu centro, con la organización de aula, los tiempos y el plan de contingencia para cuando el material falle.
6. **Evaluar y defender** el aprendizaje resultante: qué se observa, en qué momento, con qué registro, cómo se distingue lo que aprendió el equipo de lo que aprendió cada alumno, y qué se le contesta al equipo directivo cuando pregunta qué ha dado de sí la inversión en robots.

Cómo está hecho este manual

Cuatro bloques de cinco epígrafes. Cada epígrafe tiene una extensión pareja —alrededor de mil quinientas palabras—, una figura propia y, al final, **tres aplicaciones prácticas**.

Las aplicaciones prácticas son el curso. La lectura sola no sostiene cien horas de formación y este manual no finge lo contrario: **la lectura de los veinte epígrafes y el estudio de las veinte figuras son 7,5 horas**, el repaso y las cuatro pruebas de bloque suman 5, y las **87,5 horas de trabajo**

aplicado restantes están en las sesenta aplicaciones prácticas. El cómputo es una estimación hecha aplicación por aplicación, sobre lo que cuesta producir cada documento, y no un reparto a partes iguales. Por eso los bloques no valen lo mismo: el 1 suma 21 horas, porque su trabajo no es leer sino localizar y comprobar —rastrear el pensamiento computacional en tu currículum, en tu decreto autonómico y en la programación de tu centro cuesta cinco horas y media solo en el epígrafe 1.2—; el 2 es el más ligero, con 16,5, porque se trabaja sobre materiales que ya existen; el 3 es el más caro, con 29, porque ahí se fabrica lo que después entra en clase, y la secuencia desenchufada del epígrafe 3.1 se rehace dos veces; y el 4 suma 21. Tampoco valen lo mismo dos epígrafes del mismo bloque.

Cada aplicación práctica dice qué produce, cómo se llama el archivo que te queda y cuánto cuesta en tiempo. Y lleva una etiqueta que dice de quién depende hacerla. Las que dicen **lo haces solo** se hacen con lo que ya tienes en tu aula y en tu cuaderno, y son la mayoría. Las que dicen **lo trabajas con material público** se apoyan en documentos que cualquiera puede leer: el BOE y el boletín de tu comunidad, el currículum de tu etapa, el catálogo público de un proveedor, el banco de tareas Bebras, un reto STEAM publicado por otro centro. Y tres dicen **lo pides y esperas**, porque empiezan por una petición a alguien de tu propio centro: el proyecto educativo y la programación de tu equipo o departamento, en el epígrafe 1.2; el inventario del material que custodia otra persona, en el epígrafe 3.4; y el acuerdo de coordinación con quien da el curso anterior y el siguiente, en el epígrafe 3.5. En las tres, la petición es parte del trabajo, y la parte que no depende de la respuesta puedes adelantarla con lo que ya tienes.

Ninguna de las sesenta aplicaciones prácticas exige comprar nada, y esto no es una promesa blanda. Cincuenta y cinco se resuelven con papel, con el currículum de tu comunidad y con los documentos de tu centro. Las cinco que pueden apoyarse en robot o en tableta —el banco de retos y el protocolo de sesión del epígrafe 3.3, la versión ampliada del reto del 3.4, el plan de subida del 3.2 y la sesión observada del 4.2— llevan escrita su alternativa desenchufada con el mismo detalle y con el mismo tiempo asignado: el banco de retos se hace para el alumno-robot si no hay robot, el protocolo se cierra con la contingencia en papel, el reto vive en su versión de coste cero y la sesión que se observa puede ser la secuencia desenchufada del 3.1. Quien no tenga aparato hace el curso entero sin quedarse a medias en ningún epígrafe.

Cada fuente se cita con su clase, y la clase va marcada: **[E]** cuando es evidencia empírica, **[R]** cuando es una revisión o un marco, y **[P]** cuando es recomendación profesional de este curso. Buena parte de lo que un docente necesita aquí —cuántos robots hacen falta, cuánto debe durar un turno, cómo se rebaja un reto a coste cero, cuánto tiempo se va en baterías y emparejamientos— **no tiene ningún estudio detrás**, y cuando es así se escribe «este curso sostiene» y no «la investigación demuestra». Hay además una regla que atraviesa el manual entero: ninguna cifra de eficacia se escribe sin decir contra qué se comparó. El manual remite cuando un contenido es de otro curso del catálogo: el formato del proyecto largo es del curso 08; los instrumentos de evaluación, del 07; el diseño de la situación de aprendizaje, del 05; las plataformas, las cuentas y la protección de datos, del 04; las barreras de acceso, del 06; las mecánicas de juego, del 09; las altas capacidades, del 13; la orientación hacia lo técnico y su sesgo por sexos, del 16; cualquier explicación apoyada en hemisferios o pirámides de retención, del 17; y lo propio de la Formación Profesional, del 20.

La normativa y los datos de este manual están verificados a 23 de septiembre de 2026, la normativa contra el texto consolidado del boletín oficial o, donde no ha sido posible, contra una reproducción íntegra que se identifica como tal. La bibliografía del final dice para qué se ha usado cada fuente y en qué epígrafe, y las que no han podido verificarse del todo van en un apartado propio que explica por qué. Este es un terreno que se mueve: los decretos de currículum se modifican, la regulación autonómica sobre dispositivos cambia de un curso a otro y la evidencia se revisa. Comprueba la vigencia antes de apoyar en una norma una decisión de centro.

Cómo se evalúa

Conviene que lo sepas antes de empezar, porque cambia cómo lees el manual.

Solo se evalúan las cuatro pruebas de bloque. Cada una tiene veinte preguntas de opción múltiple y se supera con doce aciertos. Superadas las cuatro, se expide el certificado. No hay nada más.

Las aplicaciones prácticas no se corrigen, no se recogen y no puntúan. No hay que subirlas a ningún sitio ni enviárselas a nadie. Son trabajo tuyo contigo mismo, y por eso cada una termina en un documento con nombre que se queda en tu carpeta. Si te saltas una, no pasa nada administrativamente; lo que pasa es que no tendrás ese documento, y varios se necesitan después. El mapa curricular del epígrafe 1.2 es el punto de partida del anclaje de la secuencia del epígrafe 3.1 y del reto del 3.4; el inventario de cuatro columnas del epígrafe 1.5 decide el escalón desde el que se escribe el plan de subida del 3.2; la secuencia desenchufada del 3.1 es la sesión sobre la que se aplica el plan de observación del 4.2; y las tres decisiones del 4.5 se escriben sobre lo que quedó decidido en el bloque 3.

Hay una tarea final optativa, descrita al cierre del manual: la propuesta completa de pensamiento computacional para tu grupo, con su anclaje curricular, su secuencia, su plan de subida, su reto, su encaje en el calendario y su sistema de evaluación. Es optativa de verdad. No condiciona el certificado, no se entrega y nadie la corrige. Se incluye porque es lo único de este curso que se lleva tal cual a una reunión de ciclo o de departamento, y lo único que otro compañero podría conducir sin ti.

Las pruebas de bloque preguntan por lo que el manual dice, no por lo que se te ocurra ni por lo que circule sobre la robótica educativa. Están construidas con cuidado para que no se aprueben por estrategia: sin claves que se adivinen por la longitud, sin claves que sean la única opción cauta de las cuatro y sin indicaciones que anuncien la respuesta.

Qué necesitas para seguirlo

Poca cosa material, y nada que haya que comprar: ni robot, ni tableta, ni kit, ni licencia. Un procesador de textos, un navegador y acceso a la documentación de tu centro —el proyecto educativo, la programación de tu equipo de ciclo o de tu departamento, el calendario escolar de este curso y tu horario con el grupo—. Te hará falta también el real decreto de enseñanzas mínimas de tu etapa y el decreto de currículo de tu comunidad, que localizarás en el epígrafe 1.2. Papel, tijeras y una impresora para las tarjetas del epígrafe 3.1, y cinta de pintor si quieres marcar la cuadrícula en el suelo del aula.

Y una cosa que no es material y que aquí no es opcional: **un grupo real y un saber básico real sobre los que trabajar.** Casi todas las aplicaciones prácticas parten de ellos —un saber de tu programación de este curso, una actividad que ya pusiste, una sesión que se rompió, una nota que ya pusiste, el armario de tu centro con lo que de verdad arranca esta tarde— y sobre un grupo inventado no se pueden hacer. Elige desde el bloque 1 qué etapa, qué área y qué grupo, y sostén esa elección hasta el final: el curso está diseñado para que el mismo saber básico atraviase los cuatro bloques. Si ahora mismo estás fuera del aula, trabaja sobre un grupo y un área concretos de tu experiencia reciente, con la documentación real de un centro; lo que exige pasar algo en clase —la sesión observada del epígrafe 4.2— tendrá que esperar a que vuelvas.

Dos advertencias. La primera, que varios documentos llevan datos de alumnos concretos: identifícalos con un código o con iniciales y no los saques del centro. La segunda, que nada de lo que escribas sobre un alumno es un juicio sobre él: el registro del error, la pregunta de dos minutos o la previsión de los puntos de rotura describen qué sabe y qué necesita en ese contenido, no lo que es.

No necesitas saber programar, no necesitas haber tocado nunca un robot y no necesitas saber estadística: el manual explica cada cifra que usa y lo que no se puede deducir de ella. Si das clase en Infantil, el bloque 2 te dice qué sostiene esa edad y con qué muestra se ha medido. Y si en tu centro no hay un solo aparato, este manual está escrito para ti antes que para nadie.

Concepto de pensamiento computacional, prescripción curricular y alcance de sus efectos

Salir sabiendo qué te exige tu currículum con la letra de la norma delante, qué se puede afirmar de esta forma de trabajo con la evidencia en la mano y desde qué escalón vas a empezar tú.

- 1 Delimitación del pensamiento computacional frente a la competencia informática
- 2 Presencia del pensamiento computacional en el currículum vigente por etapa educativa
- 3 Evidencia sobre la transferencia a otras áreas y afirmaciones sin respaldo
- 4 Actividades desconectadas: alcance del trabajo sin dispositivos
- 5 Diagnóstico inicial: inventario de recursos, horario y criterios de adquisición

1.1. Delimitación del pensamiento computacional frente a la competencia informática



Qué es pensar computacionalmente y por qué no es «saber informática».

En la reunión de coordinación alguien dice que en el centro «se hace robótica». Si preguntas qué es exactamente, salen cinco respuestas: el aula de informática de quinto, el proyecto de la semana cultural, los robots de suelo del armario, una formación que hizo medio claustro hace dos años y un saber básico de tu programación que nadie sabe de qué área es. Las cinco cosas tienen nombre propio y no significan lo mismo, y mientras se confundan no hay manera de decidir qué se enseña, quién lo enseña ni con qué se compra material. Los cinco términos se separan con una definición escrita, que es la que usará el resto del curso.

1.1.1. Cinco nombres para cinco cosas

Competencia digital es usar herramientas digitales: buscar, valorar información, comunicarse, crear contenido, cuidar los datos propios y ajenos. Su objeto es la herramienta.

Programación es escribir instrucciones que una máquina ejecuta. El informe del JRC de la Comisión Europea (Bocconi y otros, 2016) advierte de que «codificar» y «programar» se usan como sinónimos para nombrar el acto de escribir instrucciones, cuando programar abarca algo más ancho: analizar un problema, diseñar una solución e implementarla.

Robótica educativa es una de las formas de materializar esas instrucciones: en vez de aparecer un resultado en la pantalla, un aparato se mueve por el suelo. Es un soporte, no un método.

STEAM no es ninguna de las anteriores: es un modo de organizar el trabajo integrando áreas —ciencia, tecnología, ingeniería, artes y matemáticas—. Qué cuenta como integración de verdad y qué es decoración se trabaja en el epígrafe 2.4, que es su sitio.

Y **pensamiento computacional** es la forma de abordar un problema: partirlo en trozos, ver qué se repite, decidir qué información sobra y escribir una secuencia de pasos que otro pueda ejecutar sin interpretarla. El JRC lo dice sin rodeos: varios autores lo distinguen de la competencia digital porque su rasgo propio es el foco en los procesos de resolución de problemas y en la creación de soluciones computables. No necesita ninguna máquina para existir, y esa es la frase que sostiene el curso entero.

Cinco cosas distintas que en el centro se llaman igual

Pensamiento computacional: formular el problema y la solución

Competencia digital

- Usar herramientas digitales
- Su objeto es la herramienta
- El JRC la distingue del pensamiento computacional
- El marco docente no incluye el pensamiento computacional

Programación

- Escribir instrucciones que una máquina ejecuta
- Analizar, diseñar e implementar
- «Conceptualizar, no programar» (Wing, 2006)
- Son actividades distintas (JRC, 2022)

Robótica educativa

- Una forma de materializar las instrucciones
- El error se ve en el suelo del aula
- Es un soporte, no un método
- Tener robots no acredita el aprendizaje

STEAM

- Un modo de integrar áreas
- Ciencia, tecnología, ingeniería, artes y matemáticas
- No es una operación de pensamiento
- Qué es integración y qué decoración: 2.4

Fuente: elaboración propia a partir de Bocconi y otros (JRC, 2016 y 2022), Wing (2006 y 2008) y Shute, Sun y Asbell-Clarke (2017).

FIGURA 1.1

1.1.2. De dónde viene el término, y qué dijo exactamente quien lo puso en circulación

La palabra la puso en circulación Jeannette Wing en una columna de opinión de dos páginas y media en *Communications of the ACM* (2006). Allí escribe que el pensamiento computacional «involves solving problems, designing systems, and understanding human behavior, by drawing on the concepts fundamental to computer science», y que es «a fundamental skill for everyone, not just for computer scientists». En el apartado en que dice qué es y qué no es, la primera línea es la que más falta hace en un claustro: *conceptualizing, not programming*. Conceptualizar, no programar. Y añade que la informática no es escribir programas.

Antes de citarlo, mira qué es ese texto: un manifiesto breve en una sección de opinión. No es una definición operativa ni trae ninguna lista de operaciones; esa llegó después y por otras fuentes. La propia Wing lo reformuló en 2008 en *Philosophical Transactions of the Royal Society A*, y lo acotó bastante: dice que la esencia del pensamiento computacional es la abstracción y que la computación es la automatización de nuestras abstracciones. La definición operativa que hoy usa la Comisión Europea la formuló Wing en 2011: los procesos de pensamiento que intervienen en formular problemas y sus soluciones, de modo que la solución se represente de una forma que un agente de procesamiento de información pueda ejecutar. Se cita a través del informe del JRC de 2016, que es quien la adopta, porque la nota original ya no se sirve.

1.1.3. La crítica que hay que conocer antes de defenderlo

En 2017, en la misma revista que publicó a Wing, Peter Denning firmó la objeción que ningún material escolar suele contar. Sostiene que, por el afán de incluir a todos los campos que usan computación, las definiciones recientes se volvieron difusas y prometieron de más, y que la afirmación general —que esto beneficia a todo el mundo— «has never been substantiated with empirical research». Su conclusión es literal y se lee entera: el pensamiento computacional beneficia sobre todo a quien diseña computaciones, y los beneficios para quien no diseña no están sustanciados [R].

No es una razón para no enseñarlo; es la razón para no venderlo. Y no es una voz aislada. El informe del JRC de 2016 abre reconociendo que falta consenso sobre la definición. El de 2022, seis años y treinta países después, sigue escribiendo que no hay acuerdo sobre una definición de referencia, y que en los centros se usan indistintamente pensamiento computacional, programación y codificación, siendo actividades distintas. Lo que transfiere y lo que no transfiere esto a otras áreas, con sus cifras y sus límites, es el epígrafe 1.3.

ATENCIÓN · DOS FRASES QUE SE CAEN SOLAS

«Pensar computacionalmente enseña a pensar mejor en general» y «esto es alfabetización del siglo XXI, como leer y escribir». La primera no está sustanciada empíricamente, y así lo escribe Denning. La segunda procede del manifiesto de Wing de 2006, que es una columna de opinión, no un resultado. Si tu defensa del proyecto descansa en cualquiera de las dos, no descansa en nada.

1.1.4. La definición con la que trabaja este curso

Aquí se trabaja con una definición operativa, y se dice de dónde sale y por qué: la elige este curso, no la respalda un consenso del campo, porque el consenso no existe. Es la de Shute, Sun y Asbell-Clarke (2017), que definen el pensamiento computacional como el fundamento conceptual necesario para resolver problemas de forma eficaz y eficiente —algorítmicamente, con o sin ayuda de ordenadores— con soluciones reutilizables en contextos distintos [R]. Se elige por dos motivos. El primero, ese inciso: es la única de las tres definiciones disponibles que dice expresamente que la máquina puede no estar. El segundo, que mete la depuración dentro del propio constructo, y no fuera como un contratiempo del aula.

Que las listas de operaciones no coinciden entre sí ya se ve aquí: Selby y Woollard (2013) hablan de abstracción, descomposición, diseño algorítmico, evaluación y generalización, sin reconocer el patrón como operación aparte; Brennan y Resnick (2012) reparten siete conceptos, cuatro prácticas y tres perspectivas. El 2.1 se ocupa de eso.

1.1.5. Qué cambia en tu centro cuando los cinco términos quedan separados

Cambian tres cosas. La primera, de quién es esto: si lo que se hace es competencia digital, el marco es uno; si es pensamiento computacional, vive en el currículo de tu etapa y es de las áreas que lo tienen prescrito, que es lo que verás en el epígrafe 1.2 con la letra de la norma delante. La segunda, qué se compra: un armario con seis robots no acredita que en el centro se trabaje el pensamiento computacional, igual que una biblioteca no acredita comprensión lectora. La tercera, qué se dice fuera: cuando alguien pregunte qué hacéis, la respuesta útil no es «damos robótica», sino «trabajamos descomposición y algoritmos en tal área, y el robot es el soporte que usamos en esta parte».

NORMATIVA · EL PENSAMIENTO COMPUTACIONAL NO ESTÁ EN EL MARCO DE REFERENCIA DE LA COMPETENCIA DIGITAL DOCENTE

El MRCDD, publicado por Resolución de 4 de mayo de 2022 (BOE núm. 116, de 16 de mayo de 2022), excluye expresamente el pensamiento computacional, y lo razona: «Se ha optado por no incluir el ítem 1.9. Pensamiento computacional del S4T 2020, que tampoco estaba incluido de forma explícita en el DigCompEdu, en el MRCDD, pues se considera que se trata de un conjunto de estrategias de razonamiento vinculadas con la computación y la programación, pero que no es una función docente». Lo que ese marco regula es tu competencia digital; el pensamiento computacional del alumnado vive en el currículo, no ahí.

Aplicación práctica

1. **Audita el vocabulario de tu centro.** Recoge en una tabla las cinco o seis veces que en el último curso se dijo «robótica», «programación» o «digitalización» en tu centro —actas, programación de aula, la web, el cartel de la semana cultural—: **lo haces solo**. A cada una ponle al lado cuál de los cinco términos la describe de verdad y quién se hace cargo de ella. Guárdalo como [auditoria-de-vocabulario.md](#). **Una hora.**
2. **Escribe tu definición operativa.** Una página: qué vas a llamar tú pensamiento computacional este curso, con la fuente de la que la tomas, y dos listas debajo —lo que incluye y lo que deja fuera—: **lo haces solo**. En «deja fuera» tienen que aparecer por escrito el manejo de herramientas y el aparato. Pruébala contra tres actividades que ya hagas: si las tres entran, la definición es demasiado ancha. Guárdalo como [definicion-operativa.md](#). **Una hora y media.**
3. **Revisa las promesas de un material.** Coge la web de tu centro o el material de una formación sobre robótica que tengas a mano y copia sus tres afirmaciones más fuertes: **lo trabajas con material público**. Al lado de cada una escribe si se sostiene, y con qué. Guárdalo como [promesas-revisadas.md](#). **Una hora.**

1.2. Presencia del pensamiento computacional en el currículo vigente por etapa educativa

Dónde te lo pide el currículo, con qué palabras y en qué curso.

La pregunta que casi nadie contesta con precisión es la más simple: ¿qué estás obligado a dar? En los claustros se responde de memoria, y la memoria falla justo donde importa, que es en las palabras. No es lo mismo que la norma diga *interpretación* que *reconocimiento*, ni que separe dos recursos con una «o» o con una «y»: de eso depende si puedes cerrar el curso sin haber enchufado nada. El mapa por etapas se lee con la letra delante, y lo que la norma calla obliga tanto como lo que dice, hasta tu decreto autonómico y tu propia programación.

1.2.1. Infantil: programar secuencias, y tres palabras que no aparecen

El Real Decreto 95/2022, de 1 de febrero (BOE núm. 28, de 2 de febrero de 2022), lo sitúa en el área «Descubrimiento y Exploración del Entorno», competencia específica 2 del segundo ciclo, que une las destrezas del pensamiento computacional a los procedimientos del método científico. Su criterio de evaluación 2.5 lo concreta: «Programar secuencias de acciones o instrucciones para la resolución de tareas analógicas y digitales, desarrollando habilidades básicas de pensamiento computacional».

Lo que no está también obliga. En el RD 95/2022 no aparecen las palabras *algoritmo*, *robótica* ni *actividades desenchufadas*, y nadie puede exigirte en Infantil un robot ni programación por bloques apoyándose en esta norma. Y un segundo matiz que conviene tener localizado: por su disposición final primera, los criterios de evaluación y los saberes básicos del primer ciclo (0-3) carecen de carácter de normativa básica y son orientativos [R].

1.2.2. Primaria: dos áreas, dos redacciones y la conjunción que decide

El Real Decreto 157/2022, de 1 de marzo (BOE núm. 52, de 2 de marzo de 2022), lo reparte en dos sitios. En **Matemáticas** vive en el bloque «D. Sentido algebraico», y la norma escribe la progresión con tres verbos: primer ciclo, «Estrategias para la interpretación de algoritmos sencillos»; segundo ciclo, «Estrategias para la interpretación y modificación de algoritmos sencillos»; tercer ciclo, «Estrategias para la interpretación, modificación y creación de algoritmos sencillos». Interpretar, modificar y crear, en ese orden y en ese ciclo. La programación por bloques y la robótica educativa aparecen por primera vez en el segundo ciclo, y siempre dentro de un paréntesis de ejemplos: pedir a un alumno de primer ciclo que cree un algoritmo es pedir lo que el currículo no pide.

En **Conocimiento del Medio Natural, Social y Cultural** está la competencia específica 3: «Resolver problemas a través de proyectos de diseño y de la aplicación del pensamiento computacional, para generar cooperativamente un producto creativo e innovador que responda a necesidades concretas». Y en sus saberes del bloque B está la frase que sostiene este curso entero, con una conjunción que es del legislador y que no se convierte en otra.

NORMATIVA · LOS DOS LITERALES QUE TIENES QUE COPIAR BIEN

Real Decreto 157/2022, anexo II. Matemáticas, saberes del bloque D, segundo ciclo: «Estrategias para la interpretación y modificación de algoritmos sencillos». Conocimiento del Medio Natural, Social y Cultural, bloque B, segundo ciclo: «Iniciación en la programación a través de recursos analógicos (actividades desenchufadas) o digitales (plataformas digitales de iniciación en la programación, aplicaciones de programación por bloques, robótica educativa...)».

La norma pone primero el recurso analógico, nombra expresamente las actividades desenchufadas y ofrece lo digital como alternativa, no como complemento ni como requisito. Arriba está además el descriptor CD5 del perfil de salida, que sí es normativa básica: el alumnado

«se inicia en el desarrollo de soluciones digitales sencillas y sostenibles».

1.2.3. ESO y Bachillerato: dónde está garantizado y dónde es elección

El Real Decreto 217/2022, de 29 de marzo (BOE núm. 76, de 30 de marzo de 2022), lo lleva a la competencia específica 4 de Matemáticas y, sobre todo, a un bloque entero de saberes de Tecnología y Digitalización titulado «Pensamiento computacional, programación y robótica».

Aquí está lo que más se confunde. El artículo 8.3 dice: «En al menos uno de los tres cursos, todo el alumnado cursará la materia Tecnología y Digitalización». En uno; en cuál y con cuántas horas lo decide cada comunidad. Y Digitalización de 4.º es materia de elección por el artículo 9.2. En ESO, por tanto, esto está garantizado para todos una sola vez en tres años y lo demás depende de la optatividad. En Bachillerato, el Real Decreto 243/2022, de 5 de abril (BOE núm. 82, de 6 de abril de 2022), establece Tecnología e Ingeniería I y II: no hay materia «TIC» estatal, y donde exista es creación autonómica.

1.2.4. Qué anexo obliga, qué anexo no, y dónde está el horario

Dos precisiones que ahorran discusiones. La primera: los anexos I y II —perfil de salida, criterios y saberes— son normativa básica; el anexo III, el de orientaciones para el diseño de situaciones de aprendizaje, «carece del carácter de normativa básica», y lo dice la disposición final primera de los tres reales decretos. Un centro no puede descartar el saber de algoritmos, pero sí diseñar la situación de aprendizaje como quiera.

La segunda: el horario no está en el anexo III, sino en el **anexo IV**, al que remiten el artículo 12.1 del RD 157/2022, el 14.1 del RD 217/2022 y el 19 del RD 243/2022. Y no asigna ni una hora al pensamiento computacional como tal: sale de las áreas que ya lo llevan dentro. Para tu comunidad queda el resto: las enseñanzas mínimas «requerirán el 60 por ciento de los horarios escolares para aquellas comunidades autónomas que no tengan lengua cooficial, y el 50 por ciento para las que la tengan» (art. 11.3 del RD 157/2022). Ese 40 % —o 50 %— es donde caben las materias autonómicas y la optatividad.

De la norma estatal a tu programación



FIGURA 1.2

1.2.5. Tu decreto autonómico: cómo se localiza y qué dos cosas se le pregunta

Busca en el boletín oficial de tu comunidad el decreto de currículo de tu etapa, comprueba que es el vigente y no uno derogado, y hazle dos preguntas: qué horario asigna al área que lleva dentro el pensamiento computacional y qué optativas de programación o robótica ofrece. Dos ejemplos de la dispersión que hay, y solo son ejemplos. En Andalucía, la Orden de 30 de mayo de 2023 (BOJA núm. 104, de 2 de junio de 2023) incluye en su artículo 5 «Computación y Robótica» entre las optativas propias de 1.º, 2.º y 3.º de ESO. En la Comunidad de Madrid, el Decreto 65/2022, de 20 de julio (BOCM de 26 de julio de 2022), tiene «Ciencias de la Computación» como optativa de 1.º y 2.º, y da a Tecnología y Digitalización tres horas semanales en 2.º y dos en 3.º. En las demás, el decreto equivalente. Y un aviso: «Tecnología, Programación y Robótica» es denominación LOMCE y ya no aparece en el decreto madrileño vigente.

1.2.6. La cadena hasta tu programación, y una casilla que no vas a encontrar

De la norma estatal al decreto autonómico, del decreto al proyecto educativo del centro y de ahí a la programación de tu equipo de ciclo o de tu departamento: esa cadena tienes que poder recorrerla entera. La autonomía pedagógica del artículo 120 de la Ley Orgánica 2/2006 permite al centro decidir cuándo y cómo entra el pensamiento computacional y con qué materiales; no permite decidir **si** entra, porque los saberes básicos son normativa básica.

ATENCIÓN · BUSCAR EN EL MARCO DOCENTE LA CASILLA QUE TE OBLIGA

El Marco de Referencia de la Competencia Digital Docente no tiene ninguna competencia de pensamiento computacional, y el epígrafe 1.1 recoge con la letra por qué decidió no incluirlo. Además no es un reglamento, sino un Acuerdo de la Conferencia Sectorial de Educación publicado por resolución. Lo que te obliga está en el currículo del alumnado, no ahí.

Aplicación práctica

1. **Levanta el mapa curricular de tu etapa.** Abre el real decreto de enseñanzas mínimas que te corresponde y localiza el área, la competencia específica, los criterios y los saberes básicos que te obligan a trabajar el pensamiento computacional: **lo trabajas con material público**. Cópialos con la letra exacta, indicando anexo y ciclo o curso, y marca cuáles son de tu curso y cuáles del anterior y del siguiente. Guárdalo como [mapa-curricular-de-mi-etapa.md](#). **Una hora y media.**
2. **Ficha tu decreto autonómico.** Localiza en el boletín oficial de tu comunidad el decreto de currículo vigente de tu etapa: **lo trabajas con material público**. Anota su identificación completa y su fecha, comprueba que no está derogado, extrae el horario del área que lleva dentro el pensamiento computacional y lista las optativas de programación o robótica con su curso y sus requisitos. Guárdalo como [mi-decreto-autonomico.md](#). **Dos horas y media.**
3. **Recorre la cadena dentro de tu centro.** Pide el proyecto educativo y la programación didáctica de tu equipo de ciclo o de tu departamento: **lo pides y esperas**. Busca en ambos dónde aparece el pensamiento computacional, copia lo que digan y contrástalo con los dos documentos anteriores; anota los tres desajustes más claros entre lo que la norma pide y lo que el centro tiene escrito. Guárdalo como [cadena-en-mi-centro.md](#). **Una hora y media.**

1.3. Evidencia sobre la transferencia a otras áreas y afirmaciones sin respaldo



Qué transfiere de verdad esto a otras áreas, y qué se está prometiendo de más.

Estás en la reunión en la que se decide si el proyecto de robótica sigue el curso que viene. Alguien dice que programar enseña a pensar, que mejora las Matemáticas y que hasta ayuda a leer. Otro contesta que eso no está demostrado. Los dos hablan de memoria, y se llevará la decisión el que hable más alto. Este epígrafe existe para que no seas ninguno de los dos: te da las cifras enteras, con lo que midió cada estudio, contra qué grupo lo comparó y qué limitación declaran sus propios autores. No sirve para vender el proyecto: sirve para defenderlo por lo que es.

A lo largo del manual verás tres etiquetas detrás de las fuentes. [E] es evidencia empírica: alguien lo midió. [R] es revisión o marco: alguien ordenó lo que otros midieron. [P] es recomendación profesional: nadie lo ha medido y se dice.

1.3.1. Qué preguntó el metaanálisis de transferencia y qué contestó

La fuente central es el metaanálisis de Scherer, Siddiq y Sánchez Viveros (2019), publicado en el *Journal of Educational Psychology*: un metaanálisis de tres niveles y efectos aleatorios sobre 105 estudios y 539 tamaños de efecto, medidos en g de Hedges [E]. Encuentra un efecto global de transferencia moderado, $g = 0,49$, IC 95 % [0,37, 0,61], y lo reparte en dos: transferencia cercana —a lo que se parece a programar— $g = 0,75$ [0,39, 1,11], y transferencia lejana $g = 0,47$ [0,35, 0,59]. Dentro de la lejana, los beneficios se concentran en pensamiento creativo (0,73), destrezas matemáticas (0,57), metacognición (0,44), destrezas espaciales (0,37) y razonamiento (0,37). Y abajo, justo lo que más se promete fuera: el rendimiento escolar se beneficia poco ($g = 0,28$ [0,14, 0,42]) y en competencia lectora el efecto es indistinguible de cero (IC 95 % [-0,12, 0,08]); los autores escriben que no encontraron apoyo para una transferencia positiva a la lectura. Declaran dos límites: excluyeron muestras de alumnado con dificultades específicas o condiciones clínicas e incluyeron solo estudios publicados en inglés.

1.3.2. La cifra que ordena el curso: contra qué se compara

En el mismo metaanálisis hay un análisis de moderadores que vale más que el efecto global. Cuando el grupo de control no hace nada especial, programar da $g = 0,65$ [0,51, 0,78]. Cuando el grupo de control hace otra actividad equivalente y bien diseñada, $g = 0,16$ [0,01, 0,33] [E]. Cuatro veces menos: buena parte de lo que se atribuye a programar es lo que aporta cualquier actividad buena puesta en el sitio de la nada. A eso se suma el sesgo de publicación, medido: la literatura publicada da 0,60 [0,45, 0,75] y la gris 0,34 [0,15, 0,52]. El efecto no desaparece, pero está inflado. Los mismos autores, en un artículo de opinión de 2021, añaden dos precisiones: la transferencia no ocurre sola, requiere un diseño instruccional orientado a ella, y con los diseños no aleatorizados que dominan esta literatura no caben conclusiones estrictamente causales [R]. Se dice «se asoció con», no «causó».

1.3.3. Del pensamiento computacional a las áreas STEM: moderadores completos, cifras ninguna

La revisión sistemática con metaanálisis de Li y Oon (2024), en *International Journal of STEM Education*, trabaja sobre 37 estudios, 7.832 estudiantes y 96 tamaños de efecto, e identifica 36 beneficios [E]. Sus efectos se describen como moderados en el global, en la transferencia cercana y en la lejana, y son mayores sobre lo cognitivo que sobre lo no cognitivo —actitudes, motivación, autoeficacia—. Aquí no se escribe ninguna cifra, y la razón se dice: los valores numéricos no se han podido verificar contra el artículo completo. Lo verificado son los moderadores, que es la parte útil: para el efecto global y para la transferencia cercana resultan significativos cuatro —nivel

educativo, tamaño muestral, estrategia instruccional y duración de la intervención—; para la transferencia lejana, solo dos —nivel educativo y tamaño muestral—. Traducido a tu programación: cómo diseñas la sesión y cuánto dure mueve lo cercano, y no mueve lo lejano.

1.3.4. Los dos metaanálisis de robótica de 2024 recortan cosas distintas

Circulan dos cifras de 2024 que parecen contradecirse y no lo hacen. Hong (2024) mide robots educativos frente a enseñanza sin robots, con el pensamiento computacional como único resultado, sobre 27 estudios: $SMD = 0,558$ [E]. Ouyang y Xu (2024) mide la robótica dentro de la educación STEM, sobre 21 estudios y 30 tamaños de efecto, con tres resultados: efecto global $g = 0,488$ [0,094, 0,882], rendimiento de aprendizaje 0,665, actitudes 0,497 y, sobre pensamiento computacional, $g = 0,079$, $p = 0,434$, IC 95 % [-0,119, 0,277]: no significativo [E]. Lo que separa un resultado del otro no es el aparato: es el recorte, el diseño de la actividad y aquello con lo que se compara. Ouyang y Xu probaron ocho moderadores y solo salió significativo uno, la disciplina; la condición del grupo de control **no** resultó significativa, de modo que el argumento de la comparación se apoya en Scherer y no aquí. Los autores explican su resultado nulo: el pensamiento computacional no se desarrolla por el mero uso de robots, y recomiendan combinarlos con problemas mal estructurados.

ATENCIÓN · EL 0,558 NUNCA SE ESCRIBE SOLO

Es la cifra que aparece en las presentaciones de proveedor, y sola induce a error: su intervalo de confianza no está verificado y procede de un estudio que solo mide pensamiento computacional frente a no usar robots. Va siempre junto al 0,079 no significativo de Ouyang y Xu. Y hay un segundo salto que no se da: la robótica mejora las **actitudes** con $g = 0,497$, IC 95 % [0,023, 1,017], efecto moderado y muy impreciso. Actitudes no son vocaciones: ninguna de estas fuentes mide elección de itinerario ni carrera. Para la orientación hacia lo técnico y su sesgo por sexos, el propietario es el curso 16.

1.3.5. El dato español: positivo, pequeño y con su letra pequeña

La Escuela de Pensamiento Computacional e Inteligencia Artificial del INTEF publicó en 2022 los resultados de la edición 20/21, con 18 comunidades y ciudades autónomas. En el Nivel I —Infantil y primeros cursos de Primaria, con **actividades desenchufadas**— 119 docentes completaron la implementación y se midió a 2.580 estudiantes, 1.824 experimentales y 756 de control, con más del 80 % de las propuestas en cinco o seis sesiones. Los tamaños del efecto: $d = 0,315$ en Infantil de cinco años, 0,156 en 1.º de Primaria, 0,185 en 2.º y 0,228 en 3.º [E]. Los autores concluyen un efecto positivo sobre el desarrollo del pensamiento computacional «aunque el tamaño del efecto es pequeño», y declaran ellos mismos tres limitaciones: diseño solo post-test, grupos no aleatorios y sin prueba previa que permita conocer el nivel de partida. La buena noticia y la advertencia van en la misma frase: cinco o seis sesiones sin un solo aparato, hechas por docentes corrientes en aulas corrientes, mueven algo, y ese algo es pequeño y no se puede atribuir con seguridad a la intervención.

Las cuatro cifras que ordenan este curso

g = 0,49 Transferencia global de aprender a programar 105 estudios y 539 tamaños de efecto; IC 95 % [0,37, 0,61]	0,65 → 0,16 El mismo efecto contra no hacer nada y contra otra actividad Control no tratado frente a control activo; cuatro veces menos	g = 0,079 Robótica sobre pensamiento computacional dentro de STEM No significativo (p = 0,434); 21 estudios, 30 tamaños de efecto	d = 0,315 Infantil de 5 años tras 5 o 6 sesiones desenchufadas en España Solo post-test, grupos no aleatorios y sin pretest, según sus autores
---	---	---	--

Fuente: elaboración propia a partir de Scherer, Siddiq y Sánchez Viveros (2019), Ouyang y Xu (2024) e INTEF (2022).

FIGURA 1.3

1.3.6. Qué puedes decir y qué no

Con estos números se puede decir esto: programar transfiere, transfiere poco, transfiere mejor a lo que se le parece, transfiere a matemáticas y prácticamente nada a la lectura, y transfiere cuatro veces menos cuando se compara con hacer otra cosa buena. No se puede decir que enseñe a pensar en general, ni que mejore el rendimiento escolar, ni que ayude a leer, ni que el robot desarrolle por sí solo el pensamiento computacional. Este curso sostiene [P] que la consecuencia es de diseño y no de catálogo: si tu propuesta tiene que valer la pena frente a otra buena actividad, y no frente a no hacer nada, lo que la sostiene es el problema que plantea y no el material que usa. Para la conversación con dirección y familias, el epígrafe 4.5; para lo que se puede afirmar del desenchufado, el epígrafe 1.4.

ATENCIÓN · LA TRAMPA DE CITAR SOLO EL EFECTO GLOBAL

Decir «g = 0,49» y parar es la forma elegante de prometer de más. Sin la condición de comparación delante, esa cifra describe un mundo en el que el grupo de control no hacía nada.

Aplicación práctica

1. **Redacta la ficha de lectura del metaanálisis de transferencia.** Sobre los datos de 1.3.1 y 1.3.2, escribe una ficha de una página con cinco entradas: qué se midió, en qué etapas, contra qué condición de comparación, qué cifra corresponde a cada condición y qué limitaciones declaran los autores: **lo trabajas con material público**. Añade una sexta entrada propia: qué frase de las que se dicen en tu centro queda desmentida ahí. Guárdalo como **ficha-transferencia.md**. **Una hora y media.**
2. **Redacta la ficha de lectura de los dos metaanálisis de robótica.** En dos columnas, una por estudio, anota qué mide cada uno, sobre cuántos estudios, con qué resultado y qué moderador resultó significativo: **lo trabajas con material público**. Cierra con tres líneas que expliquen por qué las dos cifras conviven sin contradecirse. Guárdalo como **ficha-robotica.md**. **Una hora y media.**
3. **Escribe tu respuesta a «programar enseña a pensar».** Media página, dirigida a un compañero de claustro real y con sus palabras, no con las tuyas: **lo haces solo**. Tiene que contener la cifra global, el par 0,65 frente a 0,16 y lo que pasa con la lectura, y tiene que terminar sin prometer nada. Léela en voz alta antes de guardarla. Guárdalo como **respuesta-programar-ensena-a-pensar.md**. **Una hora.**

1.4. Actividades desconectadas: alcance del trabajo sin dispositivos

Qué puedes enseñar sin un solo aparato, y hasta dónde llega el papel.

Esta es la pregunta que decide si el curso te sirve. En tu centro puede que no haya robots, o que los haya sin pilas, o que el aula de informática esté ocupada de octubre a junio. Y aun así, el saber básico está en tu programación y hay que darlo. Son dos preguntas distintas y no se mezclan: hasta dónde llegas sin enchufar nada —bastante más lejos de lo que se dice, y con datos delante— y dónde está el techo del papel, que se conoce antes de montar la secuencia. Ninguna de las dos respuestas es una consigna a favor ni en contra del aparato.

1.4.1. Lo primero no es la evidencia: es la letra de la norma

Antes de discutir qué funciona hay que saber qué se te pide. En Primaria, el saber básico de Conocimiento del Medio del segundo ciclo que copiaste en el epígrafe 1.2 ofrece los recursos analógicos —y los llama actividades desenchufadas— o los digitales. La norma no te obliga a lo digital: admite el papel como vía suficiente.

En Matemáticas ocurre lo mismo por otro camino. El saber del segundo ciclo pide interpretar y modificar algoritmos sencillos, y mete la programación por bloques y la robótica educativa en un paréntesis de ejemplos: son una manera de darlo, no el contenido. En Infantil el criterio 2.5 del segundo ciclo pide «Programar secuencias de acciones o instrucciones para la resolución de tareas analógicas y digitales, desarrollando habilidades básicas de pensamiento computacional», y en esa norma no aparecen las palabras *algoritmo*, *robótica* ni *actividades desenchufadas*. En ESO, el bloque de Tecnología y Digitalización se titula «Pensamiento computacional, programación y robótica»: ahí el aparato sí está en el título, y eso pesa, como se verá al final.

ATENCIÓN · «EL CURRÍCULO OBLIGA A COMPRAR MATERIAL»

No lo dice. El saber de Conocimiento del Medio que copiaste en el epígrafe 1.2 ofrece los recursos analógicos o los digitales, con la conjunción del legislador, y nombra primero los analógicos; en Matemáticas, la programación por bloques y la robótica educativa van dentro de un paréntesis de ejemplos. Quien te diga lo contrario no ha leído el saber básico entero.

1.4.2. Lo que se ha medido del desenchufado, con su base debajo

Hay un metaanálisis que se cita mucho y casi siempre a medias. Chen y otros (2023) revisaron cuarenta y nueve estudios publicados entre 2006 y 2022 y llevaron trece de ellos, con dieciséis tamaños de efecto, al metaanálisis: $g = 1,028$, con un intervalo de confianza del 95 % que va de 0,641 a 1,415. Es un efecto grande [E]. Y es una cifra que no sirve para planificar, por dos razones que van juntas: la base, trece estudios, y la amplitud del intervalo, que recorre desde lo moderado hasta lo muy grande. Un intervalo así permite esperar algo, no dimensionar nada. Los autores declaran además que se limitaron a estudios en inglés y a determinadas bases de datos. Como contraste doméstico están los datos del INTEF que viste en el epígrafe 1.3: efectos de 0,156 a 0,315 con actividades desenchufadas, positivos y pequeños, y con tres limitaciones que declaran los propios autores. Un metaanálisis de trece estudios en inglés y un informe español de aulas corrientes apuntan a lo mismo por caminos distintos.

Y hay algo que la literatura no confirma. Huang y Looi (2021) revisaron cuarenta publicaciones, y en la promesa de que lo desenchufado es la vía de acceso equitativa a la informática no encuentran una confirmación, sino zonas de desconocimiento y tensión [R]. Empezar sin aparatos resuelve un problema de material; no resuelve por sí solo un problema de equidad.

1.4.3. Las dos cifras de 2025 que nadie puede restar

El estudio más citado ahora es el metaanálisis comparativo de Wang, Fan y Yu (2025), con treinta y siete estudios publicados entre enero de 2006 y marzo de 2025. Da dos efectos: las actividades enchufadas, $g = 0,606$; las desenchufadas, $g = 0,501$ [E]. Cada uno está calculado **contra su propio grupo de control**. No es una comparación directa entre las dos formas de trabajar, no procede de una prueba de superioridad y, por tanto, no autoriza a decir que una gane a la otra. La diferencia entre ambas no es un resultado: es la distancia entre dos mediciones distintas.

Lo que sí es resultado de ese trabajo, y es lo que ordena el diseño de tus sesiones, es el análisis de moderadores. En Educación Primaria, en materias STEM y en intervenciones de más de cuatro semanas, lo desenchufado y lo enchufado son equivalentes. Lo enchufado rinde más en Secundaria y en la universidad, y en intervenciones cortas. Y lo desenchufado rinde más en un único formato: el aprendizaje basado en juego, el terreno de los juegos de mesa y de cartas, que son las herramientas desenchufadas más frecuentes en los estudios de Chen y otros.

Qué dice el metaanálisis de 2025, moderador a moderador

	Actividades desenchufadas	Actividades enchufadas
Efecto frente a su propio control	$g = 0,501$	$g = 0,606$
Educación Primaria	Equivalente	Equivalente
Secundaria y universidad	Sin ventaja	Con ventaja
Intervención de más de cuatro semanas	Equivalente	Equivalente
Intervención de menos de cuatro semanas	Sin ventaja	Con ventaja
Materias STEM	Equivalente	Equivalente
Aprendizaje basado en juego	Con ventaja	Sin ventaja

Fuente: Wang, Fan y Yu (2025), metaanálisis de 37 estudios (2006-2025). Cada efecto está calculado contra su propio grupo de control: no es una comparación directa entre ambas condiciones.

FIGURA 1.4

ATENCIÓN · LA RESTA QUE NO SE PUEDE HACER

«0,606 menos 0,501, luego el aparato aporta una décima»: esa frase no se sostiene, y es la que vas a oír. Son dos efectos medidos cada uno contra su propio control, con estudios y alumnado distintos. Si alguien la usa a favor del robot, sobreinterpreta; si la usas tú a favor del papel, haces lo mismo al revés. Del estudio se citan los moderadores, no la resta.

1.4.4. Empezar por el papel no cuesta comprensión

Queda la pregunta de orden, la que decide tu primera unidad: ¿se pierde algo por empezar sin enchufar? Hermans y Aivaloglou (2017) hicieron un experimento controlado con asignación aleatoria y treinta y cinco niños de ocho a doce años de un solo centro, durante ocho semanas: un grupo empezó con un entorno de programación por bloques y el otro con materiales desenchufados, y desde la quinta semana ambos trabajaron con el entorno digital. Al final no hubo diferencia en comprensión de conceptos. Sí la hubo en autoeficacia, a favor del grupo que empezó desenchufado, y en la variedad de bloques distintos que ese grupo usó por su cuenta, incluidos algunos que las sesiones no habían tratado [E]. Treinta y cinco niños de un colegio: la muestra va delante de la conclusión, y los propios autores advierten de que puede no ser representativa.

Kırçali y Özdener (2023) trabajaron con ciento nueve alumnos de sexto durante seis semanas en tres condiciones —plataforma web, herramientas desenchufadas y entorno por bloques—. El único grupo que mejoró respecto de sí mismo fue el desenchufado, pero entre los tres grupos no aparecieron diferencias significativas [E]. Ni gana el papel ni da igual: con esa muestra y esa duración, el estudio no separa las tres condiciones.

1.4.5. Dónde está el techo, y cómo se llama

La conclusión operativa es explícita: en Primaria se puede llegar lejos sin equipo, siempre que la secuencia sea de tres a cinco sesiones repartidas en cuatro o cinco semanas, que es como se reparte en el epígrafe 3.5; en Secundaria el papel es el punto de partida, no el destino. Y hay un techo que ninguna cifra mide, así que se etiqueta como lo que es: este curso sostiene que, sin una máquina que ejecute literalmente lo escrito y no lo que querías escribir, la depuración auténtica no tiene dónde ocurrir [P]. Nadie ha comparado la depuración en los dos entornos: por eso es un razonamiento y no un hallazgo. Qué se gana y qué se pierde en cada escalón se desarrolla en el epígrafe 2.3; cómo se sube, con el material que haya, en el epígrafe 3.2.

Aplicación práctica

1. **Lista los saberes de tu programación que puedes trabajar sin ningún dispositivo.** Abre tu programación y el currículo de tu etapa, y copia con la letra de la norma cada saber básico y cada criterio relacionado con el pensamiento computacional que te corresponda este curso: **lo trabajas con material público.** Marca cada uno con una de tres etiquetas —se trabaja entero sin aparato, se trabaja en parte, necesita máquina— y anota al lado, en una línea, con qué material lo harías. Guárdalo como [saberes-sin-dispositivo.md](#). **Una hora y media.**
2. **Escribe dónde está tu techo y por qué.** Sobre la lista anterior, quédate con los saberes marcados «necesita máquina» o «en parte» y redacta media página por cada uno: **lo haces solo.** Di qué es exactamente lo que el papel no sostiene —ejecución literal, error visible, depuración sobre una máquina que no interpreta— y distingue lo que está medido de lo que es razonamiento tuyo. Guárdalo como [techo-del-papel.md](#). **Una hora.**
3. **Redacta la nota de una página para el claustro.** Una página con las cifras y sus límites: **lo haces solo.** Incluye el efecto de Chen y otros (2023) con sus trece estudios y su intervalo, los datos del INTEF con sus limitaciones declaradas, y las dos cifras de 2025 con la advertencia de que cada una va contra su propio control y no se restan. Cierra con los moderadores, que son lo que decide tu programación. Guárdalo como [nota-desenchufado-claustro.md](#). **Una hora.**

1.5. Diagnóstico inicial: inventario de recursos, horario y criterios de adquisición



Desde qué escalón empiezas tú: inventario, horas y filtro de compra.

En el armario de tu centro hay material comprado con una subvención, usado una semana y guardado. No falló el aparato: nadie decidió antes de comprar. La decisión que no se tomó entonces es la de este epígrafe, y es tuya: desde dónde arrancas. No se decide por lo que promete un catálogo, sino por cuatro datos que puedes reunir esta semana en tu centro: qué hay, cuántas horas tienes, qué saberes admiten anclaje y quién más está dentro. De ahí sale el escalón de arranque, y de ahí sale el filtro que evita el armario lleno de material parado.

1.5.1. Las cuatro columnas del inventario

La primera columna es **el material real, y su estado hoy**. No el que figura en el inventario de secretaría: el que se puede encender esta tarde. Cuenta las unidades, anota si arrancan y si tienen cargador, y añade una línea que casi nunca está escrita: de quién es. Un material sin responsable no se repara.

La segunda columna son **las horas, y de qué área salen**. El horario estatal no asigna ni una hora al pensamiento computacional como tal: lo que hay son saberes dentro de áreas que ya tienen su horario —Matemáticas y Conocimiento del Medio en Primaria, Tecnología y Digitalización en Educación Secundaria Obligatoria—. La pregunta correcta no es «cuántas horas le dedico», sino «de qué área salen y a qué saber sustituyen». El reparto del calendario es del epígrafe 3.5; aquí basta el número.

La tercera columna son **los saberes de tu programación que admiten anclaje**. Vuelve al mapa que hiciste en el epígrafe 1.2 y marca los que ya te obligan. En Primaria son dos, y el Real Decreto 157/2022 los escribe así. Anexo II, Matemáticas, bloque D, segundo ciclo: «Estrategias para la interpretación y modificación de algoritmos sencillos». Anexo II, Conocimiento del Medio Natural, Social y Cultural, bloque B, segundo ciclo: «Iniciación en la programación a través de recursos analógicos (actividades desenchufadas) o digitales (plataformas digitales de iniciación en la programación, aplicaciones de programación por bloques, robótica educativa...)». Esa disyuntiva del legislador es la que sostiene que se pueda empezar sin comprar nada.

La cuarta columna es **quién más del claustro está dentro**, con nombres, área y grado de compromiso real: quien coordinaría, quien cedería una sesión, quien prestaría material y quien no se opondría. La autonomía del centro llega hasta el cuándo y el cómo, no hasta el si, como viste en el epígrafe 1.2: por eso esta columna se rellena con nombres y no con intenciones.

1.5.2. Del inventario al escalón de arranque

Con las cuatro columnas delante, el escalón se elige casi solo. Hay cuatro, ordenados por lo que exigen de gestión, no por lo que valen:

Desenchufado puro. Papel, tarjetas, tiza, cuerpo y suelo. No depende de ningún aparato y sirve para el grupo completo. Es el escalón por diseño, no la opción del que no tiene.

Desenchufado con un dispositivo del docente. El mismo trabajo más un ordenador proyectado que ejecuta delante de todos lo escrito en papel. Añade la comprobación pública del error sin multiplicar los aparatos.

Programación por bloques con rotación. Exige dispositivos para un subgrupo y una organización en estaciones. La rotación por estaciones es un modelo de organización descrito y nombrado por el Clayton Christensen Institute en 2012, dentro de una taxonomía que clasifica lo que existe; no es un resultado de eficacia y aquí no se presenta como tal [R].

Robot por equipo. El más caro en material y, sobre todo, en gestión: carga, emparejamiento, turnos y reparto de papeles.

Con la etiqueta correcta: no hay ningún estudio comparativo que mida cuál de estos cuatro escenarios enseña más. Lo que sigue es criterio profesional razonado [P], y la regla que este curso sostiene es elegir **el escalón más bajo que sostengan tus cuatro columnas**, porque es el que se puede repetir en enero sin depender de nada. Subir escalón es del epígrafe 3.2.

Del inventario al escalón de arranque



Fuente: elaboración propia.

FIGURA 1.5

1.5.3. El filtro de compra: cinco preguntas antes de gastar un euro

Este curso no recomienda ninguna marca, ningún modelo ni ningún kit, ni con nombre propio ni por descarte. Da un filtro, que sirve igual para decidir una compra que para explicar por qué no procede. Son cinco preguntas, y se contestan por escrito:

1. **¿A qué saber básico de mi programación se ancla?** Si la respuesta no cita un saber o un criterio concreto, el kit no resuelve un problema curricular.
2. **¿Cuántos alumnos trabajan a la vez con él?** Divide el grupo entre las unidades que se comprarían. Ese número decide la organización de cada sesión.
3. **¿Qué pasa cuando se pierde una pieza?** Si el conjunto deja de funcionar y el repuesto no se vende suelto, lo que se compra tiene fecha de caducidad.
4. **¿Qué licencia y qué cuentas exige?** Si necesita una cuenta por alumno, la decisión ya no es solo pedagógica ni es tuya: el registro de menores, el consentimiento y la protección de datos son del curso 04, y allí hay que ir antes de crear una sola cuenta.
5. **¿Quién lo mantiene en septiembre?** Con nombre. Un material sin persona responsable vuelve al armario en el segundo trimestre.

ATENCIÓN · EL ARMARIO LLENO

El patrón se repite: llega material por una subvención, se usa en la semana cultural, sale en la foto de la web del centro y se guarda. No falló el aparato: faltó la decisión previa. Nadie había escrito a qué saber se anclaba, cuántos alumnos lo usarían a la vez ni quién lo iba a mantener. El filtro se aplica antes de la compra; recuperar un armario parado cuesta un curso entero.

1.5.4. Dónde encaja Código Escuela 4.0, y dónde no

Es probable que en tu centro se nombre este programa al hablar de material, y hay que situarlo con precisión porque se cita mal. Código Escuela 4.0 es un programa de cooperación territorial: financiación de equipamiento y de acompañamiento al profesorado. El acuerdo de Consejo de Ministros de 9 de mayo de 2023 transfirió 298,4 millones de euros: 199 millones para equipamiento y 99,4 para apoyo y formación del profesorado, con 65 millones adicionales previstos para 2024. El reparto territorial de 2024 movió 58 millones de euros.

NORMATIVA · PROGRAMA DE COOPERACIÓN TERRITORIAL CÓDIGO ESCUELA 4.0

Resolución de 5 de septiembre de 2024, de la Secretaría de Estado de Educación (BOE-A-2024-18091, BOE núm. 218, de 9 de septiembre de 2024). Objetivo del programa: «generalizar y facilitar al profesorado y al alumnado de los centros educativos sostenidos con fondos públicos el desarrollo de las competencias digitales relacionadas con el pensamiento computacional y la programación». Periodo de ejecución: de 1 de septiembre de 2024 a 31 de diciembre de 2026.

La distinción que hay que retener: el programa paga el armario; lo que obliga es el saber básico del real decreto. No es una obligación curricular ni un método, no prescribe secuencia y no impone aparato. Que llegue material por esta vía no cambia lo que tienes que enseñar; cambia con qué puedes hacerlo, y solo si el filtro da verde.

1.5.5. Una cifra que no vas a leer en este curso

Habrás visto que el 65 % de los escolares de hoy trabajará en empleos que aún no existen. Se usa para justificar compras, y aquí no se va a usar. Un reportaje de verificación de *Chalkbeat* de 21 de noviembre de 2017 rastreó la cifra hasta un libro de 2011 cuya propia autora dejó de emplearla al admitir que carecía de fuente, y halló que el organismo internacional que más la difunde la referencia con vídeos, sin base documental. Lo exacto no es que sea falsa —nadie lo ha demostrado—, sino que **no tiene fuente identificable**. Si tu argumento para arrancar necesita ese dato, el argumento no es el material.

Aplicación práctica

1. **Levanta el inventario de cuatro columnas de tu centro.** Recorre el armario del aula de material y pregunta en secretaría y a quien coordine digitalización: **lo haces solo**. Escribe las cuatro columnas —material y su estado hoy, horas disponibles y área de la que salen, saberes de tu programación que admiten anclaje, y profesorado implicado con nombre y grado de compromiso— y cierra cada una con una cifra, no con una impresión. Guárdalo como **inventario-de-cuatro-columnas.md**. **Una hora y media.**
2. **Elige tu escalón de arranque y justifícalo por escrito.** Sobre el inventario anterior, nombra el escalón desde el que empiezas y escribe qué columna lo sostiene: **lo haces solo**. Escribe también los tres que descartas y por qué, con el dato del inventario que lo descarta en cada caso. Añade una línea más, la que evita discusiones en marzo: qué tendría que cambiar en tus columnas para subir un escalón. Guárdalo como **escalon-de-arranque.md**. **Una hora y media.**

3. **Aplica el filtro de compra a un kit real, sin comprarlo.** Abre el catálogo público de un proveedor cualquiera y elige un kit anunciado para tu etapa: **lo trabajas con material público**. Contesta por escrito las cinco preguntas del apartado 1.5.3 con lo que el catálogo dice y, sobre todo, con lo que no dice, que suele ser lo más informativo. No se compra nada ni se pide presupuesto: el producto del trabajo es el documento. Guárdalo como [filtro-de-compra-aplicado.md](#). **Una hora y media.**