Pensamiento Computacional en el Aula

Reunión Coordinadores TIC
20 noviembre 2017

Fernando Posada Prieto
fernandoposada@gmail.com
Centro del Profesorado de Lanzarote
Noviembre 2017

Índice de contenidos

① Concepto de Pensamiento Computacional.
② Formas de trabajar PC en el aula.
③ Recurso para trabajar PC SIN dispositivos.
④ Recursos para PC CON dispositivo y SIN robot.
⑤ Herramientas para crear aplicaciones.
⑥ Scratch: estándar de programación visual educativa.
⑦ Recursos para PC CON dispositivo y CON robot
⑧ Criterios didácticos
⑨ Sugerencias metodológicas.
⑩ Idea final.

1 Concepto de Pensamiento Computacional ...

¿De qué hablamos?

Tendencia europea y mundial

Pensamiento computacional en el currículum oficial

¿Por qué ahora?

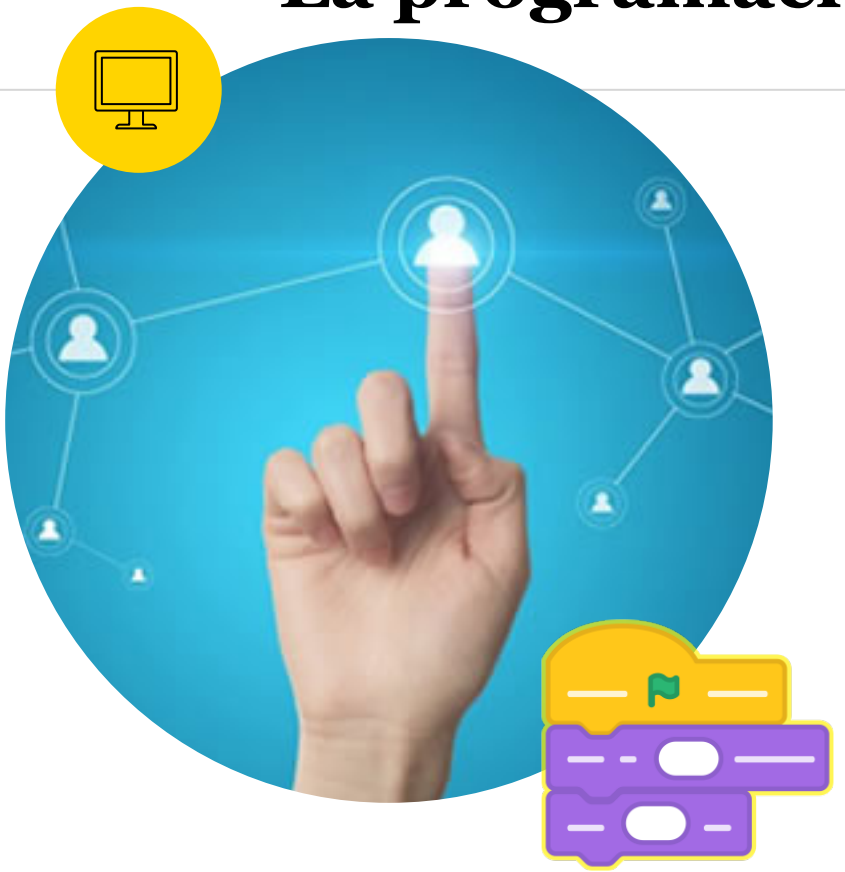
De ciudadanos **CONSUMIDORES** de tecnología a ciudadanos **PRODUCTORES** de tecnología

Pensamiento Computacional para fomentar ...

La creación de tecnología como fuente de riqueza

Estimular la formación de un colectivo de creadores digitales que afronten desafíos complejos, interdisciplinares, innovadores y de interés público.

La programación visual



- Las herramientas de **programación visuales** han facilitado el trabajo escolar con PC en todas las edades.
- La creación de scripts se realiza configurando las propiedades de **bloques** y conectándolos en una secuencia lógica.

Concepto
Pensamiento Computacional

Procesos del pensamiento humano implicados en la formulación de problemas y su resolución aplicando un agente de procesamiento de información, bien humano o bien máquina

“

2010 Jeannete Wing

HABILIDADES del PC

① **Reformular problemas** → adaptarlos a la computación.
② **Organizar/analizar datos** → patrones y conclusiones.
③ **Representar datos** → descomposición y abstracción.
④ **Automatizar** → diseño creativo de algoritmos.
⑤ **Codificar** → crear el programa.
⑥ **Depurar** → resolver los errores.
⑦ **Analizar posibles soluciones** → eficiencia.
⑧ **Generalizar** → transferencia de la solución.

ACTITUDES del PC

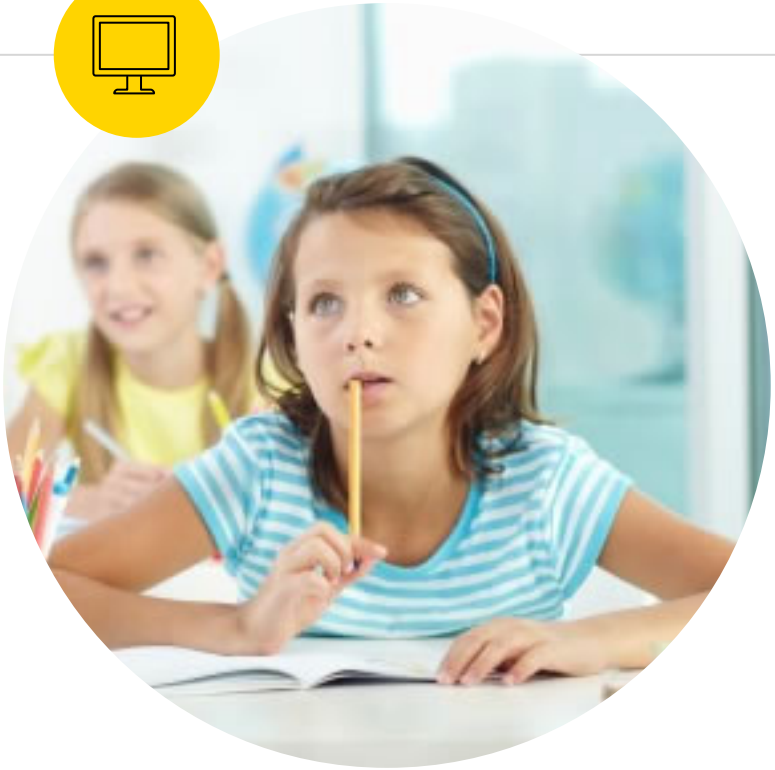
① Confianza en el manejo de la complejidad
② Persistencia al trabajar los problemas difíciles
③ Tolerancia a la ambigüedad
④ Habilidad para enfrentarse a problemas
⑤ Habilidad para comunicarse y trabajar con otros

Conceptos computacionales básicos

1. Abstracciones y generalizaciones.
2. Sistemas de símbolos y representación abstracta.
3. Noción algorítmica del control de flujo.
4. Descomposición estructurada del problema.
5. Pensamiento iterativo, recursivo y paralelo.
6. Lógica condicional.
7. Depuración y detección sistemática de errores.
8. Restricciones de eficiencia y desempeño.

Grover y Pea (2013)

Algo más que aprender a programar



Computacional Thinking = Resolución de problemas con metodología propia que incorpora ...

- **CREATIVIDAD**
- **RAZONAMIENTO LÓGICO**
- **PENSAMIENTO CRÍTICO**

CON y SIN máquinas

- El PC como sistema de resolución de problemas.
- SIN máquinas: tareas con juegos, puzzles, pinturas de colores, dibujos, tarjetas, cartulinas ...
- Ej: Computer Sciences Unplugged csunplugged.org

Situaciones para aplicar PC

- Trazando RUTAS
- Pintando FORMAS
- Haciendo BÚSQUEDAS
- Estableciendo ORDEN
- Contando HISTORIAS
- Resolviendo PROBLEMAS
- Creando APLICACIONES
- Moviendo ROBOTS
- Sintiendo EMOCIONES ...

2

Formas de trabajar PC en el aula

① SIN dispositivos: lápiz y papel.
② CON tableta/PC y SIN robot.
③ CON robot y CON/SIN tableta/PC

EN LA GRANJA DE MI TÍO

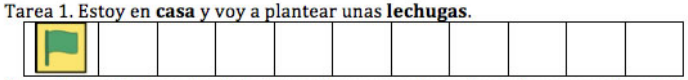
Bloques de órdenes:

Consignas iniciales:

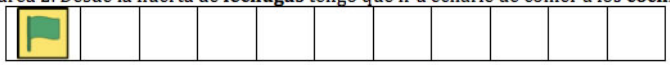
- **Punto de vista.** Imagina que eres el robot en todo momento.
- **Entradas**. En las parcelas hay que entrar por las puertas.
- **Salidas.** Se inicia la salida de cada parcela mirando hacia la puerta.
- **Lago central**. No se puede pasar por el estanque de los patos.
- **Pasos.** En cada paso se avanzará una casilla en horizontal o vertical.

Tareas a codificar:

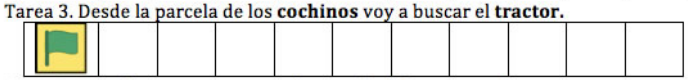
Tarea 1. Estoy en **casa** y voy a plantear unas **lechugas**.



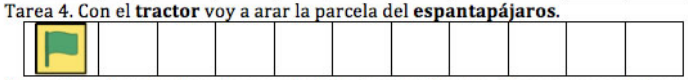
Tarea 2. Desde la huerta de **lechugas** tengo que ir a echarle de comer a los **cochinos.**



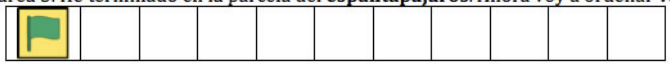
Tarea 3. Desde la parcela de los **cochinos** voy a buscar el **tractor.**



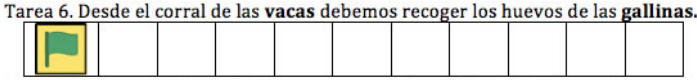
Tarea 4. Con el **tractor** voy a arar la parcela del **espantapájaros**.



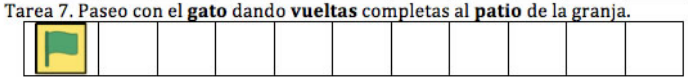
Tarea 5. He terminado en la parcela del **espantapájaros**. Ahora voy a ordeñar **vacas**.



Tarea 6. Desde el corral de las **vacas** debemos recoger los huevos de las **gallinas.**



Tarea 7. Paseo con el **gato** dando **vueltas** completas al **patio** de la granja.

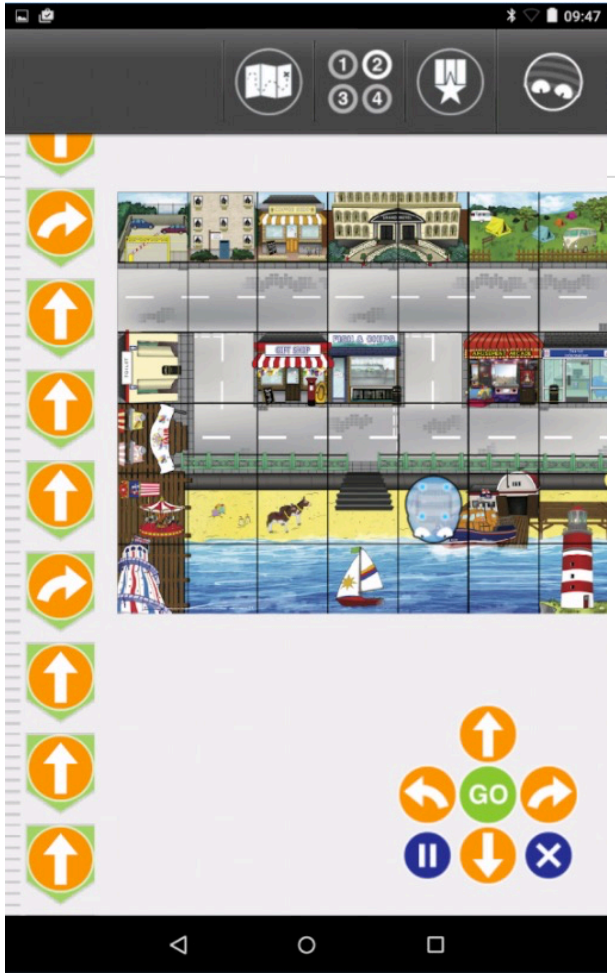


App Blue-Bot

- Es una aplicación para dispositivos móviles Android e IOS que permite trabajar CON robot y SIN robot.

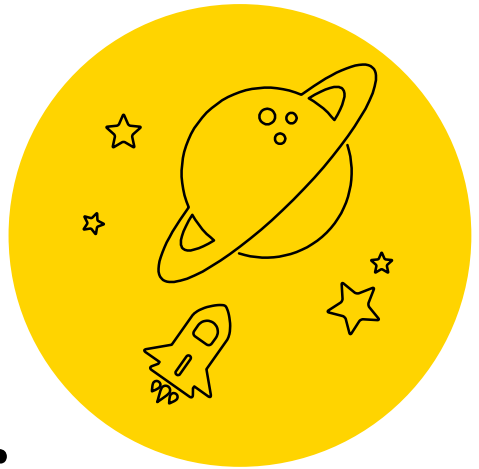
App Blue-Bot

- 2 modos: Explore (exploración) y Challenge (desafío).
- En **Modo Exploración**: Step by step, Basic programming, Repeats (bucles) y 45 Degree Turns.
- En **Modo Desafío**: Get from A to B, Obstacules, Fewer Buttons y Random Instructions.



La robótica del tapiz

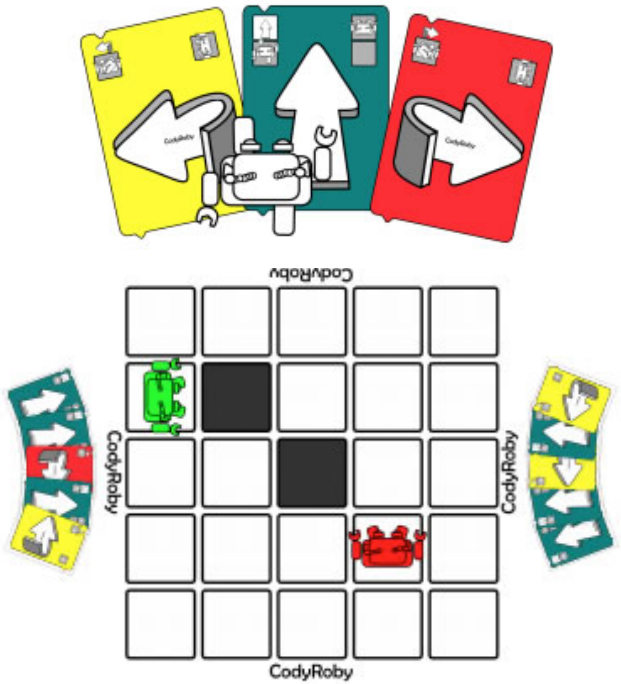
- Cuenta-Cuentos: los 3 cerditos.
- Narrar un cuento clásico programando el movimiento del robot sobre el tapiz.



3 Recursos para trabajar PC SIN dispositivos

SIN ordenadores
SIN tabletas
SIN robots

Cody & Roby



- Es un juego de cartas y tablero por equipos.
- Se piensa una ruta del robot, se crea la secuencia de cartas (Cody) y se interpreta sobre el tablero (Roby).
- Dos variantes del juego : Follow Me y el Duelo.
- Permite trabajar el pensamiento computacional sin ordenadores.
- + Información: codeweek.it/cody-roby-en/

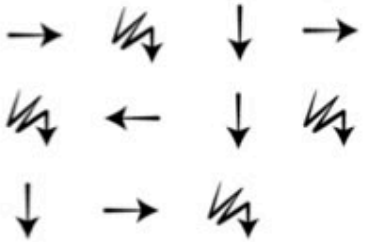
GRÁFICOS CUADRICULADOS

Idea inicial

En esta actividad se proporciona una figura dibujada en una parrilla de 4x4 cuadros y se solicita crear un programa que reproduzca esa figura mediante una secuencia de instrucciones visuales. El punto de partida es siempre el cuadro situado en la esquina superior izquierda (estrella).

Figura

Programa



Instrucciones

- Lee la secuencia de órdenes de izquierda a derecha y de arriba hacia abajo.
- Se inicia desde la cuadricula situada en la esquina superior izquierda.
- Sigue la secuencia desplazando tu dedo sobre la rejilla.

Práctica:

Por parejas. Uno elige una de las imágenes y crea la secuencia de órdenes. A continuación oculta con la mano la figura de la izquierda y le pide al compañero/a que reproduzca la imagen en la parrilla en blanco situada a la derecha leyendo la secuencia escrita. Al concluir se comparan las parrillas de ambos lados. En la última parrilla se puede dibujar libremente el motivo gráfico.

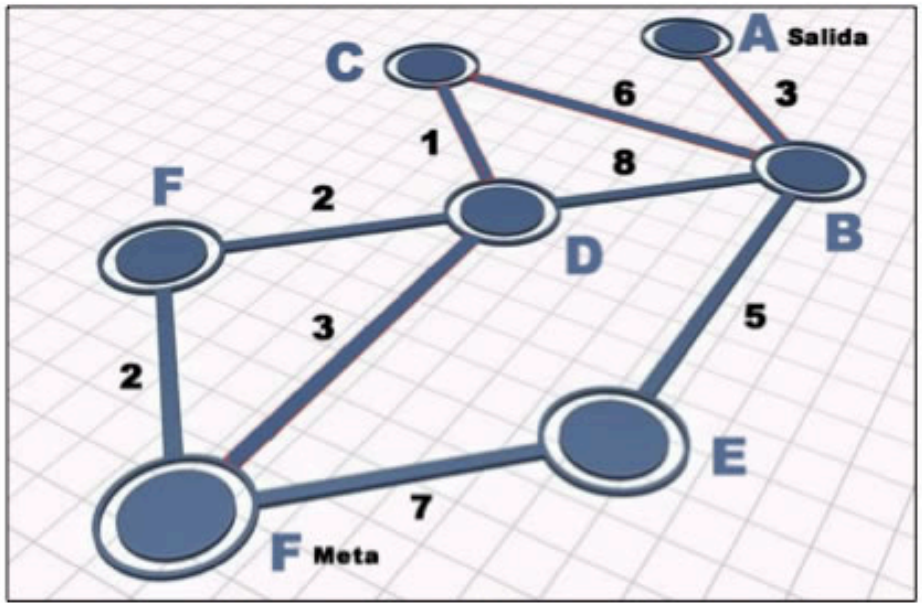
MIS BARCOS

Instrucciones:

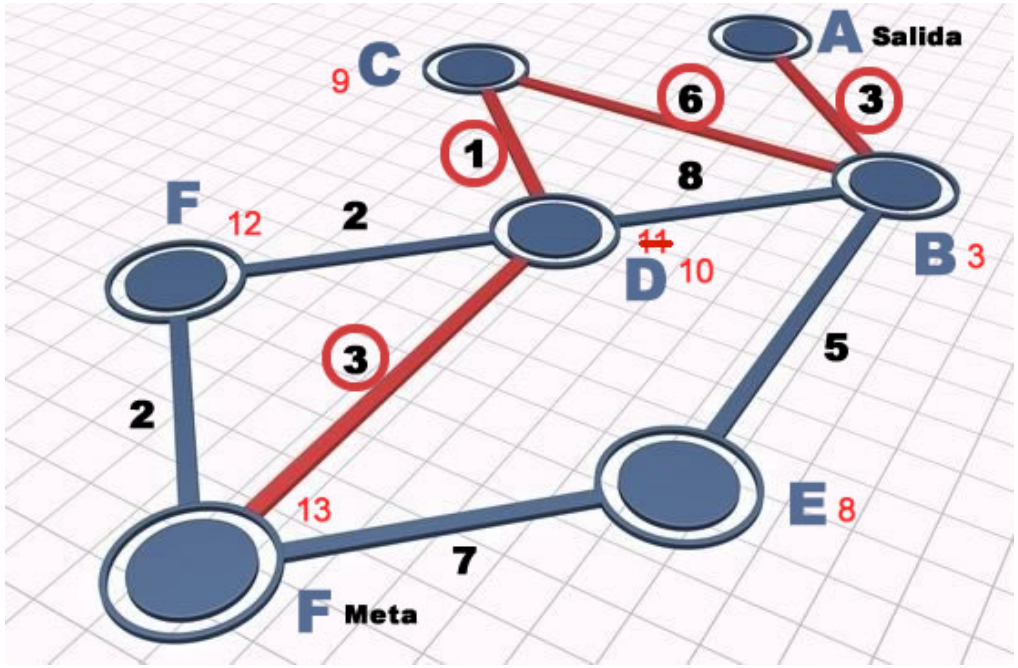
- Juego por parejas donde cada persona tiene su hoja.
- Cada jugador rodea con lápiz una letra en la flota Mis Barcos. Es el puerto donde se sitúa su barco oculto. ¡No muestres la hoja al compañero!
- Por turnos se trata de adivinar la letra que ha rodeado al adversario.
- Puedes puntear o dibujar flechas en las naves para recordar los disparos a la flota contraria.
- Cuando se dice la letra y se falla, el receptor debe indicar si su barco oculto está situado en una letra "a la izquierda" o a "la derecha" siguiendo el orden alfabético de la A a la Z.
- ¿Cuántos tiros necesitamos para localizar la ubicación del barco?

EL CAMINO MÁS CORTO

Instrucciones: Encuentra el camino más corto para llegar desde A hasta F. Anota varias opciones y el valor total de cada uno. Investiga en Internet en qué consiste el **ALGORITMO DE DIJKSTRA** y aplícalo.



Algoritmo de Dijkstrad



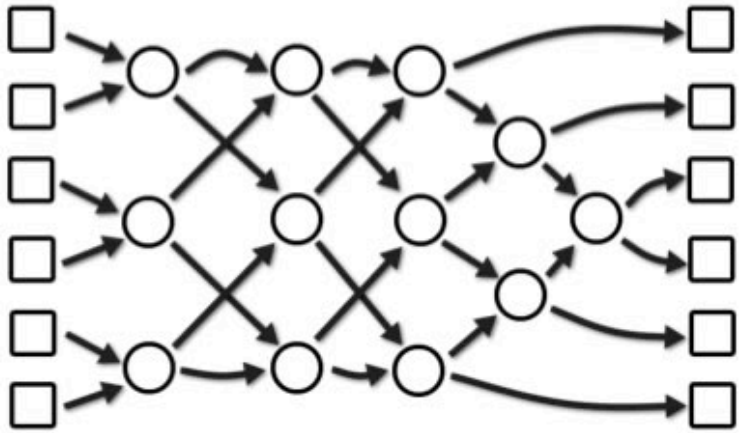
- Con pequeños: probar y valorar distintas rutas.
- Con mayores: Buscar información y aplica algoritmo de Dijkstrad.

Ordenación por árbol

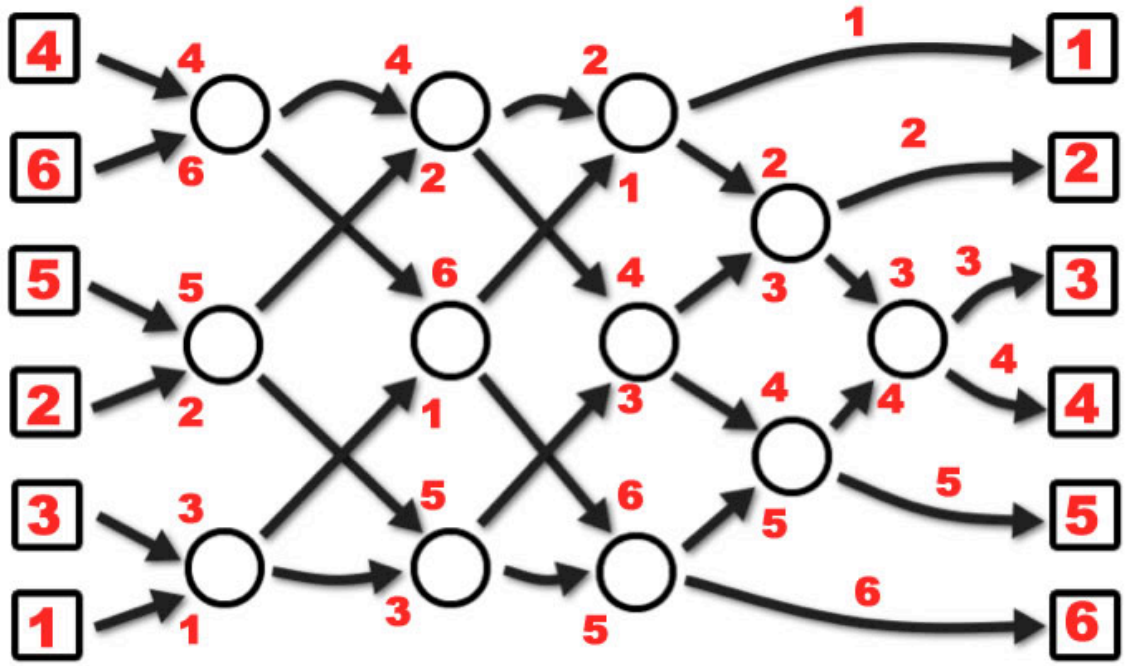
Adaptación de la actividad "Redes de ordenamiento" del manual 2008 Computer Science Unplugged
Fuente: csunplugged.org/

Instrucciones:

- Sitúa los números del 1 al 6 desordenados en los recuadros de la columna izquierda.
- **Consigna**: Al llegar los dos números a un círculo (a una plaza) el más pequeño se va por la calle de arriba y el más grande por la calle de abajo.
- Observa que al final los números aparecen ordenados en los recuadros de la columna derecha con independencia del grado de desorden inicial.



Ordenación por árbol



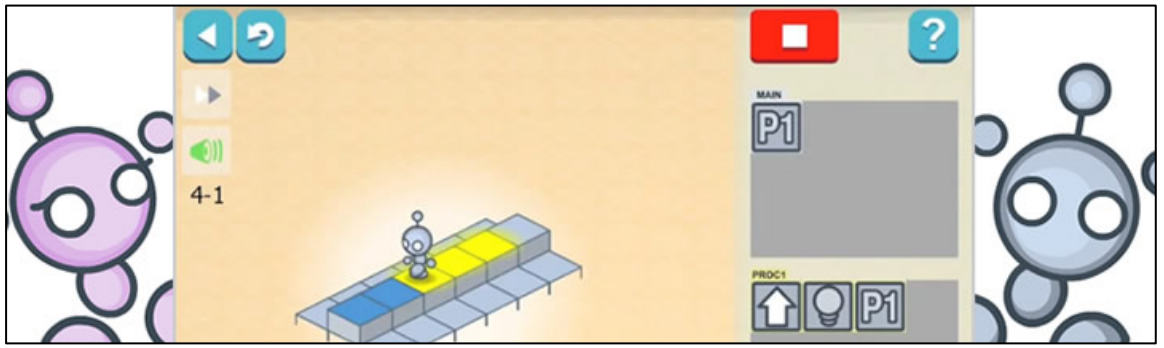
4 Recursos para PC CON dispositivo y SIN robot

CON PC
CON Tabletas
SIN Robot

App Lightbot One Hour

- Juego de programación visual de 20 niveles.
- Resolver situaciones problema mediante la conexión de una secuencia de bloques.
- Aplicación en línea acccesible desde el navegador o App.

lightbot.com/hour-of-code.html



App Bit by Bit

- Es un juego pensando para el aprendizaje del pensamiento computacional.
- Se puede jugar sin necesidad de leer ni contar.
- Se proponen situaciones para resolver aplicando codificación, anticipación, secuenciación, simultaneidad, división en problemas más sencillos ...

Code.org

- Fundación internacional para la promoción del PC. Propone interesantes tutoriales autoguiados y secuencias de trabajo organizados por edades.
- Promueve la iniciativa “La hora del código”.

Blockly Games

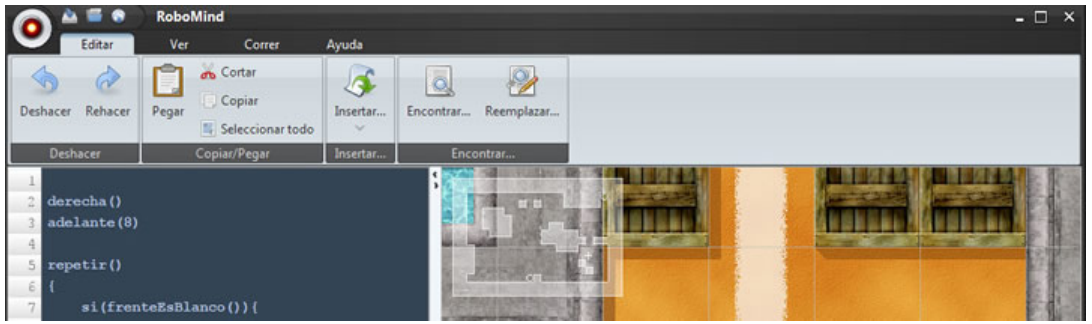
blockly-games.appspot.com

- Recopilación de juegos para desarrollar el pensamiento computacional.
- Más adecuado para alumnado de Primaria.
- Descarga local: bit.ly/blockygames

RoboMind

www.robomind.net/es

- Lenguaje de programación textual donde se programa el comportamiento de un robot.
- Dispone de un amplio repertorio de acciones, bucles, condicionales, procedimientos, etc.
- Ideal para Secundaria y Universidad



Code Combat

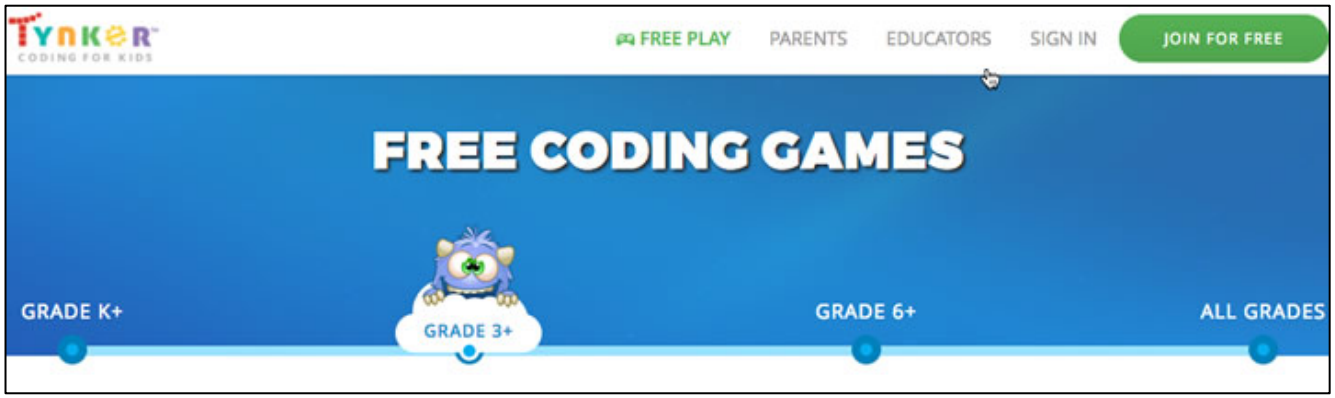
codecombat.com

- Juego de aventuras basado en programación textual.
- Se trata de programar el movimiento del avatar por los distintos escenarios.
- Para Primaria y Secundaria.

Tynker

www.tynker.com

- Editor en línea o bien app para Android/iOS
- Basado en programación visual para el diseño de juegos y aplicaciones.
- Para alumnado a partir de 8 años.



Alice

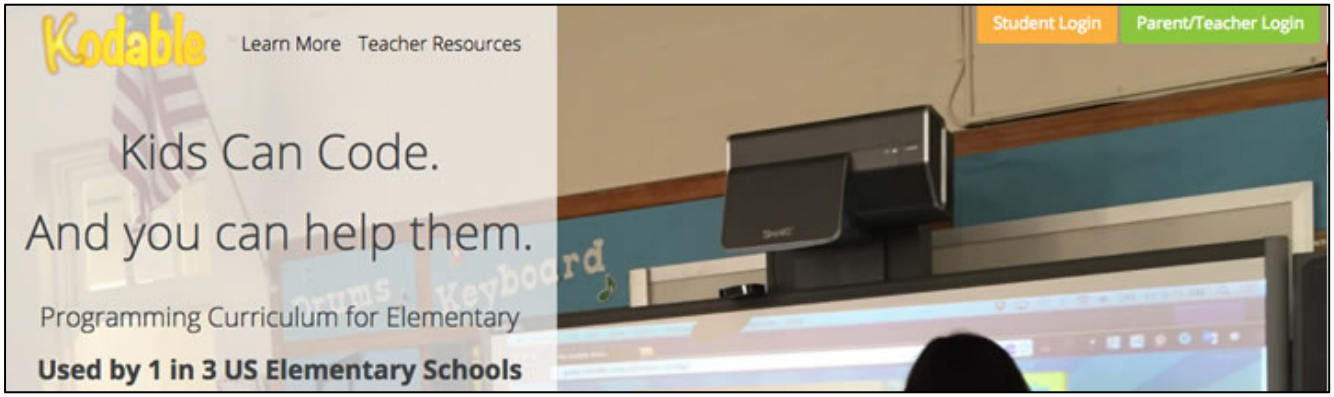
www.alice.org

- Software gratuito que se instala en el ordenador.
- Está orientado a la creación de animaciones interactivas, narración de historias, etc
- Para alumnado de Bachillerato y Universidad

Kodable

www.kodable.com

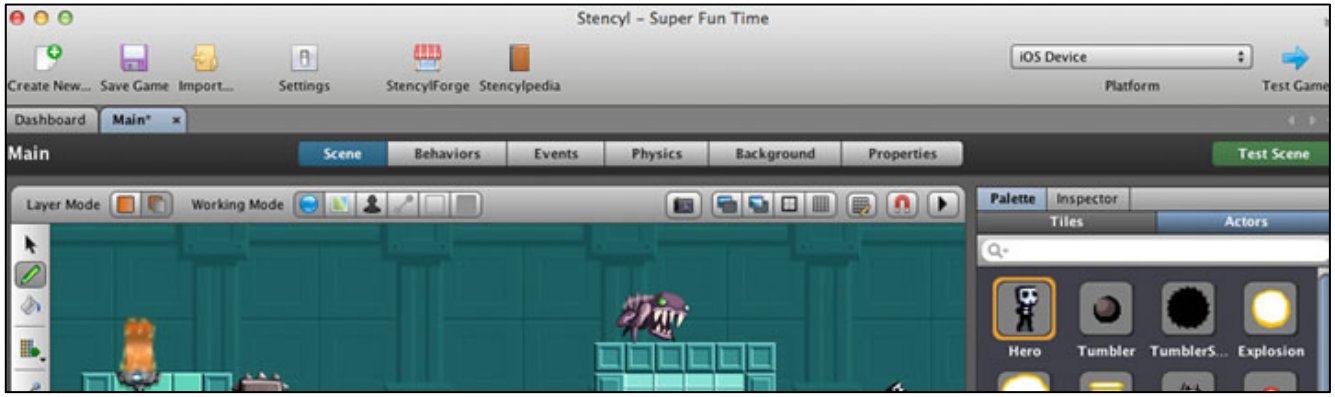
- Propone una secuencia progresiva de ejercicios para trabajar conceptos computacionales en función de la edad
- Tiene versión web y versión app para móviles.
- Para Primaria y Secundaria



Stencyl

www.stencyl.com

- Software multiplataforma instalable para crear juegos mediante programación visual.
- Para Secundaria.



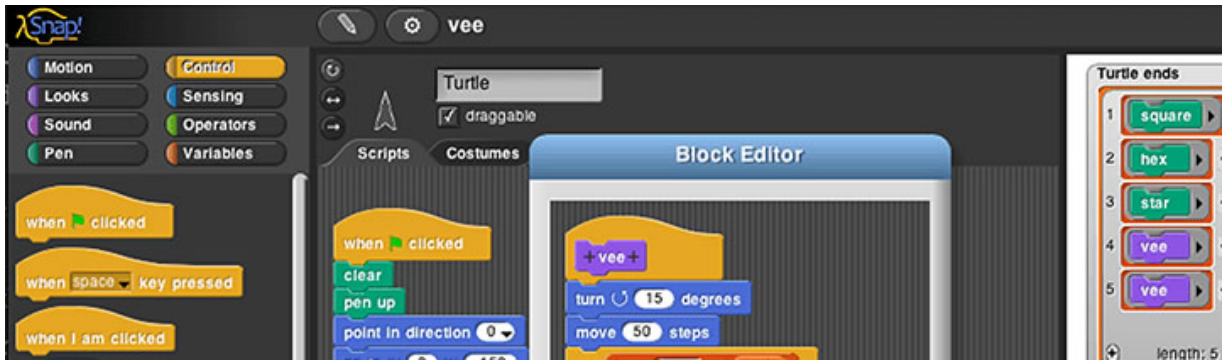
CodeHunt www.codehunt.com

- Juego en línea para aprender a programar.
- El cazador de código acumula puntos al descubrir el fragmento de código perdido.
- Para Bachillerato y Universidad

Snap!

snap.berkeley.edu

- Implementación extendida de Scratch
- Permite construir nuevos bloques (procedimientos)
- Ideal para alumnado de Secundaria y Universidad



5

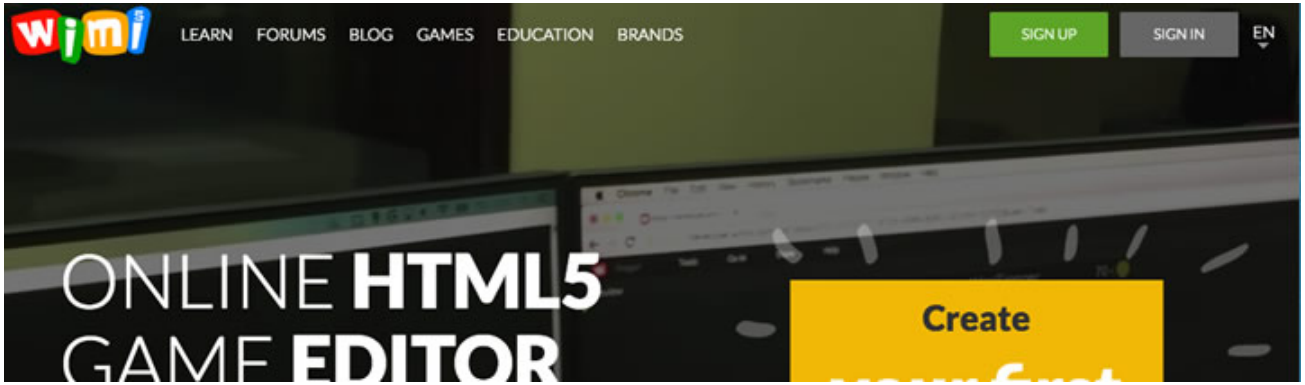
Herramientas para crear aplicaciones

Orientado a producto digital

WIMI5

www.wimi5.com

- Editor en línea gratuito para crear y publicar videojuegos en formato HTML5
- La lógica del juego se construye conectando bloques funcionales.
- Para Bachillerato y Universidad



“Taller de WIMI5”

canaltic.com/talleres/wimi5

- Taller para alumnado donde se propone el diseño de juegos interactivos: movimiento, sprites, control teclado y pantalla, scroll, colisiones, contadores, etc.
- Metodología: Descubrimiento Guiado y Resolución de Problemas.

Mit App Inventor 2

ai2.appinventor.mit.edu

- Herramienta gratuita, completa y basada en programación visual para el diseño de apps para dispositivos móviles Android.
- Se basa en la programación visual como Scratch.
- AI2 representa una evolución natural de Scratch en el desarrollo del pensamiento computacional.

“Taller de MIT App Inventor 2”

canaltic.com/talleres/ai2

- Taller para alumnado para el diseño de app tipo siguiendo un proceso creativo completo: diseño, programación y publicación.
- Metodología: Descubrimiento Guiado y Resolución de Problemas.

6 Scratch

Estándar de programación visual

ScratchJr

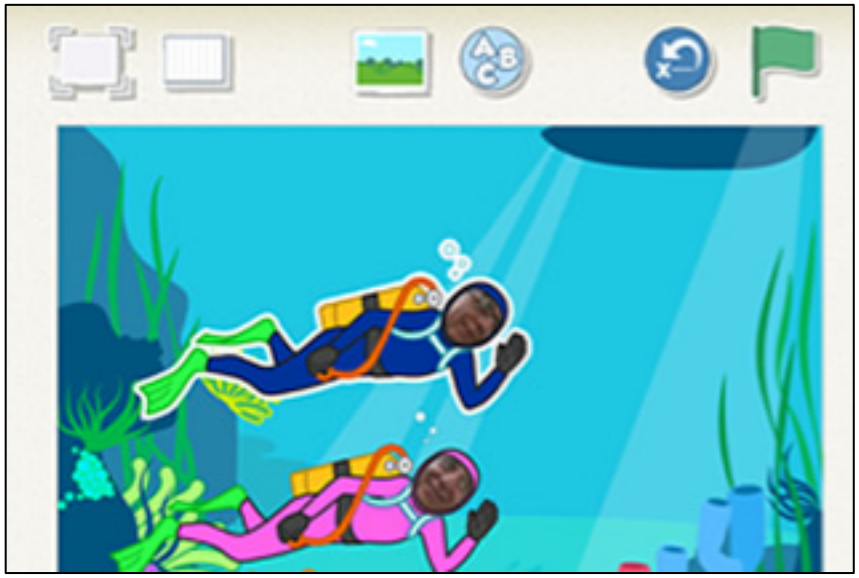
- App Android/iOS para la iniciación a la programación de niños/as de 5-7 años: **www.scratchjr.org**
- Inspirado en Scratch para crear historias e interactivos.
- Permite editar imágenes, añadir tus propios audios, incorporar tus propias imágenes, combinar escenas, ...

“Aprendemos a programar con ScratchJr”

canaltic.com/talleres/sjr

- Taller para alumnado con proyectos: Surf en la playa, La rana saltarina, Carrera de coches y Paseando a Toby.
- Metodología: Descubrimiento Guiado y Resolución de Problemas.

Paseo submarino con ScratchJr

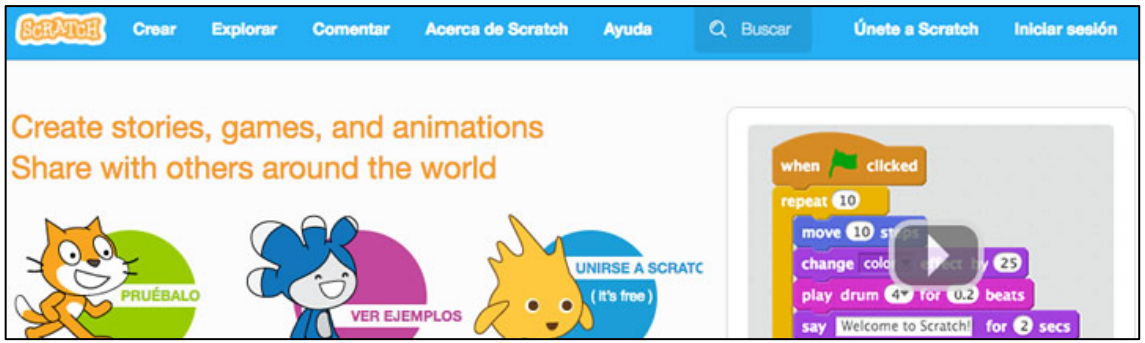


- Desplazamientos: trayectorias, direcciones y sentidos.
- Eventos: al pulsar bandera verde.
- Procesos simultáneos o paralelos.
- Concepto de velocidad.
- Diseño personalizado de objetos.
- Inmersión en la realidad ficticia.

Scratch

scratch.mit.edu

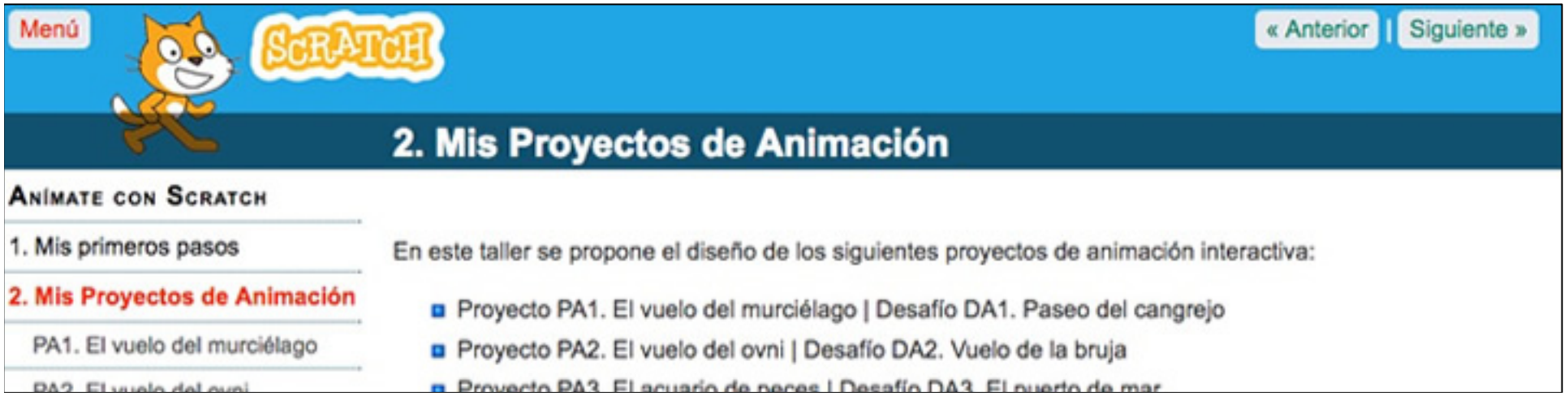
- Entorno de programación visual pensado para jóvenes
- Permite la introducción al pensamiento computacional y la expresión creativa de ideas mediante diseño de animaciones.
- Software gratuito desarrollado por el MIT.



“Anímate con Scratch”

canaltic.com/talleres/scratch

- Taller para alumnado con proyectos de Animación, de Narración y de Geometría.
- Metodología: Descubrimiento Guiado y Resolución de Problemas.

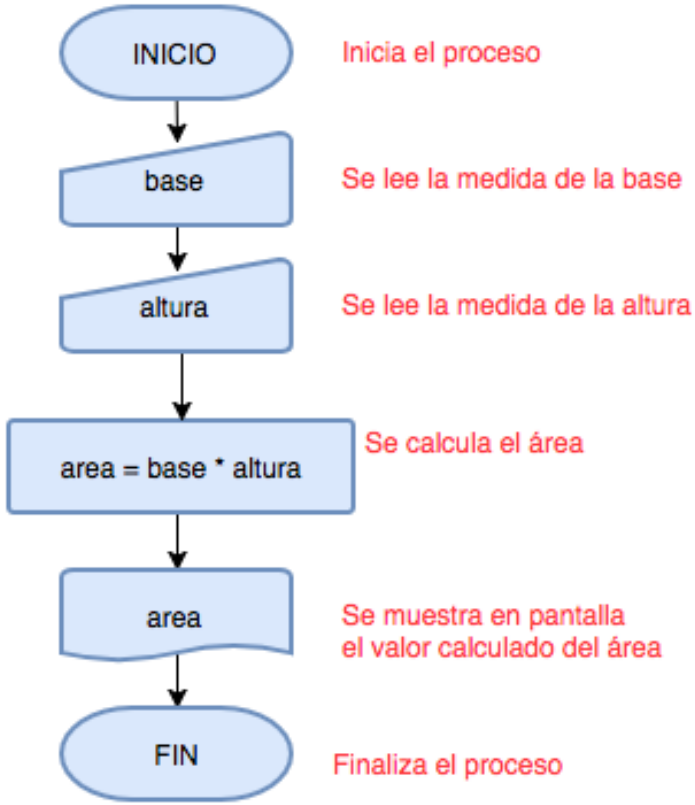


Scratch

- Herramienta muy completa de programación visual.
- Permite trabajar con el alumnado los **conceptos computacionales básicos**.
- Admite **variables y operaciones** por lo que se puede aplicar a la resolución de **problemas de la vida cotidiana**.
- El proceso creativo debe utilizar el **diseño y representación del algoritmo** como paso previo a la **codificación**.

Problema 1.
Área de un rectángulo
(Secuencia)

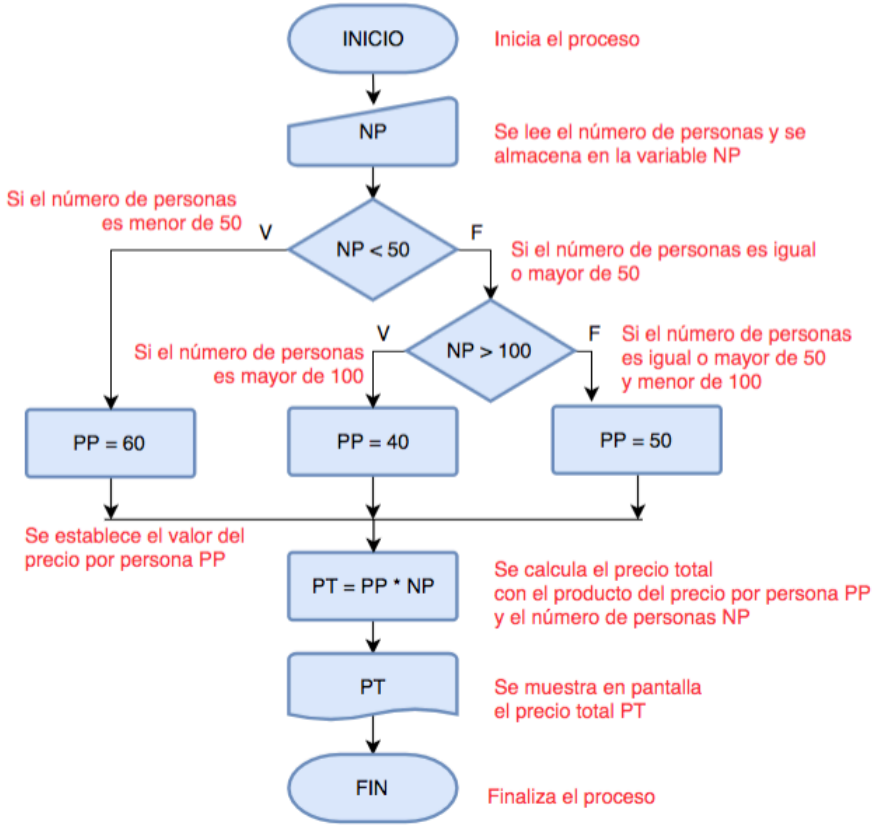
Calcula el área de un rectángulo sabiendo
la medida de su base y de su altura


INICIO
Inicia el proceso
base
Se lee la medida de la base
altura
Se lee la medida de la altura
area = base * altura
Se calcula el área
area
Se muestra en pantalla
el valor calculado del área
FIN
Finaliza el proceso

al presionar
preguntar Introduce la BASE y esperar
fijar base a respuesta
preguntar Introduce la ALTURA y esperar
fijar altura a respuesta
fijar area a base * altura
decir area por 10 segundos

Problema 2.
Banquete en un restaurante
(Decisiones)

En un restaurante que organiza banquetes colectivos cobran 60 €/ persona si el número de comensales es inferior a 50; 50 €/persona si está entre 50 y 100 y 40 €/persona si es superior a 100 invitados. Calcula el presupuesto total en función del número de personas.


INICIO
Inicia el proceso
NP
Se lee el número de personas y se almacena en la variable NP
Si el número de personas es menor de 50
V
NP < 50
F
Si el número de personas es igual o mayor de 50
V
NP > 100
F
Si el número de personas es mayor de 100
Si el número de personas es igual o mayor de 50 y menor de 100
PP = 60
PP = 40
PP = 50
Se establece el valor del precio por persona PP
PT = PP * NP
Se calcula el precio total con el producto del precio por persona PP y el número de personas NP
PT
Se muestra en pantalla el precio total PT
FIN
Finaliza el proceso

al presionar
preguntar Introduce el NUMERO PERSONAS y esperar
fijar NP a respuesta
si NP < 50 entonces
fijar PP a 60
si no
si NP > 100 entonces
fijar PP a 40
si no
fijar PP a 50
fijar PT a PP * NP
decir PT por 10 segundos

7

Recursos para PC CON dispositivo y CON robot

Robótica Educativa

Robótica educativa

- La robótica en el aula es una forma de trabajar el PC.
- Se programa comportamiento de un sistema físico.
- Se sobrepasan los límites de la pantalla digital.
- Muy atractivo para el alumnado.

Diseño y programación

Construcción de Modelos

Sin Diseño

Diseño **dirigido**

Diseño **creativo Maker**

Programación de Rutinas

Visual

Textual

Control del robot

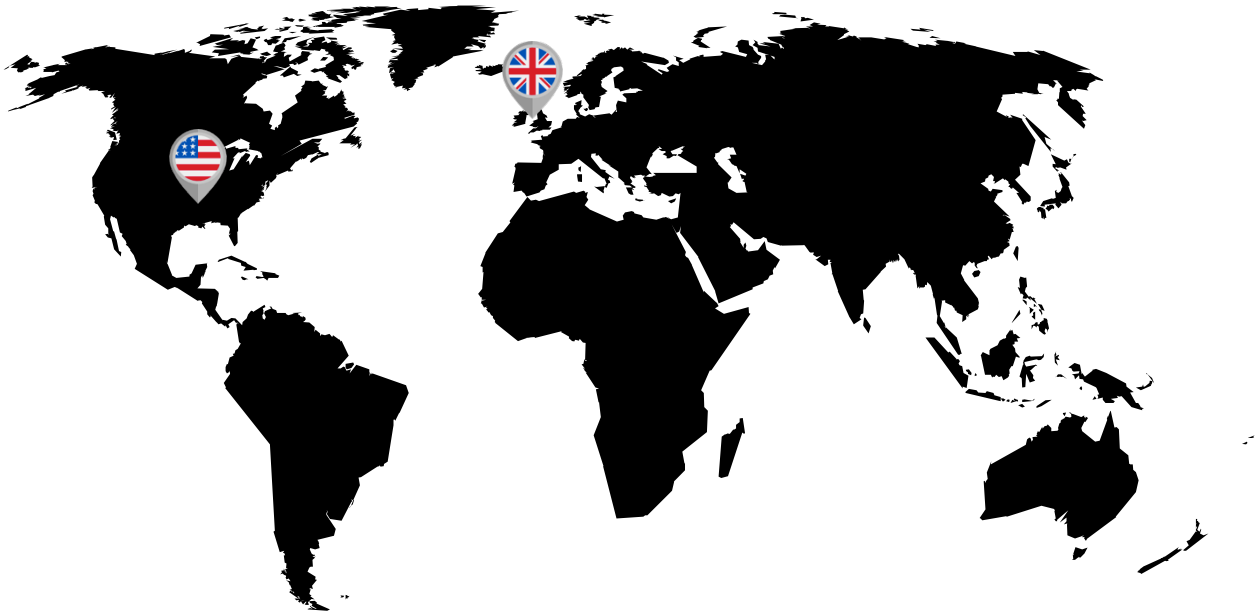
- Casi todas las soluciones de robótica educativa proponen una interfaz visual de conexión de bloques para diseñar la programación
- El programa se crea directamente en el robot o en un ordenador/tableta y se le envía por wifi, bluetooth o cable usb

Algunas soluciones de robótica

- Beebot
- BlueBot
- Jack Mouse
- Next 2
- CodiOruga
- Aisoy
- Lego WeDo 2
- mBot
- Moway
- Dash&Dot
- Zowi
- InoBot
- OzoBot
- Lego EV3
- Edison 2
- BQ Print Ev
- Fisher
- Arduino
- ...

8 Criterios didácticos

para la enseñanza y aprendizaje del
Pensamiento Computacional



El Pensamiento Computacional
en el currículum oficial de UK y USA

Asignatura COMPUTING en UK

- Sustituye a la asignatura ICT = Tecnologías de la Información y la Comunicación
- En Primaria (Key Stage 1 y 2) y en Secundaria (Key Stage (3 y 4)
- Abarca 3 áreas:
  - CIENCIAS DE LA COMPUTACIÓN
  - TECNOLOGÍAS DE LA INFORMACIÓN
  - ALFABETIZACIÓN DIGITAL

Objetivos COMPUTING en UK

- Conocer y aplicar conceptos de computación.
- Analizar y resolver problemas mediante programación.
- Evaluar y aplicar TIC emergentes de forma analítica para resolver problemas.
- Ser responsables, competentes, seguros y creativos TIC.

Modelo CSTA en USA

- **CSTA** = Computer Science Teachers Association.
- Propósito: promover computación en **K-12.**
- Modelo de estándares K-12 para trabajar **Computer Science.**
- **Estándares** = descripciones de lo que deben aprender los alumnos al finalizar una etapa.
- Se especifican estándares por tramos de edades: **K-2** (5-8 años), **K-5**, **K-8**, **K-10** y **K-12** (16-18 años).

Los estándares CSTA

Los estándares se organizan en **5 ejes conceptuales**:

- Sistemas de computación.
- Redes e Internet.
- Algoritmo y programación.
- Datos y análisis.
- Impacto de la computación.

Los ejes prácticos CSTA

Los estándares se clasifican en **7 ejes prácticos**:

- Fomento de la cultura computacional inclusiva.
- Colaboración.
- Reconocimiento y definición de problemas computacionales.
- Desarrollo y uso de abstracciones.
- Creación de artefactos computacionales
- Testeo y depuración.
- Comunicación sobre computación.

Ejemplo K-2 del CSTA

Ejemplo:

- **Edad**: K-2.
- **Estándar**: Analizar y depurar un algoritmo que incluye secuencias y bucles sencillos, con y sin dispositivo computacional.
- **Concepto**: Algoritmos y programas.
- **Práctica**: Testeo y depuración.

¿Y en mi aula?

El reto del pensamiento computacional:
Integración transversal

El reto del Pensamiento Computacional

El principal reto para poder incorporarlo al aula es el diseño y puesta en práctica de proyectos en los cuales el pensamiento computacional se integra de una **forma transversal** a todas las áreas para favorecer un aprendizaje constructivo, conectivo, competencial y tecnológico.

Computacional Thinking Teacher Resources

- Informe con 9 ejemplos de experiencias de aprendizaje [3].
- El pensamiento computacional como enfoque transversal y basado en el aprendizaje por proyectos.
- Ejemplo: “Cadena alimentaria” Ciencias Nivel K-4 (9-10 años). Crear una animación en Scratch de una cadena.

9 Sugerencias metodológicas

para la enseñanza y aprendizaje del
Pensamiento Computacional y la Robótica

Sugerencias metodológicas I

En “**Programar para aprender**” de Código 21 (2014):

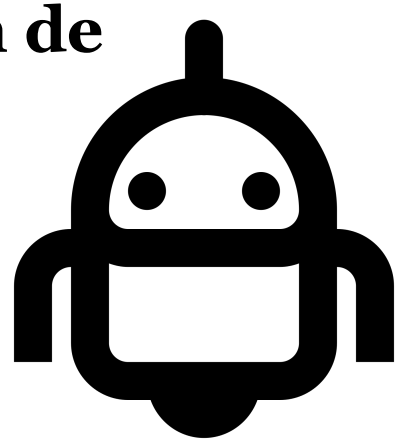
- Estimular la creatividad (computación creativa)
- Adaptarse al contexto en las situaciones
- Trabajar la exposición oral y el trabajo en grupo.
- Promover reflexión y sentido crítico sobre los productos.

Sugerencias metodológicas II

- Remezclar y reutilizar respetando derechos de autor.
- Trabajar sin ordenador.
- Buscar audiencia a los productos del alumnado.
- Tener en cuenta la inclusión de género.
- Aprovechar las interfaces multilingües.
- Evaluar: portfolio propio, entrevista y capacidad de analizar otros proyectos.

Idea final

En la actualidad disponemos de recursos para crear una **propuesta variada, adaptada y progresiva** a lo largo de la escolarización del alumnado donde se incorpore el **pensamiento computacional y la robótica** de forma **transversal** como un **sistema de resolución de problemas.**



Clasificación de recursos PC por edades

Recurso	INF 3-5 a	PRI 6-9 a	PRI 10-12 a	SEC 13-15 a	BAC 16-18 a
App Blue-Bot	X	X			
App-Web Lightbot One Hour	X	X			
App Bit by Bit		X	X		
Web Code.org	X	X	X	X	X
Web-Local Blocky Games		X	X		
Web RoboMind				X	X
Web Code Combat			X	X	X
App-Web Tynker			X	X	
Web Alice				X	X
Web Kodable			X	X	
Web Stencyl				X	X
Web CodeHunt				X	X
Web Snap!				X	X
Web WIMI5				X	X
Web MIT App Inventor 2			X	X	X
App ScratchJR	X	X			
Web-PC Scratch			X	X	X

Créditos

Pensamiento Computacional en el Aula

Reunión de Coordinadores TIC
20-nov-2017

Fernando Posada Prieto
fernandoposada@gmail.com
Centro del Profesorado de Lanzarote
Noviembre 2017

Licencia de uso de esta presentación: CC BY-NC-SA
Fotografías obtenidas de freepik.com