

# Introducción al Pensamiento Computacional: Metodología STEAM



#CompDigEdu  
CoDiCe TIC  
Junta de  
Castilla y León  
Consejería de Educación



Víctor Ruiz



Financiado por  
la Unión Europea  
NextGenerationEU

# Tecnología y aprendizaje



¿Qué te sugiere el Pensamiento Computacional?

# Buscando una definición

Nacimiento en los 80  
(Seymour Papert).

No hay definición  
consensuada.

Habilidad para el s.XXI.

“Pensamiento Computacional es la habilidad para formular, representar y resolver problemas haciendo uso de herramientas, conceptos y prácticas de las ciencias de la computación (informática), tales como la abstracción, la descomposición o el uso de simulaciones.”

Grupo KGB-L3 Universidad Rey Juan Carlos

— — —

# Problem solving

---

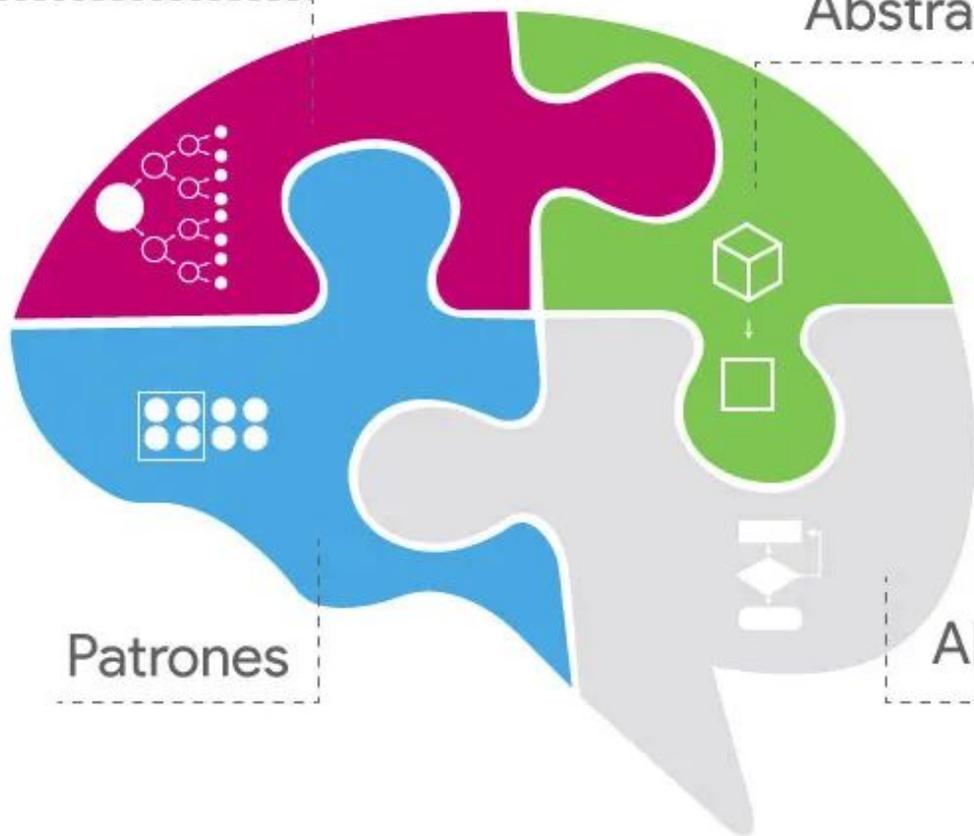
Por grupos, escribimos las instrucciones necesarias para hacer un sándwich de mantequilla de cacahuete y mermelada.





Descomposición

Abstracción



Patrones

Algoritmos

# Dimensiones del Pensamiento Computacional

Marco conceptual de  
Brennan-Resnick.

Como docentes, nos resultaría más útil una definición operacional que nos describa los rasgos característicos del Pensamiento Computacional, que una definición conceptual.

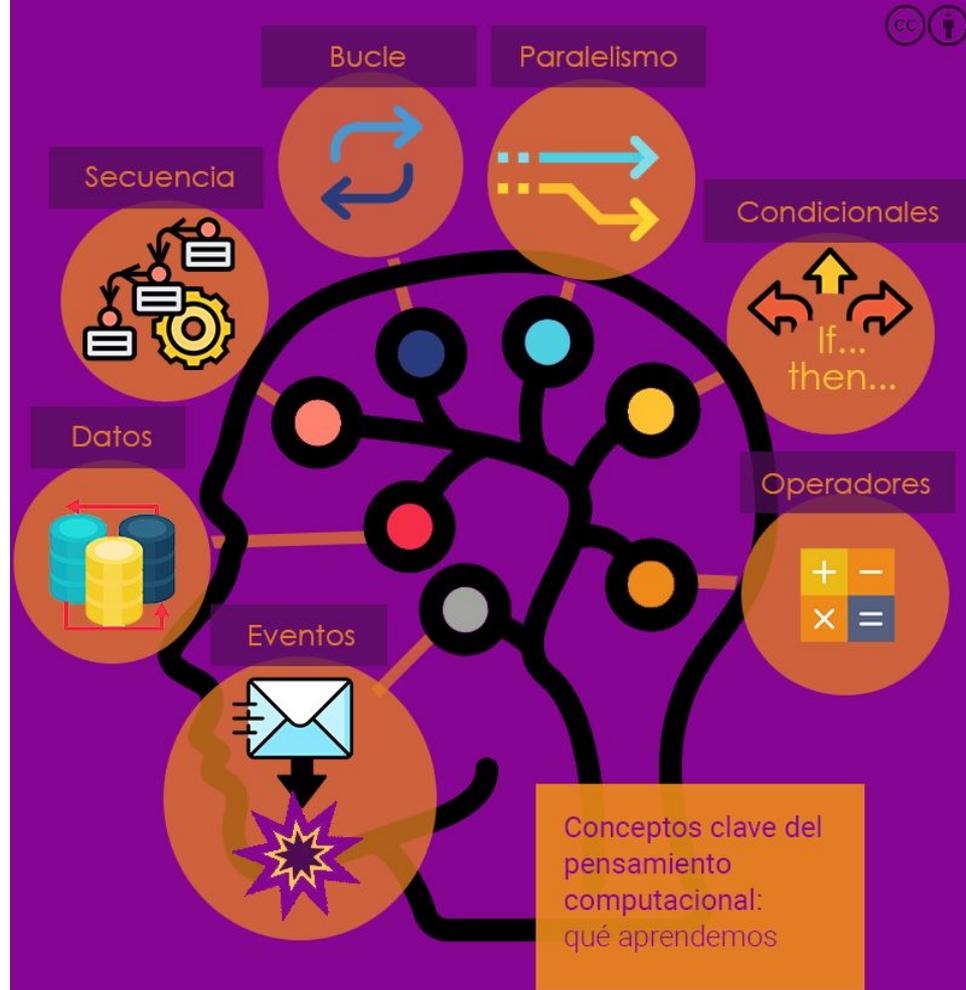
---

# PENSAMIENTO COMPUTACIONAL



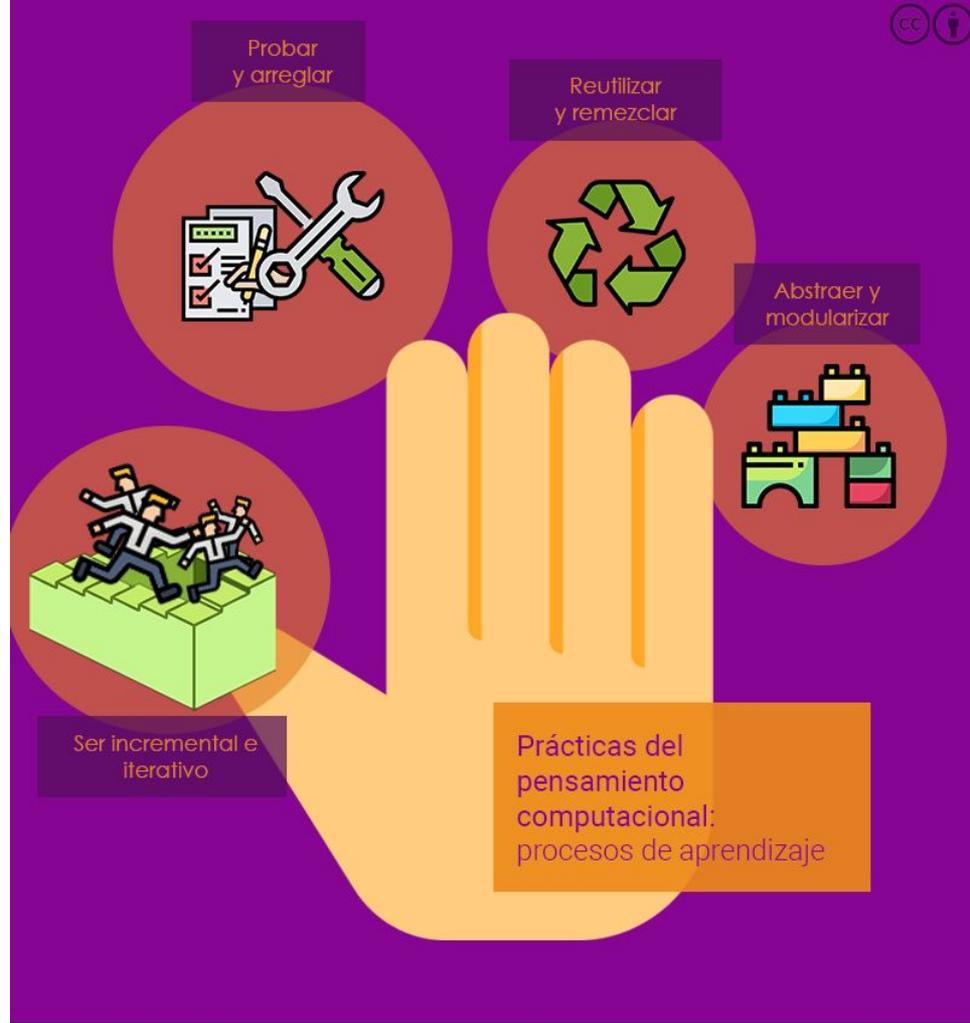
# Conceptos

Este grupo de elementos representa las principales ideas **que aprendemos** cuando desarrollamos nuestro Pensamiento Computacional.



# Procedimientos

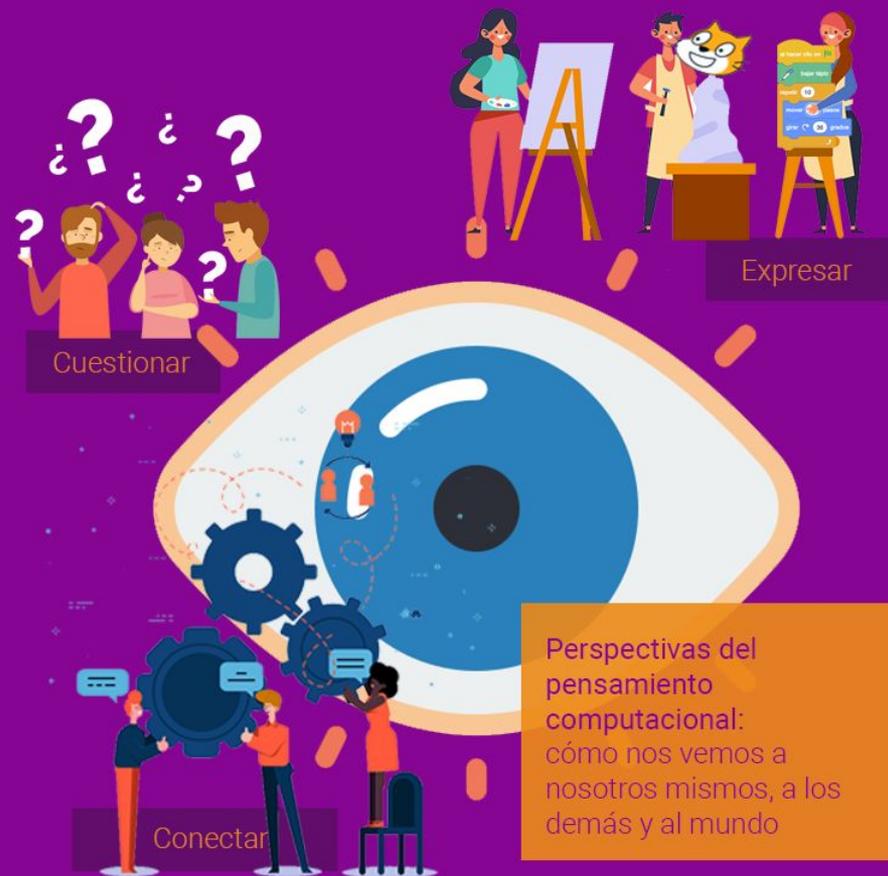
Las prácticas computacionales van a enfocarse en los procesos de pensamiento y aprendizaje, es decir, pasamos del qué aprendemos al **cómo aprendemos**.



# Actitudes

— — —

**Evoluciona la manera en que nos vemos a nosotros mismos, a los demás, al mundo que nos rodea,** y especialmente a todo lo relativo con las tecnologías digitales. Llegamos a ser más críticos con las herramientas digitales que usamos, las comprendemos mejor, somos más eficaces cuando necesitamos aprender alguna, seleccionamos las más adecuadas para expresar lo que queremos y no nos sentimos tan abrumados y perdidos en el universo digital.



# Dimensiones del Pensamiento Computacional

De forma sencilla, podemos  
identificar:

1. Secuenciación y nociones algorítmicas
2. Pensamiento lógico
3. Abstracción y descomposición
4. Paralelismo y sincronización mediante eventos
5. Representación de la información
6. Detección de errores, eficiencia y perseverancia



# 1. Secuenciación y nociones algorítmicas

---

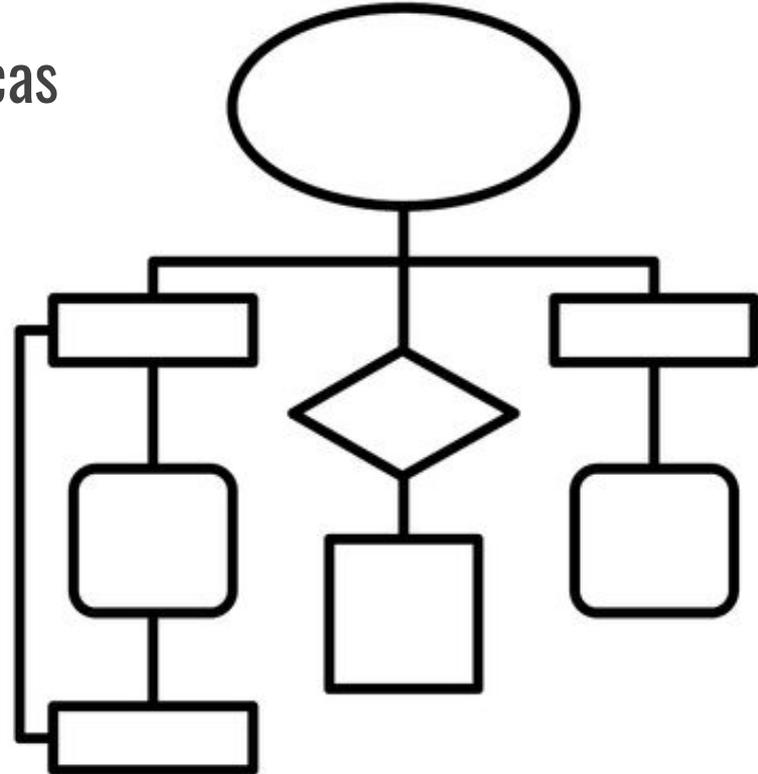
Un **algoritmo** es una secuencia de instrucciones ordenadas que permiten realizar una tarea.

Contar con **nociones algorítmicas** de control de flujo nos ayudará a simplificar esta secuencia en proyectos grandes.

Dividir y relacionar ideas, simples o complejas.

Ordenar con propósito.

Reusar y remezclar.



## 2. Pensamiento lógico

Tiene que ver con **el razonamiento, la predicción y el análisis.**

Contribuirá a que los proyectos de pensamiento computacional sean dinámicos.

Relacionar objetos reales y abstractos.

Pensamiento científico.

Argumentación, deducción.



¿De quién es el cumpleaños?



# 3. Abstracción y descomposición

La **descomposición de problemas** en partes más pequeñas que serán más fáciles de comprender, crear y mantener.

La **abstracción** permite simplificar las cosas al olvidarnos de los detalles innecesarios y centrarnos en los aspectos importantes.

Abstraer y modularizar.

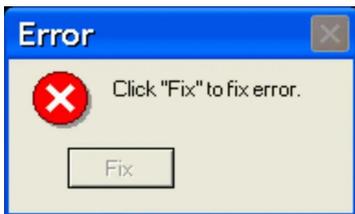
Superar al usuario.





# Abstracción

Utilizamos el concepto de **abstracción** para coger un **sistema relativamente complejo y simplificarlo para nuestro uso**. Usas la abstracción cada día en el mundo real, y puede que no lo sepas.



## 4. Paralelismo y sincronización mediante eventos

En la mayoría de las creaciones informáticas se requiere un cierto nivel de **paralelismo**, que es la posibilidad de que varias cosas ocurran al mismo tiempo.

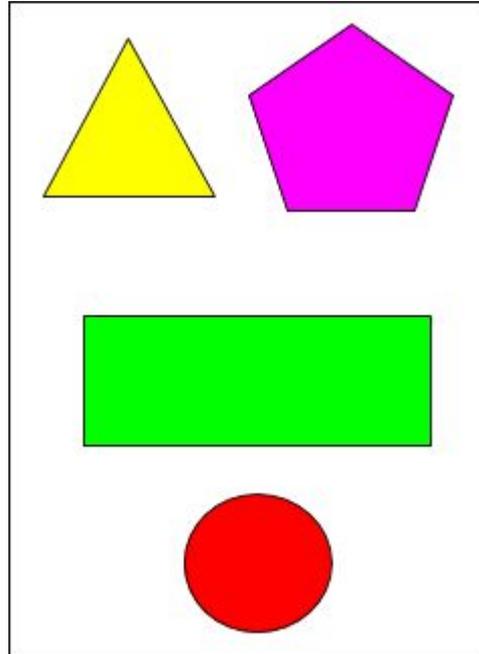
En muchas situaciones, necesitamos **sincronizar** varias acciones.

Relacionar ideas.

Solucionar problemas.



# Mando a distancia



# 5. Representación de la información

---

La capacidad de **representación de la información** nos ayudará a estructurar los datos requeridos de la forma más eficiente posible.

Comunicar resultados.

Expresión oral y escrita.

Creación de contenidos.

Ser incremental e iterativo.

Curación de información.



# 6. Detección de errores, eficiencia y perseverancia

Grace Murray Hopper

Debemos aprovechar al máximo y de la mejor manera los recursos que nos brindan las técnicas informáticas para conseguir el objetivo de **manera eficiente.**

Debemos ser constantes, reflexivos, pacientes... La **motivación** es la llave que abre la puerta de la perseverancia, y para ello, **el error** debe convertirse en nuestro mayor aliado.

Proponerse metas.

Debugging y feedback.

9/9

0800	Anttan started		
1000	" stopped - anttan ✓		
	1300 (032) MP-MC	1.2700	9.037 847 025
	(033) PRO 2	2.130476415	9.037 846 895 correct
	connect	2.130476415	4.615925059(-2)
	Relays 6-2 in 033 failed special speed test in relay		
1100	Relays changed		
1525	Started Cosine Tape (Sine check)		
	Started Multi-Adder Test.		
1545			Relay #70 Panel F (moth) in relay.
1630	Anttan started.		
1700	closed down.		

First actual case of bug being found.

Relay #70  
2145  
Relay #3376



Debugging, eficiencia y perseverancia



# Pensamiento Computacional

En la práctica.

Pensamiento  
Computacional  
Desconectado

Robótica

Programación

Impresión 3D  
Cortadora láser

Inteligencia  
Artificial

Mobile  
learning

RV y RA

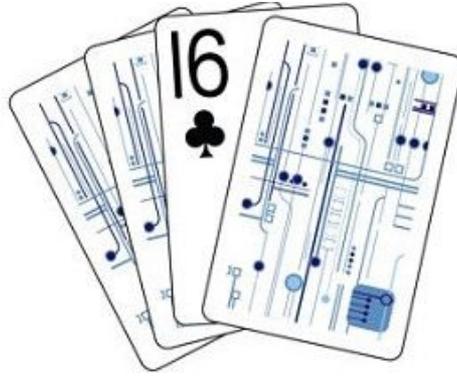
Metaverso

IoT

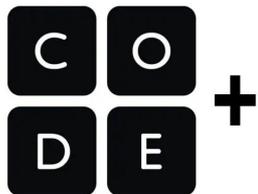
# Pensamiento Computacional Desconectado



# Pensamiento Computacional Desconectado



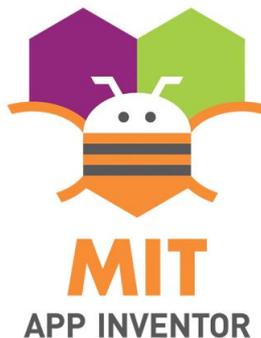
# Programación



+



A Visual Programming Language



Microsoft MakeCode



Python



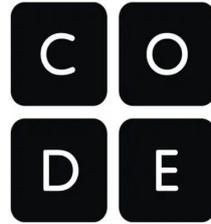
Scratch

# Programación

The Scratch logo consists of the word "SCRATCH" in a white, rounded, bubbly font. Each letter is contained within a thick, orange, irregular outline that resembles a splash or a drop.

[Narrativa Digital](#)

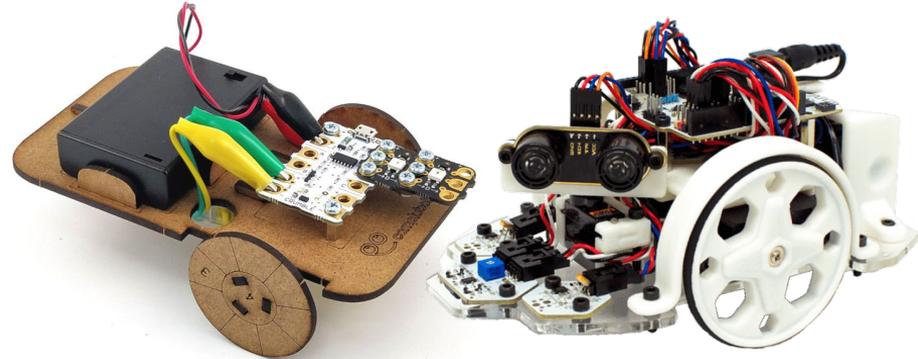
[Videojuegos](#)



# Robótica

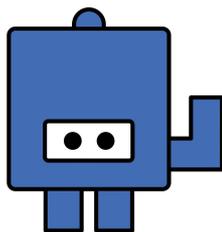
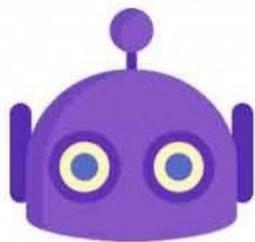


# Impresión 3D y cortadora láser



# Inteligencia Artificial

Artificial Intelligence and  
Machine Learning for Kids



**Watson**  
va a clase



**IoT**



**Realidad  
Virtual y  
Realidad  
Aumentada**



**Metaverso**



**Mobile  
learning**





# Experiencias educativas significativas

¿Integraciones curriculares y  
metodológicas?

# Pensamiento Computacional en LOMLOE

En todas las etapas no  
universitarias.



# PENSAMIENTO COMPUTACIONAL EN LA LOMLOE

## ENSEÑANZAS MÍNIMAS

**Al completar este nivel educativo el alumno o la alumna...**

**Áreas o materias**

### INFANTIL

Desarrolla, de manera progresiva, las destrezas del pensamiento computacional a través de procesos de observación y manipulación de objetos para iniciarse en la interpretación del entorno y responder de forma creativa a las situaciones y retos que se plantean.

Área 2. Descubrimiento y exploración del entorno (en el 2º ciclo)

### PRIMARIA

Se inicia en el desarrollo de soluciones digitales sencillas y sostenibles (reutilización de materiales tecnológicos, programación informática por bloques, robótica educativa...) para resolver problemas concretos o retos propuestos de manera creativa, solicitando ayuda en caso necesario.

Conocimiento del medio natural, social y cultural (en los 3 ciclos ) y Matemáticas (en los 3 ciclos)

### E.S.O.

Desarrolla aplicaciones informáticas sencillas y soluciones tecnológicas creativas y sostenibles para resolver problemas concretos o responder a retos propuestos, mostrando interés y curiosidad por la evolución de las tecnologías digitales y por su desarrollo sostenible y uso ético.

Tecnología y Digitalización (1º, 2º y 3º), Tecnología (4º), Digitalización (4º), Biología y Geología (1º, 2º, 3º y 4º), Matemáticas (1º, 2º y 3º), Matemáticas A y B (4º), Ámbito Ciencias Aplicadas (CFGB)

### BACHILLERATO

Desarrolla soluciones tecnológicas innovadoras y sostenibles para dar respuesta a necesidades concretas, mostrando interés y curiosidad por la evolución de las tecnologías digitales y por su desarrollo sostenible y uso ético

Tecnología e Ingeniería I y II, Biología, Geología y Ciencias Ambientales (1º), Geología y Ciencias Ambientales (2º), Matemáticas I y II, Matemáticas Aplicadas a las Ciencias Sociales I y II, Matemáticas Generales

# Programación, robótica y pensamiento computacional en el aula

Ponencia: Situación en  
España y propuesta  
normativa

INTEF, octubre 2018



# Educación Infantil, un posible enfoque

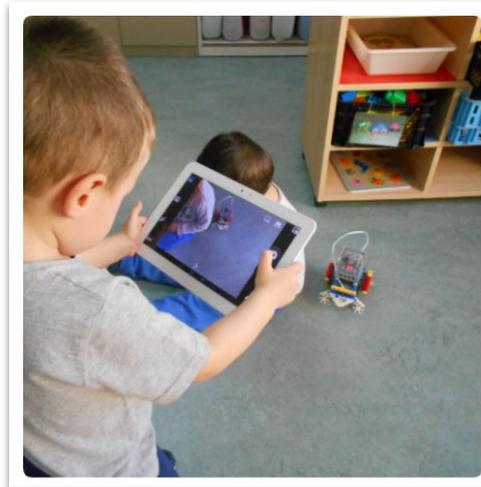
---

Se usan estas habilidades como **vehículo para el aprendizaje interdisciplinar**, utilizando una pedagogía basada en el juego que amplía lo que el profesorado ya hace de manera habitual en sus aulas en su día a día.

Para ilustrar este enfoque tomemos por ejemplo el concepto de secuenciación. La programación y la robótica ofrecen otro escenario para trabajar este mismo concepto, y el alumnado aprenderá sobre él de un modo práctico, divertido y tangible.

# Educación Infantil

- Desafíos.
- Tapetes.
- Gamificación.
- Manipulativos.
- Psicomotricidad.
- Nuevos rincones.
- Productos finales.
- Alfabetización digital.
- Realidad aumentada.
- Creatividad.
- Resolución de problemas.
- Aprendizaje colaborativo.
- Espíritu crítico.
- Comunicación.
- ...



# Educación Primaria, un posible enfoque

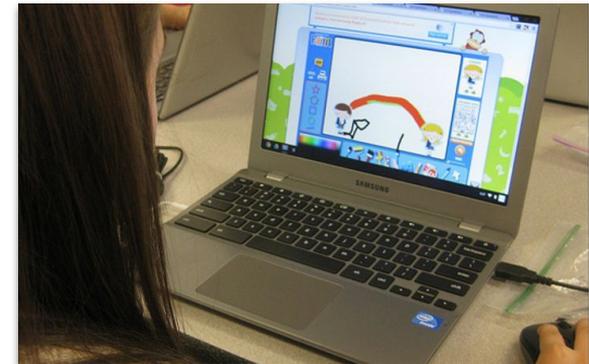
---

Mientras en los **primeros cursos** se usan habitualmente recursos muy similares a los utilizados en Infantil, como actividades desenchufadas, sencillos robots programables y lenguajes de programación visuales basados en flechas y símbolos, **en cursos más altos** se hace uso de lenguajes de programación visuales basados en bloques -como Scratch, Blockly o Kodu-, robots y placas algo más avanzadas -como Makey Makey, Crumble, Lego WeDo o Arduino- y en algunos casos, incluso, lenguajes de programación basados en texto de los utilizados en la industria, como pueden ser Python, Java o Swift.



# Educación Primaria

- Desafíos.
- Gamificación.
- Centros de interés.
- Estaciones.
- Productos finales.
- Alfabetización digital.
- Alfabetización informacional.
- Capacidad de análisis.
- Creatividad.
- Resolución de problemas.
- Aprendizaje colaborativo.
- Espíritu crítico.
- Comunicación.
- ...



# Educación Secundaria, un posible enfoque

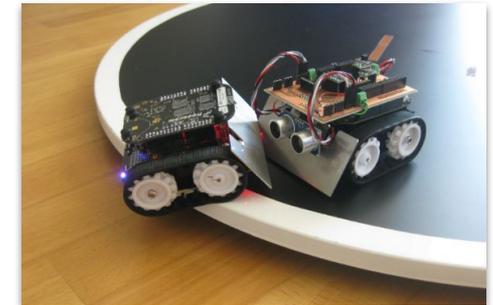
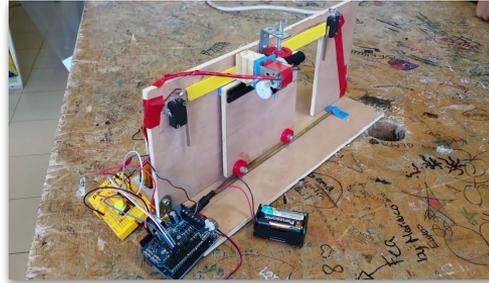
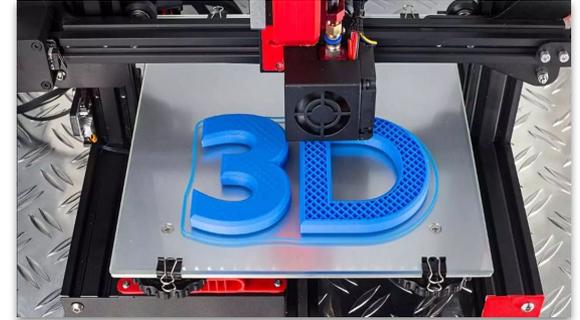
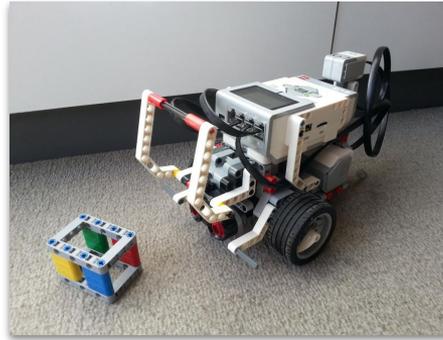
---

Es habitual que, especialmente **en los primeros cursos**, se utilicen lenguajes de programación visuales como Scratch, con el que se crean juegos, simulaciones, demostraciones y proyectos artísticos, entre otros tipos de proyectos, o Alice, muy utilizado para la creación de narraciones digitales. También se utilizan otros entornos como Minecraft Edu para el modelado y la programación 3D, o App Inventor para el desarrollo de aplicaciones para dispositivos móviles. Por su parte, **en los cursos más altos** es habitual que se utilicen lenguajes de programación profesionales basados en texto, como Python, Swift o JavaScript.

# Educación Secundaria

---

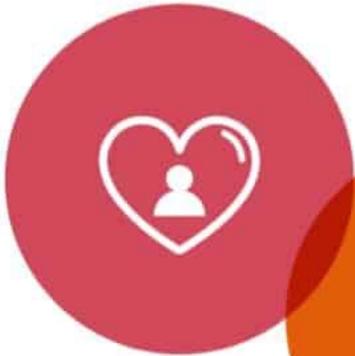
- Retos y gamificación.
- Centros de interés.
- Productos finales.
- Alfabetización digital e informacional.
- Narrativa digital.
- Creación de contenidos.
- Capacidad de análisis.
- Creatividad.
- Resolución de problemas.
- Aprendizaje colaborativo.
- Espíritu crítico.
- Comunicación.
- ...



**Algunos aliados interesantes**

# Pensamiento de Diseño

EMPATIZA



IDEA



PRUEBA



DEFINE



MAQUETA



IMPLEMENTA



# MAKERS

Desarrollan su creatividad

Se divierten creando

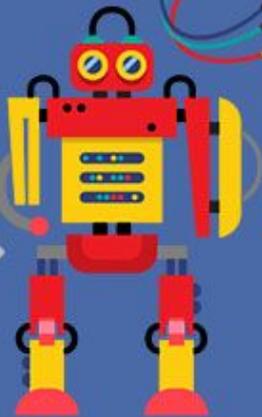
Aprenden de sus errores

Se inspiran en su entorno



Comparten

Trabajan en grupo



Interaccionan sin miedo con la tecnología

Investigan, se documentan y relacionan temas

# Conclusiones

**¡Muchas gracias!**