



Computational Thinking Unplugged

Pensamiento computacional desenchufado

Miguel Zapata-Ros

Universidad de Murcia, Murcia (España)

<http://orcid.org/0000-0003-4185-5024>mzapata@um.es

ARTICLE INFO

Key words:

Computational thinking
Computational thinking unplugged
Evocation principle
First principles of learning
Learning activities
KIBO
Bee bot
Cs unplugged
Playmaker

Palabras clave:

Pensamiento computacional
Pensamiento computacional desenchufado
Evocación
Principales principios de aprendizaje
Actividades de aprendizaje
KIBO
Bee bot
Cs unplugged
PlayMaker

ABSTRACT

The idea of computational thinking unplugged refers to a set of activities that are developed to encourage children skills that can be recalled later, to promote computational thinking. These activities are designed to be included in the early stages of cognitive development (early childhood education, the first stage of primary education, games at home with parents and friends, etc.). The skills are designed so that they can be evoked in other stages. In secondary education, in technical training, in professional or even higher education. The activities are usually done without computers and mobile screens, with cards, cards, classroom games or playground games, mechanical toys, etc. In this paper, it is highlighted that there is a series of data, ideas and circumstances that make it possible now, and not before, that unplugged computational thinking is implemented. Finally, we describe activities, initiatives and experiences that are already being developed, and we make proposals for activities and their guides for preschool teachers and caregivers.

RESUMEN

La idea de pensamiento computacional desenchufado (*Computational thinking unplugged*) hace referencia a un conjunto de actividades que se elaboran para fomentar en los niños habilidades que pueden ser evocadas después, para favorecer el pensamiento computacional. Estas actividades están pensadas y diseñadas para ser incluidas en las primeras etapas de desarrollo cognitivo (educación infantil, primer tramo de la educación primaria, juegos en casa con los padres y los amigos, etc.). Las habilidades están pensadas para que puedan ser evocadas en otros ciclos y niveles educativos, en la educación secundaria, en la formación técnica, en la profesional o en la educación universitaria incluso. Las actividades se suelen hacer sin ordenadores y sin pantallas móviles, con fichas, cartulinas, juegos de sala de clase o juegos de patio, juguetes mecánicos, etc. En este trabajo se pone de relieve que hay una serie de datos, ideas y circunstancias que hacen posible ahora, y no antes, que se implemente el pensamiento computacional desenchufado. Por último, describimos actividades, iniciativas y experiencias que se están desarrollando ya, y hacemos unas propuestas de actividades y de sus guías para profesores y cuidadores de preescolar.

1. Introducción

La idea de pensamiento computacional desenchufado (*Computational thinking unplugged*) hace referencia al conjunto de actividades, y su diseño educativo, que se elaboran para fomentar en los niños, en las primeras etapas de desarrollo cognitivo (educación infantil, primer tramo de la educación primaria, juegos en casa con los padres y los amigos, etc.) habilidades que luego pueden ser evocadas para favorecer y potenciar un buen aprendizaje del pensamiento computacional (González-González, 2019) en otras etapas o en la formación técnica, profesional o en la universitaria incluso. Actividades que se suelen hacer con fichas, cartulinas, juegos de salón o de patio, juguetes mecánicos, etc.

Hay una serie de datos, ideas y circunstancias que avalan un trabajo como este, y hacen posible ahora que se implementen actividades, iniciativas y experiencias de pensamiento computacional desenchufado.

Como anécdota, pero que es importante señalar a la hora de tener en cuenta cómo se ha forjado este trabajo y en qué contexto lo ha hecho, hay que decir que se ha originado en las últimas actividades realizadas por el autor, a socaire de un ciclo organizado por ANEP-Uruguay y las Fundaciones Telefónica de Uruguay y de Argentina (ver los vídeos en <https://youtu.be/b7zvtibGOT0> y en https://youtu.be/yOSGEM7x_7k) de conferencias, talleres y eventos rurales en colegios y pueblos del interior de Uruguay, con niños y maestros de Primaria y de Secundaria. Pero sobre todo en el microclima que se creó en Arequipa, en el CITIE 2018 (Villalba-Condori, García-Peñalvo, Lavonen y Zapata-Ros, 2019), con la concurrencia de referentes muy relevantes en lo que ha sido el desarrollo del pensamiento computacional, con su experiencia y testimonio, como son entre otros Artemis Papert y Margaret Minsky, junto con otras individualidades que trabajan el tema.

En el ambiente de charlas, discusiones, exposiciones de ideas hubo una comunidad de enfoques que, además de modificar los esquemas de cada uno abrió espacios de desarrollo en el trabajo pienso de cada uno, al menos en mi caso así fue. Y también explícitamente en el caso de Artemis para colaborar con sus ideas y con sus creaciones gráficas, a partir de Turtle Art y Tortugarte, en la elaboración de materiales para los ciclos iniciales de la educación. Ya veremos lo que el futuro depara.

Ese era el pues el ambiente que reinaba por las dependencias y los espacios de CITIE 2018. Ahí nacieron una serie de compromisos y una serie de ideas que voy a desarrollar en esta recensión de ideas, recursos y propuestas prácticas sobre Pensamiento Computacional en las primeras etapas.

Para disipar posibles equívocos desde el principio se propone titular este documento y nombrar el constructo como “Pensamiento computacional desenchufado” (eso es algo más que desconectado). En inglés ya existe la expresión “Computer Science Unplugged”. La ha utilizado Tim Bell (Bell y Vahrenhold, 2018) de la University of Canterbury de Nueva Zelanda, con algunas diferencias prácticas y conceptuales que ya veremos. En definitiva, queremos poner de relieve que lo importante es que los niños no jueguen con trastos, no solo digitales, sino ni tan siquiera enchufados... y que a pesar de ello adquieran pensamiento computacional. O quizás por ello.

En este trabajo pues vamos a abordar la necesidad y la conveniencia de trabajar aspectos del aprendizaje previo, convergente con el pensamiento computacional y necesario para él, desde las primeras etapas del desarrollo cognitivo de los individuos. Lo vamos a justificar desde el punto de vista de la teoría del aprendizaje y del de una pedagogía necesaria a ese fin. Vamos a hacerlo desde el punto de vista experiencial.

Para ello se van a proponer un tipo de actividades, a este nivel solo de forma enunciativa y descriptiva, pero en lo posible lo más indicativa posible de cómo debería ser una propuesta más elaborada, y sobre todo la necesidad de guías para profesores de estos ciclos que doten de valor pedagógico en el sentido de los objetivos propuestos para el pensamiento computacional. Orientar el diseño y la práctica en este sentido.

Hemos sostenido en trabajos anteriores (Zapata-Ros, Noviembre 2014 y Diciembre 2014) esa necesidad sobre la base de una perspectiva y de una opción, desde el punto de vista de que se trata de una nueva alfabetización, y que como tal el pensamiento computacional debe constituir una competencia o una serie de competencias claves en igualdad a como lo son las otras, las competencias claves de la alfabetización tradicional, la de la época industrial: La lectura, la escritura, el cálculo elemental y la geometría.

Lo hacíamos directamente (Zapata-Ros, Diciembre 2014):

Las sociedades más avanzadas han visto que se trata de una nueva alfabetización, la alfabetización digital, y que como tal hay que comenzar desde las primeras etapas del desarrollo individual, al igual como sucede con otras habilidades clave: la lectura, la escritura y las habilidades matemáticas.

O bien citando informes y directrices europeas (Balanskat y Engelhardt, 2015):

La codificación (coding) es cada vez más una competencia clave que tendrá que ser adquirida por todos los jóvenes estudiantes y cada vez más por los trabajadores en una amplia gama de actividades industriales y profesiones. La codificación es parte del razonamiento lógico y representa una de las habilidades clave que forma parte de lo que ahora se llaman “habilidades del siglo 21”.

2. Principio de activación

Las destrezas del pensamiento computacional no podemos esperar que aparezcan de forma espontánea en el mismo momento en que se necesitan, en los estudios de grado o de secundaria superior.

Las habilidades que son necesarias para la programación de algoritmos complejos, las destrezas del pensamiento computacional en todo su vigor, es decir, las que son necesarias para la programación de ordenadores, para resolver problemas, o para organizar el proceso y la circulación de datos, así como para que los ordenadores realicen tareas para las que están contruidos, estas habilidades, no podemos esperar a que aparezcan, o a se manifiesten de forma espontánea. Y que lo hagan en el mismo momento de necesitarlas en los estudios de grado de Computación o de Ingeniería Informática, en la etapa de madurez del alumno que corresponde a esa edad, ni tan siquiera en la etapa de desarrollo del pensamiento abstracto, en la secundaria postobligatoria o incluso en Secundaria. Un análisis de estas habilidades lo podemos encontrar en los informes de Velázquez-Iturbide (2018) y de Velázquez-Iturbide y otros (2018).

En esto estas habilidades no son distintas de otras habilidades complejas que tienen que ver con el desarrollo de los individuos, que se adquieren de forma progresiva y que solo son utilizables en forma operativa en su última fase.

Esta naturaleza del aprendizaje, el enlace de las situaciones de aprendizaje con los objetivos finales a través de etapas, niveles y condiciones de aprendizaje, es la que justifica el diseño instruccional y de ello no se libra la adquisición de las habilidades computacionales ni, siendo distinto, el pensamiento computacional. Los aprendizajes complejos se dividen, se fraccionan en aprendizajes más simples, más cercanos a las capacidades de los individuos y más lejanos del momento que adquieren su mayor eficiencia o su mayor operatividad práctica, o incluso que nunca lo alcancen porque no exista, como sucede en el caso que no lo alcancen ese punto en su dominio propio, por sí mismas, sino como habilidades auxiliares a otras. Así pasa con los conocimientos y las habilidades básicas y con las competencias clave.

En este punto es donde obtienen su justificación en las teorías del aprendizaje, en el principio de activación, y en la forma en como transitar desde que se adquieren las habilidades hasta que son útiles en su destino final. Este tránsito y la forma de organizarlo es lo que constituye la base del diseño instruccional. Por tanto, son dos núcleos clave que está en la justificación en la teoría del aprendizaje y en la base de una pedagogía del pensamiento computacional: El principio de activación y el diseño instruccional.

En esta parte nos vamos a dedicar exclusivamente al principio de activación. La otra cuestión, la consideración del pensamiento computacional como competencia clave de la nueva alfabetización tampoco la abordaremos en este punto, es una elaboración o una consecuencia elaborada del principio de activación.

Así pues, vamos a justificar con este principio la necesidad y la conveniencia de trabajar aspectos del aprendizaje previos, convergente con el pensamiento computacional y necesarios para él, desde las primeras etapas del desarrollo cognitivo de los individuos. Es lo que va a justificar después qué actividades y cómo se organizan juegos en la infancia para que habilidades de secuenciación o de encaje, entre objetos computacionales o entre variables y tipos de datos, por ejemplo, se activen y fluyan en la fase de resolver problemas con algoritmos y programas en las etapas de enseñanza profesional o universitaria. Esto obviamente sería una ejemplificación extrema. En un caso más normal, la adquisición se produciría de una forma más progresiva a través de las distintas etapas educativas, los niveles e incluso dentro de estos y de los módulos y unidades instruccionales que los componen.

En su trabajo, Merrill (2002) desarrolla lo que llama unos principios fundamentales del aprendizaje (*first principles*) lo hace decantando los principios subyacentes en los que hay consensos, en los que hay un acuerdo esencial, en todas las teorías y que previamente ha identificado. Ese trabajo está expuesto y desarrollado en su trabajo *First principles of instruction* (Merrill, 2002). en *Educational technology research and development*, incluido como capítulo en el tercer volumen de los libros de Reigeluth *Instructional-design theories and models: Building a common knowledge base* (Merrill, 2009). Y de forma resumida en *First principles of instruction: A synthesis* (Merrill, 2007). También son glosados como base del nuevo paradigma instruccional de Reigeluth, cuya versión oficial pueden encontrar en RED número 50, en el artículo *Teoría instruccional y tecnología para el nuevo paradigma de la educación* (Reigeluth, 2016).

En este último trabajo, Reigeluth (2016) distingue entre principios universales y escenarios particulares. Cuando aplicamos con mayor precisión un principio o un método instruccional, por lo general descubrimos que hace falta que éste sea diferente para diferentes situaciones y perfiles de aprendizaje, o una mayor precisión para obtener objetivos contextualizados y personalizados. Reigeluth (1999) se refirió a los factores contextuales que influyen en los efectos de los métodos como “escenarios”.

Los principios fundamentales de instrucción (*first principles*) los propone y los define Merrill (2002) en *First principles of instruction*. Este documento se refiere a los métodos variables como programas y prácticas. Un *principio fundamental* (Merrill, 2002), o un método básico según Reigeluth (1999a), es un aserto que siempre es verdadero bajo las condiciones apropiadas independientemente del programa o de la práctica en que se aplique, que de esta forma dan lugar a un método variable. Teniendo en cuenta como el mismo Merrill (2002) las define:

Una práctica es una actividad instruccional específica. Un programa es un enfoque que consiste en un conjunto de prácticas prescritas. Las prácticas siempre implementan o no implementan los principios subyacentes ya sea que estos principios se especifiquen o no. Un enfoque de instrucción dado solo puede enfatizar la implementación de uno o más de estos principios de instrucción. Los mismos principios pueden ser implementados por una amplia variedad de programas y prácticas.

De esta forma Merrill propuso un conjunto de cinco principios instruccionales prescriptivos (o “principios fundamentales”) que mejoran la calidad de la enseñanza en todas las situaciones (Merrill, 2007, 2009). Esos principios tienen que ver con la centralidad de la tarea, la *activación*, la demostración, la aplicación y la integración.

Para ello Merrill (2002) propone un esquema en fases como el más eficiente para el aprendizaje, de manera que centran el problema y crean un entorno que implica al alumno para la resolución de cualquier problema. En cuatro fases distintas, cuando habitualmente solo se hace en una: la de demostración, reduciendo todo el problema a que el alumno pueda demostrar su conocimiento o su habilidad en la resolución del problema en una última fase. Son las FASES DE INSTRUCCIÓN (ver Figura 1).

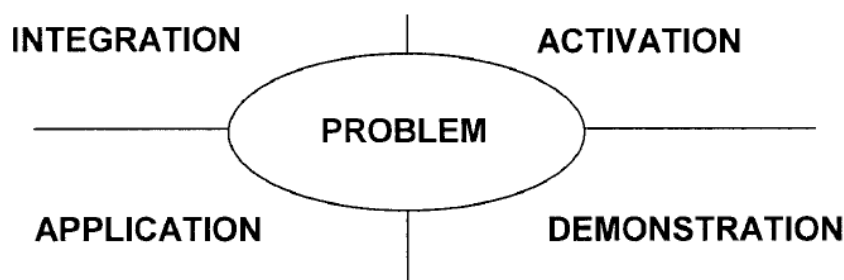


Figura 1. Fases de la instrucción según Merrill (2002)

Las fases son (a) activación de experiencia previa, (b) demostración de habilidades, (c) aplicación de habilidades, y (d) integración de estas habilidades en actividades del mundo real.

Así la figura anterior proporciona un marco conceptual para establecer y relacionar los principios fundamentales de la instrucción. De ellos uno tiene que ver con la implicación y la naturaleza real del problema, así percibida por el alumno, y los cuatro restantes para cada una de las fases. Así estos cinco principios enunciados en su forma más concisa (Merrill 2002) son:

1. El aprendizaje se promueve cuando los estudiantes se comprometen a resolver problemas del mundo real. Es decir, el aprendizaje se promueve cuando es un aprendizaje centrado en la tarea.
2. El aprendizaje se favorece cuando existen conocimientos que se activan como base para el nuevo conocimiento.
3. El aprendizaje se promueve cuando se centra en que el alumno debe demostrar su nuevo conocimiento. Y el alumno es consciente de ello.
4. El aprendizaje se promueve igualmente cuando se centra en que el aprendiz aplique el nuevo conocimiento.
5. El aprendizaje se favorece cuando el nuevo conocimiento se tiende a que se integre en el mundo del alumno.

Pero, de todos estos principios, el que justifica sobremedida la inclusión del pensamiento computacional, como *pensamiento computacional desenchufado* en las primeras etapas, es el principio de activación. En él nos vamos a centrar, y no solo en su aplicación para el diseño instruccional en la fase de activación, en la que el conocimiento existente se activa, sino en las fases en las que se crean los conocimientos y habilidades que son activados, y en cómo hacerlo para que la activación sea más eficiente.

En su trabajo *Teoría instruccional y tecnología para el nuevo paradigma de la educación*, Reigeluth (2016, pág. 4) caracteriza el principio de activación de manera que:

- El diseño educativo de actividades, organización, recursos, etc. debe ser tendente a activar en los alumnos estructuras cognitivas relevantes, haciéndoles recordar, describir o demostrar conocimientos o experiencias previas que sean relevantes para él.

- La activación puede ser social. La instrucción debe lograr que los estudiantes compartan sus experiencias anteriores entre ellos.
- La instrucción debe hacer que los estudiantes recuerden o adquieran una estructura para organizar los nuevos conocimientos.

Los trabajos de Merrill (2002) y Reigeluth (2016) hacen énfasis en la fase de evocación, pero no en la fase de crear estructuras cognitivas, experiencias y en general conocimientos y habilidades que puedan ser evocados. Ni tampoco en crear una pedagogía o un diseño educativo que incluya, o tendente a favorecer, elementos cognitivos de enlace que promuevan la evocación. Tampoco a fomentar la investigación sobre estos temas, o a investigar qué tipos de enlaces fortalecen más las estructuras cognitivas de enlace y de evocación.

A partir de lo que dicen, sobre las características del diseño instruccional que implica el principio de activación, los ítems anteriores, podemos concluir que la instrucción, en la fase de crear elementos para ser evocadora, debe:

- Crear estructuras cognitivas que incluyan conocimientos, habilidades, elementos de reconocimiento que permitan distinguir al alumno y otorgar relevancia en su momento de forma fluida a esas habilidades para conseguir su efectividad en ese momento a partir de elementos contextuales, metáforas, etc.
- Otorgar a esas habilidades elementos de reconocimiento que permitan la evocación.
- Asociar esas habilidades a tareas que tengan similitud con las que se en su momento sean necesarias para resolver los problemas a los que ayuda la evocación. En nuestro caso, a los problemas computacionales, o habilidades propias a los elementos que constituyen el pensamiento computacional.
- Diseñar instruccionalmente las actividades que sean relevantes para evocar los elementos de pensamiento computacional (Pérez-Paredes y Zapata-Ros, 2018).
- Propiciar experiencias de aprendizaje compartido en las primeras etapas y hacer que esos grupos y experiencias sociales sean estables a lo largo del tiempo. Las experiencias compartidas crean elementos de activación a través de grupos o de pares alumnos. El propiciar grupos y claves de comunicación, de lenguaje, y que esos grupos sean estables a lo largo del tiempo aumenta la potencia de evocación.
- Crear estructuras cognitivas en los alumnos capaces de recomponerse y aumentar en el futuro. Dotar a los conocimientos y habilidades de referencias y de metadatos que permitan ser recuperados mediante evocación.

Debe pues potenciarse una pedagogía que establezca valores en estas ideas y principios para las primeras etapas.

El principio de activación es pues clave para tenerlo en cuenta cuando se diseña la educación infantil y del primer ciclo de primaria teniendo en el horizonte los aprendizajes futuros, también el Pensamiento Computacional.

Merrill ha sido quien más lo ha trabajado, pero no solo.

Como señalamos en otro trabajo (Zapata-Ros, 2018b), Bawden (2008) habla de habilidades de recuperación, y remite a lo expuestas en otro trabajo anterior (Bawden, 2001). En las habilidades que señala se constatan ideas como la de construir un “bagaje de información fiable” de diversas fuentes, la importancia de las habilidades de recuperación, utilizando una forma de “pensamiento crítico” para hacer juicios informados sobre la información recuperada, y para asegurar la validez e integridad de las fuentes de Internet, leer y comprender de forma dinámica y cambiante material no secuencial. Y así una serie de habilidades donde como novedad se introducen las *affordances* de conocimiento en entornos sociales y de comunicación en redes, y la idea de relevancia. Solo que en este caso son habilidades sobre el proceso de la información, y su posterior recuperación. Obviamente no son habilidades para desarrollar en esta etapa. Sin embargo, sí sería interesante indagar sobre la recuperación de habilidades que se desarrollan mediante juegos de infancia como son habilidades cinestésicas.

3. Pensamiento computacional en la infancia

Desde junio de 2014 hemos argumentado, aportando muy diversas razones, acerca de por qué debían incluirse en el currículo de Educación Infantil y de primaria actividades de Pensamiento Computacional. He aquí un resumen.

En el apartado anterior hemos hablado del principio de activación. Basándonos en él hemos sostenido desde hace tiempo la necesidad de favorecer aprendizajes a través de juegos y de otras actividades que estén cognitiva o cinestésicamente conectadas con las habilidades de computación. También hemos sostenido

que esto es fácilmente asimilable por el público no especializado (Zapata-Ros, 2014): Al igual que sucede con los deportistas y con los músicos, a los niños para que programen bien, o simplemente para que no se vean excluidos de esta nueva alfabetización, que es el pensamiento computacional en la Sociedad del Conocimiento, debería fomentarse en ellos desde las primeras etapas competencias que puedan ser activadas en otras etapas de desarrollo, y en otras fases de la instrucción, correspondientes a las etapas del pensamiento abstracto y a las de rendimiento profesional. Y citábamos el desarrollo de determinadas habilidades, como son las de seriación, encaje, modularización, organización espacial, etc., que, en estudios posteriores de grado, de bachillerato o de formación profesional, pudiesen ser activadas para elaborar procedimientos y funciones en la creación de códigos, o para desarrollar algoritmos propios de esta etapa.

Así lo decíamos el 18 de junio de 2014 en Tumblr (Zapata-Ros, 2014) cuando hablábamos de que no hay que esperar a la universidad ni tan siquiera a la educación secundaria para iniciar el aprendizaje de habilidades de programación, y que al igual a como sucede en otras habilidades instrumentales y claves las habilidades necesarias para la codificación han de ser detectadas y desarrolladas desde las primeras etapas.

O el 5 de Noviembre de 2014 cuando insistíamos en que es fundamental que, al igual que sucede con la música, con la danza o con la práctica de deportes, se fomente una práctica formativa del pensamiento computacional desde las primeras etapas de desarrollo. (Zapata-Ros, 2014)

La conclusión de ello es que (Zapata-Ros, 2015) hay multitud de áreas del aprendizaje que conviene explorar e investigar en esta nueva frontera. Y en la planificación de los *currícula* tendrá que plantearse esta dicotomía: Enseñar a programar con dificultad progresiva (si se quiere incluso de forma lúdica o con juegos) o favorecer este nuevo tipo de pensamiento. Obviamente no hace falta decir que nuestra propuesta es la segunda, que además incluye a la primera.

Por otro lado (Zapata-Ros, 2018b): de igual forma que se habla de prelectura, pre-escritura o precálculo para nombrar competencias que allanan el camino a las destrezas clave y a las competencias instrumentales que anuncian, cabe hablar de precodificación para designar las competencias que son previas y necesarias en las fases anteriores del desarrollo para la codificación. Nos referimos por ejemplo a construcciones mentales que permiten alojar características de objetos de igual forma a como lo hacen las variables con los valores: Son en este caso el color, la forma, el tamaño, etc. O también operaciones con estos rasgos como son la seriación. Evidentemente hay muchas más habilidades y más complejas en su análisis y en el diseño de actividades y entornos para que este aprendizaje se produzca. Este ámbito de la instrucción es lo que podría denominarse precodificación, (...)

Igualmente, en trabajos anteriores (Pérez-Paredes y Zapata-Ros, 2018) poníamos énfasis del carácter evocador de otros aprendizajes y de otras metodologías como era la de rincones de Montessori. De esta forma se decía que tradicionalmente, en el diseño curricular de las primeras etapas de desarrollo se ha hablado de aprendizajes o de destrezas concretas que en un futuro predispondría a los aprendices para aprender mejor en un futuro habilidades matemáticas, geométricas, de lenguaje, como son la seriación, el encaje, la discriminación de objetos por propiedades, en las primeras etapas, y también en las de desarrollo del pensamiento abstracto o para la resolución de problemas. Así se ha hablado de la modularización, del análisis descendente, de análisis ascendente, de recursividad, e incluso de sinéctica y de cinestesia... En la perspectiva Montessori (1928, 1935 y 1937) por ejemplo esto es básico. Para ello se han desarrollado ya multitud de recursos, juegos y actividades que los educadores infantiles conocen bien.

En ese mismo capítulo hablamos de los programas y trabajos que se desarrollan en UK, EE. UU. como programas específicos dentro del currículum oficial que implementan pensamiento Computacional en Educación Infantil (*Key stage 1 in UK*) y que veremos en la parte de materiales de este mismo post, o en el post de esta serie dedicado a materiales.

4. Estado de la cuestión. Iniciativas ya existentes

Nos referimos a iniciativas ya existentes de pensamiento computacional, con actividades no estrictamente de programación, como competencias claves, ya incluidas en el currículum oficial de Educación Infantil o de etapas equivalentes, *Key stage 1 in UK*, etc. y al primer ciclo de primaria. Con ligeras variantes, son las edades y ciclos escolares que hay hasta los 8 años.

En el libro *El pensamiento computacional, análisis de una competencia clave* (Pérez-Paredes y Zapata-Ros, 2018), en la parte final, a partir de la página 89, hacíamos en una tabla un resumen de las situaciones del Pensamiento Computacional en los distintos sistemas educativos de los países en los que hemos encontrado que este está recogido en los currículos, en el sentido que se apunta en el libro, y que sostenemos en este trabajo: como

competencias transversales o competencias clave que sirven para favorecer el aprendizaje de la informática y de la programación, pero también para la resolución de problemas en otras materias aportando sus formas de pensamiento y métodos específicos, y también para problemas de la vida cotidiana.

Para el análisis también hemos tenido en cuenta las iniciativas de TACCLE 3: Coding Erasmus + Project (TACCLE 3 Consortium, 2017), reseñado por García-Peñalvo (2016) y por García-Peñalvo y otros autores (García-Peñalvo et al., 2016; García-Peñalvo, Reimann y Maday, 2018; García-Peñalvo y Mendes, 2018).

De entre todo el material analizado, entresacamos estrictamente las experiencias e incidencias que tienen que ver con el pensamiento computacional desenchufado en las primeras etapas. En la época en que se hizo el trabajo solo pudimos detectar, a partir de lo publicado en los *papers* e informes referenciados por Pérez-Paredes y Zapata-Ros (2018) que difunden experiencias, con investigaciones aparejadas que aseguraban un mínimo de rigor y consistencia en su desarrollo y conclusiones, dos casos propiamente dichos que cumplieran estos requisitos: incorporar el pensamiento computacional como competencia clave que fuese en estas etapas iniciales y que estuviese recogido como parte del currículo oficial de sus países, o sistemas educativos. Se trata de *CS Unplugged* en Nueva Zelanda, y del programa *PlayMaker* de Singapur. También incluimos, aunque propiamente no se puede considerar que cumple estos requisitos, el caso de una propuesta de currículum, ya introducido en su país, Macedonia, que hacen Jovanov et al. (2016) titulado “Trabajar con computadoras y conceptos básicos de programación” o simplemente “Computación”, para abreviar. Ofrecen una visión general del estado de la educación informática en Macedonia antes del currículo propuesto y luego ofrecen una visión general de la nueva materia introductoria para alumnos de ocho años. La incluimos aquí porque, aunque es híbrido de programación y de juegos, es otra iniciativa que prescinde de la computación y de los ordenadores para desarrollar el pensamiento computacional, aunque solo sea en parte.

En cada uno de los tres casos describimos primero qué se hace o qué hay, luego describo la situación en el contexto del currículo y del sistema educativo oficial, las características del caso en relación con las características definitorias que hemos propuesto (programación solo / desarrollo de competencias específicas como área transversal), y por último decimos las referencias de los documentos de donde hemos obtenido la información.

4.1. Nueva Zelanda

4.1.1. ¿Qué hay?

CS Unplugged es una colección de actividades de aprendizaje gratuitas que enseñan Ciencias de la Computación a través de interesantes juegos y acertijos, que usan tarjetas, cuerdas, lápices de colores y muchos juegos como los de Ikea o Montessori-Amazon, del tipo de los que explicamos en el artículo de referencia de este trabajo (Zapata-Ros, 2015) y en el libro *El pensamiento computacional, análisis de una competencia clave* (Pérez-Paredes y Zapata-Ros, 2018). Fue desarrollado para que los jóvenes estudiantes puedan interactuar con la informática, experimentar los tipos de preguntas y desafíos que experimentan los científicos informáticos, pero **sin tener que aprender primero la programación**.

Las actividades para las primeras etapas podemos verlas en la web.

Bell, Alexander, Freeman y Grimley (2009) son los investigadores responsables del proyecto *CS Unplugged* y en el documento *Computer science unplugged: School students doing real computing without computers* dan una visión general inicial del proyecto y también exploran por qué se ha popularizado y describen las diferentes formas en que se ha adaptado, que son:

- Vídeos de diferentes actividades.
- Hacer pulseras codificadas en binario.
- Competiciones.
- Adaptar las actividades de *CS Unplugged* a diferentes temas del currículo.
- Actividades al aire libre.
- Actividad en línea.

También analizan y justifican los principios de aprendizaje al diseñar las actividades y discuten sus planes futuros.

4.1.2. Situación en el currículo

El programa *CS Unplugged* es un programa completo de actividades desarrollado por *CS Education Research Group* en la Universidad de Canterbury, Nueva Zelanda. Está explicado por Bell et al. (2009) y por James Lockwood y Aidan Mooney.

Básicamente está orientado a Educación Secundaria e informa al Certificado Nacional de Secundaria que incluye Ciencias de la Computación entendidas en el sentido de PC.

Pero esto implica actividades incluidas en el currículo para etapas anteriores a partir de los cinco años.

A todas estas experiencias podemos acceder y verlas con detalle en los trabajos de Bell, Alexander, Freeman y Grimley (2009) Bell, Andreae y Robins (2014), Duncan y Bell (2015, November) y de Thompson y Bell (2013, November). Y en conjunto estos trabajos e investigaciones están reseñados además por James Lockwood Aidan Mooney (2017).

4.2. Singapur

4.2.1. ¿Qué hay?

Para abordar la creciente necesidad de nuevos programas de tecnología educativa (en este caso de Pensamiento Computacional a través fundamentalmente de robótica) en las aulas de la primera infancia, se lanzó el programa *PlayMaker* de Singapur. Es un programa en línea destinado a los maestros, para introducir a los niños más pequeños a la tecnología (Chambers, 2015; Digital News Asia, 2015). Según Steve Leonard, vicepresidente de la Autoridad de Desarrollo de Infocomm de Singapur (IDA), “a medida que Singapur se convierta en una nación inteligente, nuestros hijos necesitarán sentirse cómodos creando con tecnología” (IDA Singapur, 2015).

Aprovechando el creciente movimiento STEM, el objetivo del programa *PlayMaker* no es solo promover el conocimiento técnico sino también brindar a los niños herramientas para divertirse, practicar la resolución de problemas y generar confianza y creatividad (Chambers, 2015; Digital News Asia, 2015).

Como parte del programa *PlayMaker*, 160 centros preescolares en Singapur fueron dotados de una variedad de juguetes tecnológicos que involucran a los niños con la robótica, la programación, la construcción y la ingeniería, incluyendo: *BeeBot*, *Circuit Stickers* y la robótica KIBO (Chambers, 2015). Además del lanzamiento de nuevas herramientas, los educadores de la primera infancia también recibieron capacitación en un simposio de 1 día sobre cómo usar y enseñar con cada una de estas herramientas (Chambers, 2015).

Estas escuelas piloto también reciben apoyo técnico continuo y asistencia con la integración curricular como parte de este enfoque integral (IDA Singapur, 2015).

El estudio de referencia (Sullivan y Bers, 2017) se centra en evaluar los resultados de aprendizaje y compromiso de una de las herramientas de *Playmaker* implementadas: el *kit* de robótica KIBO. KIBO es un kit de construcción de robótica diseñado específicamente para niños de 4 a 7 años para aprender habilidades básicas de ingeniería y programación (Sullivan y Bers, 2015).

4.2.2. Situación en el currículo

El objetivo del programa piloto *PlayMaker* de Singapur es proporcionar ejemplos de éxitos y de áreas donde mejorar el trabajo futuro en implementación de PC en primeras etapas. Estos ejemplos se ofrecen como resultados válidos de este año en el que se ha llevado a cabo la experiencia piloto del programa *Playmaker* de Singapur que puede ser útil no solo para el trabajo futuro en este país, sino también en otros países que están desarrollando nuevos programas para la educación de la primera infancia.

La descripción completa está en el trabajo de Sullivan y Bers (2017).

Como hemos dicho, estas situaciones se producen tanto en Educación Infantil como en Primaria, además hay un caso que es interesante y que se da también en primaria, pero en el que se mezclan elementos de pensamiento computacional como programación y como juegos, nos referimos al caso de Macedonia.

4.3. Macedonia

4.3.1. ¿Qué hay?

Jovanov et al. (2016) presentan una descripción general de una aportación al currículo macedónico. Ofrecen una visión general del estado de la educación informática en Macedonia antes de esta propuesta, hacen un análisis y luego ofrecen una visión general de la nueva materia introductoria para alumnos de ocho años. En su comunicación dan una visión general del contenido que incluye siete unidades que se impartirán en dos clases por semana.

La investigación incluye las primeras impresiones de los capacitadores que realizaron la capacitación y la elaboración de las opiniones de los maestros. Comunican que en la iniciativa organizan los contenidos en siete unidades que se impartirán en dos clases por semana:

- Primeros pasos para usar la computadora.
- Gráficos por computadora.
- Procesamiento de texto.
- Vida en línea.
- Concepto de algoritmos y programas.
- Pensamiento computacional a través de un juego.
- Creación de programas simples.

Obviamente destacamos en el sentido propuesto de pensamiento computacional desenchufado las unidades quinta y sexta. En esta a los estudiantes se les enseña la noción de programación y aprenden a través de un juego, el DigitMile, que fue especialmente diseñado para ser utilizado en este plan de estudios junto con él.

4.3.2. Situación en el currículo

En el documento de referencia tenemos constancia y la descripción de la situación y los presupuestos sobre pensamiento computacional que llevaron a los responsables políticos en Macedonia a incluir la programación como parte de una nueva asignatura obligatoria para los alumnos a la edad de 8 años.

Como en los otros casos está reseñado en el documento de revisión de literatura de James Lockwood y Aidan Mooney (2017).

El documento con la referencia básica es el trabajo de Jovanov, Stankov, Mihova, Ristov y Gusev (2016, April).

5. Actividades

El diseño instruccional del pensamiento computacional desenchufado, como en cualquier otro caso, deberá procurar enlazar intenciones, condiciones y recursos con objetivos, con resultados deseados de aprendizaje. En este caso con el desarrollo de las habilidades que constituyen los elementos del pensamiento computacional tal como lo hemos definido en (Pérez-Paredes y Zapata-Ros, 2018; Zapata-Ros, 2015). En este esquema, el del diseño instruccional ocupan un lugar clave las actividades. Sin actividades no hay aprendizajes, y es haciendo como se aprende. Pero ¿qué actividades? Las que sin duda propicien el mayor acercamiento y el mayor y más eficiente adquisición de habilidades y constructos cognitivos de las componentes del pensamiento computacional. Pero además el pensamiento computacional de este tipo supone crear espacios, organizar recursos y dotarse de metodologías adecuadas. Consistentemente con lo dicho en otros sitios, y sin ánimo de ser exclusivos, dos van a ser las componentes metodológicas dominantes: la perspectiva Montessori de los rincones de trabajo para estas etapas y el *dominio del aprendizaje (mastery learning)*.

El pensamiento computacional está constituido por los elementos que vemos en la Figura 2. Pues bien, las actividades deben desarrollar estos elementos, y habrá que definirlos y diseñarlas con elementos curriculares adecuados (guías para maestros y profesores) y materiales para alumnos.

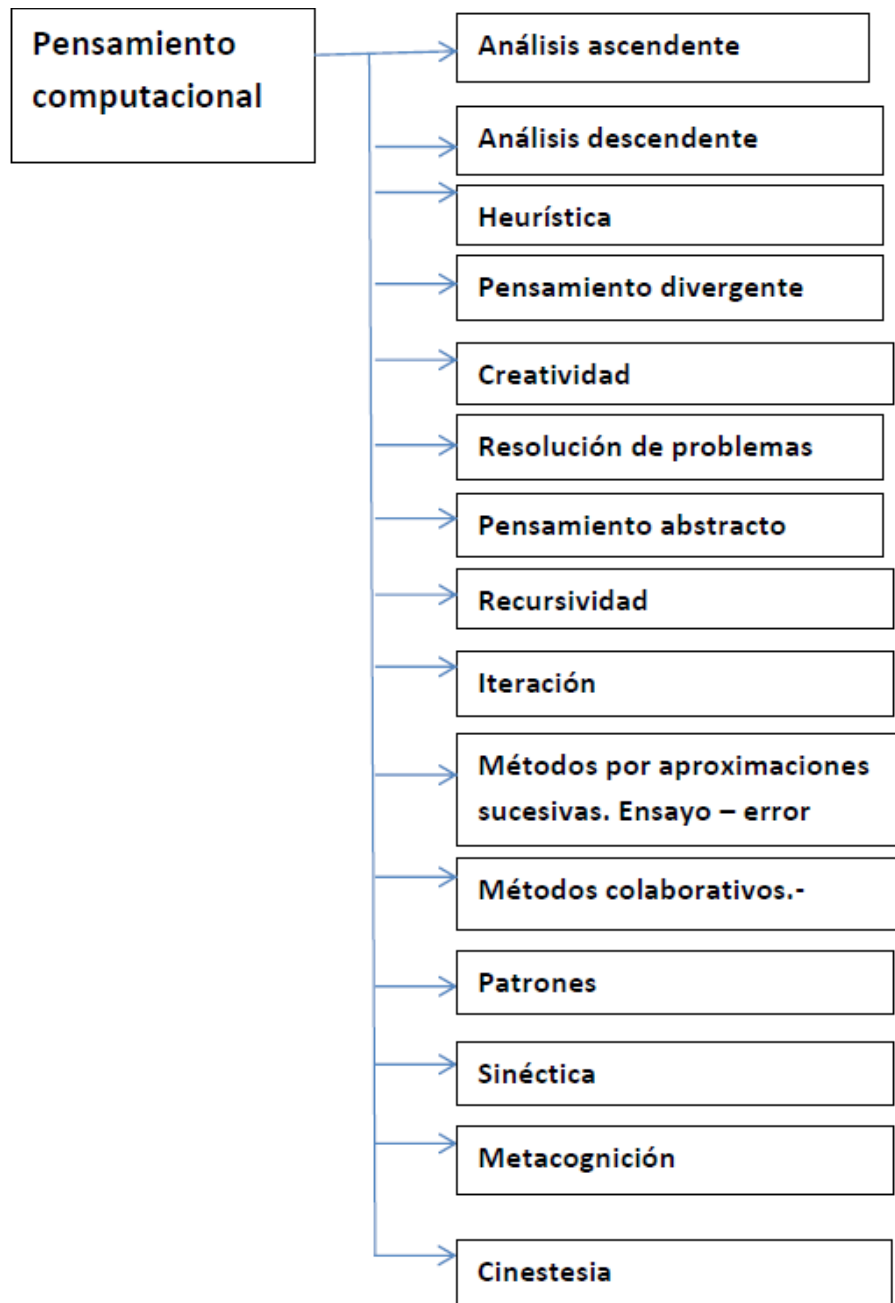


Figura 2. Elementos del Pensamiento Computacional

El trabajo que hay por delante es diseñar actividades adecuadas para cada uno o para cada racimo de ellos. Y hay que relacionarlas adecuadamente con las habilidades que desarrollan. También hay que decir cómo se verifican y en qué grado se consigue el dominio (evaluación).

El corolario es que hay que encontrar y explorar juegos y actividades con más potencial cognitivo para el desarrollo de esas habilidades. Y que hay que hacer un diseño educativo de esos juegos y de esas actividades.

Veamos un ejemplo.

Pre-álgebra para niños

Vamos a intentar introducir un juego o una actividad para desarrollo de pensamiento abstracto, pre-álgebra, en niños de entre 4 y 6 años. Conviene aclarar que el intervalo de edad lo hacemos de forma estimativa, porque igual podría el intervalo tomarse en función de otras características madurativas que se puedan tener en cuenta de manera convenientemente documentada y adaptada.

Youkara Youkara 1 PC es un juego infantil fabricado en China, cerca de Cantón, por la empresa Youkara, que se vende a través Amazon por el precio de 0,89€, cuya apariencia se muestra en la Figura 3.



Figura 3. Youkara

Es útil en principio, o está pensado, para que, con ayuda del maestro o de los padres, los niños desde los dos años se ejerciten en identificar los símbolos de los números o guarismos con la cantidad, o con el resultado de contar, abstrayendo esta cualidad de otras como es el color. Y para adquirir la práctica de las operaciones elementales a través de la práctica de contar.

Pero también podemos utilizar un juego tan versátil como este de otra forma, en el sentido señalado en el preámbulo de este artículo. Pensemos, para ello, en esta actividad: Presentar y pedir al niño que realice de forma consecutiva operaciones de multiplicar, con barras y con números indistintamente, hasta que alcance un completo dominio (ver Figura 4).



Figura 4. Propuesta de actividad con Youkara (1/5)

En ese punto podemos empezar a proponer prácticas mezclando barritas con números, donde al cabo de un rato si bien puede identificar la cantidad con el dígito, también puede identificar la cantidad o el dígito con un ente sustitutivo (ver Figura 5).



Figura 5. Propuesta de actividad con Youkara (2/5)

En este caso el elemento sustitutivo serán las barritas, y además en el mismo número. En fases alternativas podemos sustituir por una sola barrita o por un objeto, etc. y ver qué pasa (ver Figura 6).



Figura 6. Propuesta de actividad con Youkara (3/5)

Pidiendo al niño que diga a qué equivale o a qué ha sustituido el botón. Incluso poniendo botones en otras posiciones, por ejemplo, cómo se presenta en la Figura 7.



Figura 7. Propuesta de actividad con Youkara (4/5)

Repitiendo la operación hasta el dominio o hasta que el niño empiece a dar muestras de cansancio, pero rápidamente haciéndole ver el gran éxito que supone su logro.

Podemos incluso utilizar el mismo botón para otros casos y ver en ellos a qué número o cantidad sustituye (ver Figura 8).



Figura 8. Propuesta de actividad con Youkara (5/5)

Y, por último, utilizar en vez de un botón otro objeto.

Si finalmente conseguimos que adquiera el dominio en casos así habremos conseguido que adquiera un concepto muy próximo al de incógnita, ecuación y variable.

La cuestión ahora estriba en formar a maestros y dotarles de guías adecuadas, en destrezas docentes para que desarrollen en los niños un pensamiento preabstracto, que pueda ser evocado posteriormente.

Esta actividad enlaza pues con el elemento, de pensamiento computacional, que hemos considerado como **pensamiento abstracto**, como se refleja en la Figura 9.

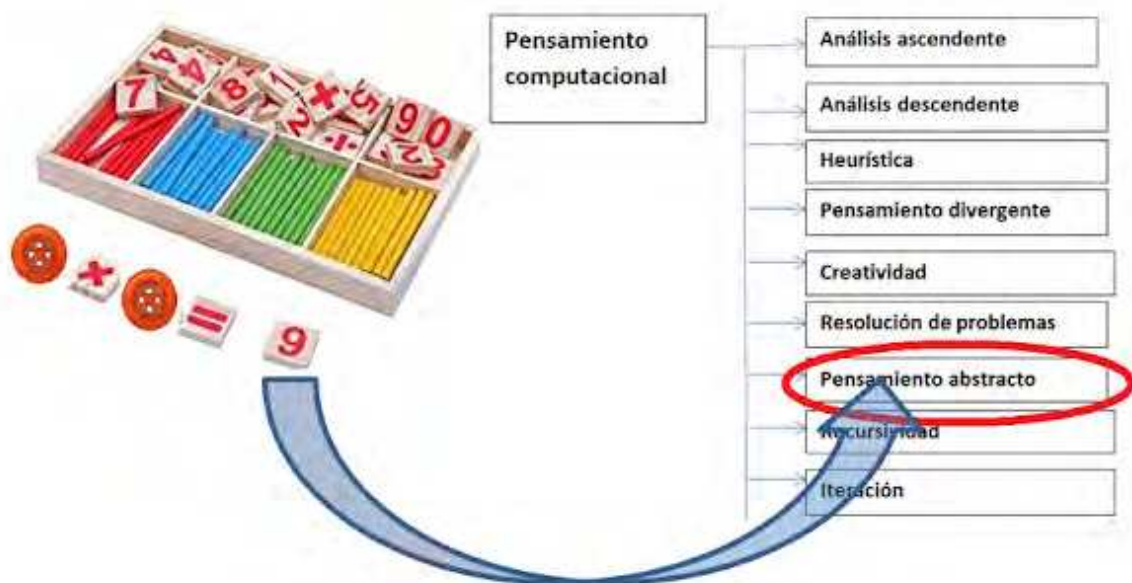


Figura 9. Relación de la actividad con el pensamiento abstracto

6. Materiales

Muchos hemos estado en Ikea y hemos visto juguetes basados en metodologías de aprendizaje por manipulación, los popularmente conocidos como juguetes Montessori. Tienen este nombre por ser esta autora la que más impulsó y desarrolló este tipo de aprendizaje, el que se produce por la manipulación autónoma por el alumno en un entorno, al que en este caso se denomina rincón, organizado para este fin. Son juguetes para que los niños, a través de la exploración y del desarrollo de sus actividades motoras y sensoriales también desarrollen otras habilidades y facultades cognitivas que en otro momento pueden facilitar aprendizajes de este tipo más complejos. Nos referimos, solo a modo de ejemplo, sin ser exhaustivos, a algunos de estos aprendizajes:

A sus habilidades de secuenciación: Por forma, tamaño, color, etc. con la consiguiente creación de ideas sobre conceptos como variable, o en sentido más amplio, de rasgos multivariantes de los objetos y de otros entes más o menos abstractos.

A sus habilidades de encaje, con discriminación de objetos por formas y tamaños, y del concepto más abstracto de encaje, qué tipos de cosas encajan con qué tipo de cosas. Pensemos en un futuro en variables y tipos de datos. En una variable booleana solo encajan datos booleanos, en una variable string solo encajan datos *string*, etc.

A su pensamiento lógico y a su capacidad de resolución de problemas. Puede parecer exagerado o traído por los pelos. Pero pensemos que con la percepción sensorial y las facultades cenestésicas se adquieren habilidades y conocimientos, se practica el ensayo-error: Como por ejemplo resolver problemas complejos como el de las Torres de Hanoi.

También, más allá de los diseños educativos y de las guías, en las iniciativas que hemos estudiado (Pérez-Paredes y Zapata-Ros, 2018; Zapata-Ros, 2018) podemos encontrar más juegos y actividades. Todas ellas tienen en común un mismo rasgo: Con muy poco artificio consiguen un conocimiento, un incremento cognitivo muy fuerte, muy considerable. Vamos a ver algunos ejemplos, pero en los lugares de *CS Unplugged* y de *Play Maker* se pueden encontrar muchos más que pueden servir de base o de referencia para diseñar adaptándolo de nuevo o recreándolo adaptado a nuestros entornos y a nuestras condiciones educativas con gran facilidad.

6.1. De tipo CS Unplugged (Informática desenchufada)

En <https://www.csunplugged.org/en/topics/> podemos encontrar 23 lecciones (actividades completas para desarrollar un tópico) para niños básicamente entre 5 y 10 años, propuestas por CS Unplugged (<https://www.csunplugged.org/es/>).

De ellas una actividad muy importante, aunque las demás también lo sean, es la que dedican a escritura, lectura e interpretación de números binarios. Ese es un tema que, con ser tan importante, se desconoce en la era digital, no solo por alumnos de cualquier nivel, sino por maestros.

En este sitio nos lo dan todo y nos lo explican con un coste cero, o casi, <https://www.csunplugged.org/en/topics/binary-numbers/unit-plan/how-binary-digits-work-junior/>

El material son cartulinas, que la página del programa nos da como imprimibles. También nos suministra un vídeo¹ de cómo se desarrolla una clase. Y donde es fácil ver cuál es el nivel de dominio.

No es este el lugar para explicar con más profundidad las actividades, guías, etc. Solo remito a que próximamente desarrollaremos nuestro propio material en español, y nuestras propias guías.

Hay otros temas, otras lecciones, por ver y tratar. Tanto las que ellos han previsto: Numeración binaria, *kitbots* y formas geométricas, redes de clasificación, detección y corrección de errores, y algoritmos de búsqueda. Así como otras que se pueden en un futuro elaborar: Álgebra, tipos de datos y variables, diagramas de flujo, operadores lógicos, etc.

6.2. Actividades con juguetes animados PlayMaker

Desde septiembre de 2015 la iniciativa Playmaker ha estado experimentando (Infocomm Media Development Authority, 2017 November) con las abejas *bee bot* en un preescolar experimental dirigido por Temasek Polytechnic. Esto fue tras un proceso de selección (curación) internacional (IMDA, 2017).

Con diversas modalidades, desde entonces hasta ahora, se han desarrollado múltiples opciones de dispositivo que **sin pantallas de ningún tipo** permiten hacer con juguetes reales lo que hacía LOGO con las órdenes elementales de la tortuga, y programarle para que camine o salga de un laberinto. Ésta es la idea y la tarea básica. El último ha sido SPRK Sphero del Lightning Lab app, en su versión para Apple que ha comercializado con el nombre de SPRK Sphero, y en el que está invirtiendo mucho, pero siempre desde el punto de vista de vincularlo a sus otros productos de Apple y a la venta de estos, más que a lo que hemos propuesto como pensamiento computacional, desenchufado o no y a la nueva alfabetización digital. De hecho, lo de no utilizar pantallas para niños es algo nominal puramente para ellos, de forma casi inmediata utilizan iPad, iPhone y Mac para programar y controlar el artilugio. Un trabajo al respecto sobre sus posibilidades lo hacen Ioannou y Bratitsis

1. <https://bit.ly/2JeqVF9>

(2017, July) en *Teaching the notion of Speed in Kindergarten using the Sphero SPRK robot*. Sin embargo, los resultados son pobres, sólo aprendizajes conceptuales sobre términos y acciones que serían igualmente posibles, e incluso más eficaces, de adquirir con un diseño mucho más simple y sin tanto aparataje tecnológico que puede complicarlo y hacerlo distractivo.

Un repertorio actual de actividades pueden encontrarlo en Sphero.edu, y de sus programas para el artilugio en <https://edu.sphero.com/>.

Pero volvamos a los juguetes de Play Maker. Del proceso gubernamental, diez juguetes fueron preseleccionados después de un proceso de investigación internacional, después fueron elegidos cuatro candidatos exitosos por un equipo de funcionarios expertos. Finalmente se implementaron Bee Bot y KIBO.

Ninguno de estos juguetes requiere una pantalla. Se trataba de que los niños no sobrepasasen ni se les indujese a pasar más de 2 horas al día usando una pantalla, incluido el uso en el hogar. Y por otra parte se deseaba fomentar la interacción social y desarrollar habilidades de comunicación.

En 2015 los juguetes fueron validados (Chambers, 2015) en el preescolar Yuhua PCF en el distrito de Jurong Lake. Fue elegido, porque es una escuela con tarifas bajas en lugar de una de las escuelas preescolares más caras de Singapur. El año 2016, se dotó un plan de \$1.5 millones para estos juguetes y su uso escolar en otros 160 centros preescolares en todo el país (Chambers, 2015).

Los procesos de validación, continuando con lo que dice la página oficial (Chambers, 2015 y Infocomm Media Development Authority, November 2017) han asegurado que los maestros pueden incorporar los juguetes en los planes curriculares oficiales, y reunieron una serie de actividades una mezcla de robots complejos y herramientas simples, procurando desarrollar una variedad de habilidades de pensamiento computacional desenchufado en los alumnos.

Todo esto está descrito en las news gubernamentales, en *New tech toys debut at pilot preschools as part of IDA's PlayMaker Programme*.

Sin embargo, no hemos encontrado registros de la investigación que validó los resultados, solo el trabajo de Sullivan y Bers (2017).

En este sentido hay que decir que, en Singapur, según la ordenación educativa los niños de 3 a 6 años asisten a centros preescolares, en su mayoría de gestión privada. Como en otros países existe una agencia pública autónoma, la Autoridad de Desarrollo de Medios de Infocomm (*The Infocomm Media Development Authority (IMDA)*), que fue la que lanzó la iniciativa *Playmaker* con el objetivo de introducir el Pensamiento Computacional en las Escuelas Infantiles y Preescolares en Singapur (IMDA, 2017). Hay más de 3000 centros preescolares en Singapur, la fase experimental fue en 160 de esos centros. La idea del IMDA para la introducción del pensamiento computacional era seleccionar juguetes sin pantalla, sin el uso auxiliar o central de *tablets*, ordenadores o *smartphones*, que involucrasen a los niños más pequeños en el juego y desarrollasen habilidades del pensamiento computacional, como son el pensamiento algorítmico.

El papel del IMDA fue proporcionar un *kit* de juguetes a los centros piloto para que los maestros los usen en el aula.

El paquete estaba compuesto por: 1) Beebot; 2) KIBO y 3) Pegatinas de circuito (*peel-and-stick electronics for crafting circuits*).

6.3. Actividades con Beebot en el programa PlayMaker

Beebot (Abeja robot) es un juguete que puede dar pasos sencillos, elementales, como lo hacía la tortuga de LOGO, pero sin tener que utilizar ordenador, pantalla y órdenes de programación. Pasos sencillos que son programables manualmente para controlar el movimiento, y para hacer secuencias de movimiento más complejas. De esta forma los niños pueden programar el juguete para que se place según un camino deseado, mediante la secuencia lógica de pasos, en su número y dirección adecuados, para llegar al destino.

Jugar a Beebot puede ayudar a los niños pequeños a desarrollar habilidades de resolución de problemas y pensamiento lógico al planificar y programar el movimiento del juguete.

En el documento de Jack Graham (2018), *reproducido y adaptado después por* Infocomm Media Development Authority de Singapur (2018 October) en *The game is on for PlayMaker*, se dan ideas y conceptos generales de las actividades que se proponen y se han llevado a cabo de forma experimental en PlayMaker. Como hemos dicho antes, la única referencia sobre la validación que da es, la única que hasta ahora hemos encontrado por otra parte es el trabajo de Sullivan y Bers (2017). *Dancing robots: integrating art, music, and robotics in Singapore's early childhood centers* en la revista *International Journal of Technology and Design Education*. Por tanto saltándome la línea argumental de este post quiero aprovechar la ocasión para remarcar la necesidad

de investigaciones que validen estas estrategias educativas utilizando este tipo de recursos, y la tendencia de pensamiento computacional desenchufado.

El trabajo de Graham (2018) es muy importante no solo porque hable de actividades y de Bee bot, también habla de KIBO y de Play Maker en general, sino porque da referencias sobre el proyecto muy importantes, señala cuales son las ideas generales, cuales son los efectos que supuestamente debe producir (outcomes) todo ello supone un cambio de paradigma, sostiene, y resuelve temas claves como son la animación a las chicas: “¿Por qué no empezar a involucrar a las niñas temprano, antes de que los estereotipos de género estén profundamente arraigados?”. Y sobre todo porque pone la evidencia de todo, en esta experiencia que después se generaliza, en el trabajo de Sullivan y Bers (2017):

La evidencia muestra que los niños en el piloto dominaron los conceptos de programación muy rápidamente. Al completar tareas que crean órdenes de instrucciones para los robots, los niños aprenden a hacer secuencias. Esta es una importante habilidad pre-matemática y de pre-alfabetización, dijo Amanda Sullivan, investigadora del equipo DevTech de la Universidad de Tufts, que creó el robot KIBO.

De todas formas, el más completo e interesante trabajo sobre propuestas de actividades con BeeBot lo hemos encontrado en la presentación Power Point de Gallagher, Thissen y Hrdina (2018) *Little Coders Computational Thinking in K-2 Classrooms* al congreso NCCE 2019. En ella dan una relación muy extensa, pero sólo nominal, de actividades en relación con los objetivos de aprendizaje de pensamiento computacional, prematemáticos y pre-STEM esperados.

6.4. Actividades con KIBO en el programa PlayMaker

KIBO fue desarrollado por investigadores en la Universidad de Tuft. Con él los niños pueden crear una secuencia de instrucciones al organizar bloques de madera KIBO. Los bloques se pueden incluir en la secuencia con las instrucciones, que se pasan al robot a través de un escaneo de códigos que llevan los bloques, y este ejecuta sucesivamente los pasos.

Podemos ver vídeo donde Nurul (2016) explica perfectamente el procedimiento.

Todo lo dicho anteriormente y lo tratado por Jack Graham (2018), reproducido y adaptado después por Info-comm Media Development Authority de Singapur (2018 October) en *The game is on for PlayMaker*, y la validación que dan, la única que hasta ahora hemos encontrado, en el trabajo Sullivan y Bers (2017) son válidos para KIBO.

El trabajo de Graham (2018) es muy importante porque da referencias sobre el proyecto (también sobre KIBO) muy importantes. Sobre todo, porque nos indica donde se evidencian los resultados de esta experiencia que después se generaliza.

Para empezar a hablar de las actividades con KIBO citaremos las guías que la propia empresa fabricante pone en circulación y regala con el kit de KIBO.

KinderLab ha lanzado una guía curricular para actividades de robótica autodirigida. Realmente va orientada a los aspectos más llamativos, pero no a los más eficaces, en el paso de la Universidad de Tufts ha perdido bastante de la potencia original. KinderLab Robotics es el fabricante del kit de robótica KIBO. Esta, llamada guía curricular, proporciona planes de lecciones autoinstructivas para las actividades de los estudiantes, que pueden realizarse bien en un equipo de diseñadores curriculares e en un centro, como actividades escolares.

Las dotaciones KIBO están diseñados para niños de 4 a 7 años, y proporcionan las componentes necesarias para “construir, programar, decidir el aspecto y dar vida a su propio robot” sin la necesidad de ordenador o dispositivo móvil. Hay cuatro tipos distintos de dotaciones disponibles; dos están disponibles en equipaciones para aulas de hasta 24 alumnos e incluyen guías y materiales para maestros.

La nueva guía curricular incluye tres programas de actividades distintas Cada una se centra en un aspecto diferente de la educación STEAM (*Science, Technology, Engineering, Mathematics and Arts*)².

- KIBO Bowling; se usa para las matemáticas.
- KIBO Dance Party; incluye arte, codificación y música.
- Diseño de formas con KIBO. Sirve para el dibujo y la geometría.

2. STEM—shorthand for science, technology, engineering, and mathematics—has quickly taken hold in education policy circles, but some experts in the arts community and beyond suggest it may be missing another initial to make the combination still more powerful. The idea? Move from STEM to STEAM, with an A for the arts (Robelen, 2011).

A las actividades se puede acceder desde la web oficial de Kinder Lab Robotics, impulsada por el grupo de la Universidad de Tufts (DEVTECH Research Group) en <http://kinderlabrobotics.com/>, y de las páginas específicas que de ahí sales. Por ejemplo, para niños de 4 a 7 años: <http://kinderlabrobotics.com/Kibo/>.

Como en el caso de Bee Bot, más información se puede obtener de la web Apolitical https://apolitical.co/solution_article/meet-the-robots-teaching-singapores-kids-tech/, que reproduce el trabajo de Graham (2018).

Actividades desarrolladas en DevTech Research Group, Tufts University

KIBO es el resultado aparentemente sencillo de décadas de investigaciones complejas sobre aprendizaje cinestésico, construccionismo y otras investigaciones precedentes sobre este tipo de aprendizajes, lideradas por la Dra. Marina Umaschi Bers, profesora en el Departamento de Desarrollo Infantil y Desarrollo Humano de Eliot-Pearson y director del Grupo de Investigación DevTech en la Universidad de Tufts. En las páginas del grupo están los trabajos sobre las investigaciones que respaldan KIBO, y las evidencias de que KIBO establece una gran diferencia entre el aprendizaje convencional y ciertos aprendizajes ayudados por este recurso, y además los niños lo hacen con gran motivación por su componente lúdico.

Para instituciones educativas

Los KIBO 18 y KIBO 21 están disponibles en varios Paquetes de Aula, que incluyen materiales de currículo (guías).

Además, se pueden comprar a la carta kits de robots individuales, bloques de programación, módulos, planes de estudio adicionales y materiales para maestros.

Para individuos

Además, KIBO está disponible en 4 configuraciones de robots diferentes. Otros artículos disponibles incluyen bloques y módulos de programación a la carta, currículo y materiales para maestros.

Un ejemplo de estas actividades es la que sigue (DevTech Research Group, 2016). Adjuntamos pues el enlace a la guía para *Literacy Activities with KIBO's Expression Module*, en la que se describen siete actividades para este tema. Este documento está disponible de forma gratuita en el sitio web de la Tufts University's Early Childhood Robotics Network.

Este folleto contiene actividades pensadas para que los niños practiquen la lectura y la escritura utilizando el Módulo de Expresión de KIBO. Las actividades se pueden realizar individualmente o se pueden integrar con un módulo de robótica. Puede elegir las actividades que más se adapten a cada profesor y a sus alumnos. No obstante, si bien están diseñadas para niños de Educación Infantil, pueden adaptarse fácilmente para alumnos de primer ciclo de Primaria, hasta 8 años.

Otro ejemplo de actividad con KIBO, que puede ser un modelo perfecto de diseño de unidad didáctica, con un diseño instruccional perfecto, es "*Where the Wild Things Are A KIBO Curriculum Unit on Programming and Robots Integrated with Foundational Literacy Topics*" (KinderLab Robotics, OCTOBER 28 2015) (DevTech Research Group of Tufts University, 2015) basado en la obra clásica infantil *Where the Wild Things Are*) de Maurice Sendak.

Esta unidad está Inspirada como hemos dicho en el libro *Where the Wild Things Are*, que suministra un enlace conceptual a los chicos. Pero los contenidos son sobre alfabetización digital y robótica. La modalidad de metodología es trabajo basado en un proyecto. Los estudiantes trabajan solos o en grupos (Metodología Montessori) para recrear el "wild rumpus" del libro original, programando sus robots KIBO para representar esta escena del libro.

Otras guías y recursos se pueden encontrar en el sitio de *Tufts University's Early Childhood Robotics Network*.

La página con los recursos y todo lo demás de Tufts University's Early Childhood Robotics Network está en <http://sites.tufts.edu/devtech/> y en The Developmental Technologies Research Group, dirigida por la Profesora Marina Umaschi Bers en el Eliot-Pearson Department of Child Study and Human Development, Tufts University (Grupo de Investigación sobre Tecnologías del Desarrollo, dirigido por la Prof. Marina Umaschi Bers en el Departamento de Estudio del Niño y Desarrollo Humano de Eliot-Pearson , en la Universidad de Tufts) <http://ase.tufts.edu/epcshd/>.

Hay una web con el repertorio completo de Kinder Lab Robotics³.

Algunos de estas *affordances* de pensamiento computacional desenchufado y sin pantallas también las podemos encontrar en Amazon o en otros sitios:

La primera de ella es el kit de Ratón, similar a beebot, en Amazon *Learning Resources Code & Go Robot Mouse Activity Set*

Hasta bien reciente ha estado igualmente disponible en Amazon la abeja programable Beet Bot.

6.5 Etiquetas adhesivas y circuitos (*peel-and-stick electronics for crafting circuits*).

Se trata de un conjunto de herramientas que consta de componentes electrónicos para quitar el plástico y pegar, tales como LED y cintas de cobre como conductor⁴. Con este kit de herramientas, los niños pequeños pueden crear proyectos interactivos de arte, o como artesanía, incrustados con adhesivos LED y sensores que responden al entorno o estímulos externos (ver la figura). Los niños pueden desarrollar su creatividad en actividades prácticas mientras aprenden y aplican conceptos básicos de electricidad, como son circuitos e interruptores. E incluso aprender circuitos lógicos. En el apartado de propuestas de actividades de este trabajo incluimos como ejemplo de actividad con este recurso la construcción de circuitos lógicos (puertas lógicas).

7. Bases para la propuesta de actividades

Con todo lo dicho en este trabajo cabría hacer una o varias propuestas de guías de actividades que sirviese de modelo. Incluso, en un estado más avanzado, cabría hacer una propuesta de lección (módulo, unidad didáctica, etc.) que incluyese, organizadas en un diseño instruccional completo, todas las actividades para esa unidad que contuviesen elementos de pensamiento computacional de este tipo, para educación infantil o de primer tramo de primaria.

No hace falta, para hacer la propuesta, que hagamos referencia y describamos todos los elementos de pedagogía específica o los principios de aprendizaje que utilizamos. Todo ello subyace y está presente en la propuesta.

Podemos tomar como ejemplos algunos que ya están consolidados y son de uso avalado por la edición y la práctica en entornos reales.

7.1. Material Montessori para actividades en las etapas de Ciclo Inicial primaria y Educación Infantil

Creo que no seríamos excesivamente osados si dijéramos que la aritmética es la ciencia de la computación utilizando números racionales positivos. Si hacemos una restricción utilizando la expresión aritmética como frecuentemente se hace, es decir como una parte de las competencias claves (las otras serían el álgebra elemental, la geometría la lectura, la escritura y la cinestesia), o sea específicamente los procesos de sumar, restar, multiplicar y dividir, estaríamos dentro de un conjunto de habilidades para las que es idónea la metodología Montessori. Con el añadido de que los materiales del aula infantil y primaria pensada por Montessori también presentan experiencias sensoriales para geometría y álgebra.

Los escritos, basados en experiencias e investigaciones, de Montessori hacen énfasis en que los niños pequeños se sienten atraídos de forma natural por la peculiaridad y las propiedades del número. Este instinto es el que hace que las matemáticas, como lenguaje, sean el producto exclusivo del intelecto humano. Es parte de la naturaleza de las personas. Las matemáticas surgen de la mente humana cuando entra en contacto con el mundo y contempla en el universo, en el mundo que le rodea, los factores de cantidad, cardinalidad, tiempo y el espacio.

Destacan la evidencia del esfuerzo del humano por comprender el mundo en el que vive, y el uso de los números para ello. Todos los humanos exhiben esta propensión matemática, incluso los niños pequeños. Por lo tanto, se puede decir que la humanidad tiene una *mente matemática*.

Hay pues un precedente a lo que consideramos el pensamiento computacional en las primeras etapas del desarrollo cognitivo, es lo que María Montessori llama *mente matemática*.

3. <http://resources.kinderlabrobotics.com/category/curriculum/>

4. Se pueden obtener en <https://bit.ly/2NjyrDZ>, <https://chibitronics.com/>, <https://bit.ly/2Y590Xn> y también en <https://amzn.to/2I-Z1Yx7>

Esta percepción la tuvieron Maria Montessori, y sus muchos colegas y colaboradores, al observar muchas evidencias espontáneas y no programadas en el contexto del desarrollo de los niños. Estos hechos se pudieron generalizar constituyendo principios que fueron la base de su metodología, al ser invariantes al lugar y al momento donde se producían. Estos principios constituyeron el ambiente preparado de su primera experiencia diseñada en la *Casa dei Bambini*.

Así pues, el trabajo de Montessori se centra en estas características universales del ser humano y de los niños. Para ilustrar este conjunto de características de la mente humana, Montessori rescata el término “mente matemática” de Blaise Pascal (1623-1662), quien dice que la esencia íntima del pensamiento humano (la mente humana) es de “de naturaleza matemática”. En *La Mente Absorbente (The absorbent mind)* Montessori (1959) escribe:

En nuestro trabajo, le hemos dado un nombre a esta parte de la mente que se construye con exactitud ... la llamamos “la mente matemática”. Tomo el término de Pascal ... quien dijo que la mente del hombre era matemática por naturaleza y que el conocimiento y el progreso proviene de la observación precisa.

Así pues, Montessori tomó el término del matemático, filósofo y teólogo francés del siglo XVII, Blaise Pascal (1623-1662). Otra coincidencia notable: los intereses de Pascal eran a la vez profundos que amplios, ¿o quizá es que la mente matemática era un ente más amplio, como ahora vemos al hablar del área Science, Technology, Engineering and Mathematics - STEM y de pensamiento computacional, y los integraba. Pascal inventó un artificio que hoy se considera un antecesor, el primero de los ordenadores, pero mecánico: la Pascalina; también produjo puntos de singularidad en la geometría, la teoría de probabilidades y la defensa del método científico como prueba de los asertos científicos (una idea nueva en su época); En esa época, en 1653, ya escribió lo siguiente en un ensayo titulado *Discours sur les Passions de l'Amour*:

There are two types of mind ... the mathematical, and what might be called the intuitive. The former arrives at its views slowly, but they are firm and rigid; the latter is endowed with greater flexibility and applies itself simultaneously to the diverse lovable parts of that which it loves.

En su trabajo *La Mente Absorbente (The Absorbent Mind)*, en la versión versión de 1949, encontramos que Montessori (1949, a través de Sackett, 2013, 2014).) utiliza el término “la mente matemática” en el capítulo “Further Elaboration through Culture”. Lo utiliza para describir una característica universal del ser humano, específicamente, que la mente “se desarrolla y funciona ... con exactitud”, a partir en este caso también a partir de las ideas de Pascal:

En nuestro trabajo, le hemos dado un nombre a esta parte de la mente que se construye con exactitud ... la llamamos “la mente matemática”. Tomo el término de Pascal ... quien dijo que la mente del hombre era matemática por naturaleza y que el conocimiento y el progreso proviene de la observación precisa.

Hay otros autores, además de Pascal y Montessori, que destacan esta característica humana universal. Traemos una que ofrece más detalles sobre este funcionamiento exacto a través de la observación, es en la creación de patrones, se debe al matemático, y especialista en Matemáticas de la cadena de radio NPR⁵, Keith Devlin (2001):

La mente humana es un reconocedor de patrones ... La capacidad de ver patrones y similitudes es una de las mayores fortalezas de la mente humana ... patrones visuales, patrones auditivos, patrones lingüísticos, patrones de actividades, patrones de comportamiento, patrones lógicos y muchos otros. Esos patrones pueden estar presentes en el mundo, o pueden ser impuestos por la mente humana como parte integral de su visión del mundo.

En el ANEXO 1 presentamos pues dos ejemplos de guías de actividades que hemos encontrado a partir de propuestas Montessori y que pueden servir de modelo.

5. “Math Guy” de NPR

7.2 Otros materiales Montessori que pueden ayudar a elaborar las guías de actividades

Un ejemplo de proyecto con una colección de libros que se dedica en su totalidad a la metodología Montessori y con un repertorio muy extenso de actividades es Libros Pedagógicos Montessori Paso a Paso de Escuela Viva. Son **libros que agrupan las presentaciones de materiales importantes desde los 2 a los 6 años**. Incluye traducidos al español varios libros escritos por María Montessori, y varios libros en los que se habla sobre la filosofía del método y en los que puedes profundizar. Lleva una buena presentación, secuenciada, ordenada, y con gráficos de calidad.

En particular recomendamos como útil por su estructura y formato utilizables para actividades de pensamiento computacional desenchufado los materiales que ofrecemos en el Anexo 2

8. Conclusiones

El diseño instruccional del pensamiento computacional desenchufado, como en cualquier otro caso, deberá procurar enlazar intenciones, condiciones y recursos con objetivos, con resultados deseados de aprendizaje. En este caso con el desarrollo de las habilidades que constituyen los elementos del pensamiento computacional tal como lo hemos definido en *El pensamiento computacional, análisis de una competencia clave* (Pérez-Paredes y Zapata-Ros, 2018) y en *Pensamiento computacional: Una nueva alfabetización digital* (Zapata-Ros, 2015). En este esquema, el del diseño instruccional ocupan un lugar clave las actividades. Sin actividades no hay aprendizajes, y es haciendo como se aprende. Pero ¿qué actividades? Las que sin duda propicien el mayor acercamiento y el mayor y más eficiente adquisición de habilidades y constructos cognitivos de las componentes del pensamiento computacional. Pero además el pensamiento computacional de este tipo supone crear espacios, organizar recursos y dotarse de metodologías adecuadas. Consistentemente con lo dicho en otros sitios, y sin ánimo de ser exclusivos, dos van a ser las componentes metodológicas dominantes: la perspectiva Montessori de los rincones de trabajo para estas etapas y el *dominio del aprendizaje (mastery learning)*.

En este libro (Pérez-Paredes y Zapata-Ros, 2018, pp. 63) y en este artículo (Zapata-Ros, 2015) decíamos que el pensamiento computacional estaba constituido por los elementos siguientes, allí los definíamos y los describíamos. Pues bien, las actividades deben desarrollar estos elementos, y habrá que definirlos y diseñarlas con elementos curriculares adecuados (guías para maestros y profesores) y materiales para alumnos.

Los elementos recordemos que se muestran en la Figura 2.

El trabajo que hay por delante es diseñar actividades adecuadas para cada uno o para cada racimo de ellos. Y hay que relacionarlas adecuadamente con las habilidades que desarrollan. También hay que decir cómo se verifican y en qué grado se consigue el dominio (evaluación).

El corolario es que hay que encontrar y explorar juegos y actividades con más potencial cognitivo para el desarrollo de esas habilidades. Y que hay que hacer un diseño educativo de esos juegos y de esas actividades.

9. Referencias

- Balanskat, A., & Engelhardt, K. (2015). *Computing our future. Computer programming and coding Priorities, school curricula and initiatives across Europe*. Brussels, Belgium: European Schoolnet. Retrieved from <https://goo.gl/i5aQiv>
- Bawden, D. (2001). Information and digital literacies: a review of concepts. *Journal of Documentation*, 57(2), 218–259. doi:<https://doi.org/10.1108/EUM0000000007083>
- Bawden, D. (2008). Origins and concepts of digital literacy. *Digital literacies: Concepts, policies and practices*, 17-32.
- Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *The New Zealand Journal of Applied Computing and Information Technology*, 13(1), 20-29.
- Bell, T., Andreae, P., & Robins, A. (2014). A case study of the introduction of computer science in NZ schools. *ACM Transactions on Computing Education (TOCE)*, 14(2), 10. doi:<https://doi.org/10.1145/2602485>
- Bell, T., & Vahrenhold, J. (2018). CS Unplugged—How Is It Used, and Does It Work? In H. J. Böckenhauer, D. Komm, & U. W. (Eds.), *Adventures Between Lower Bounds and Higher Altitudes*. Cham: Springer. doi:https://doi.org/10.1007/978-3-319-98355-4_29

- Chambers, J. (2015). Inside Singapore's plans for robots in pre-schools. How a bold new scheme is teaching tech skills to 6 year olds. *GovInsider*. <https://bit.ly/2LagRzi>
- Devlin, K. (2001) *The Math Gene: How Mathematical Thinking Evolved and Why Numbers Are like Gossip*. NY: Basic Books.
- DevTech Research Group of Tufts University (2015). Where the Wild Things Are A KIBO Curriculum Unit on Programming and Robots Integrated with Foundational Literacy Topics. <https://bit.ly/2FrC4kN>
- DevTech Research Group (2016). *Literacy Activities with KIBO's Expression Module*, <https://bit.ly/2x8zOKq>
<https://bit.ly/2ZICx9T>
- DevTech Research Group (October 2018). *Where the Wild Things Are*. <https://bit.ly/2L5zvIN> <https://bit.ly/2FrC4kN>
- Digital News Asia, (2015) <https://bit.ly/2J40a5W>
- Duncan, C., & Bell, T. (2015). A pilot computer science and programming course for primary school students. In *Proceedings of the Workshop in Primary and Secondary Computing Education* (pp. 39-48). USA: ACM. doi:<https://doi.org/10.1145/2818314.2818328>
- Gallagher, A., Thissen, S. & Hrdina, V. (2018). Little Coders Computational Thinking in K-2 Classrooms - *NCCE 2019*. <https://bit.ly/2N6AcUH>
- García-Peñalvo, F. J. (2016). A brief introduction to TACCLE 3 – Coding European Project. In F. J. García-Peñalvo & J. A. Mendes (Eds.), *2016 International Symposium on Computers in Education (SIIE 16)*. USA: IEEE. doi:<https://doi.org/10.1109/SIIE.2016.7751876>
- García-Peñalvo, F. J., Llorens Largo, F., Molero Prieto, X., & Vendrell Vidal, E. (2017). Educación en Informática sub 18 (EI<18). *ReVisión*, 10(2), 13-18.
- García-Peñalvo, F. J., & Mendes, J. A. (2018). Exploring the computational thinking effects in pre-university education. *Computers in Human Behavior*, 80, 407-411. doi:<https://doi.org/10.1016/j.chb.2017.12.005>
- García-Peñalvo, F. J., Reimann, D., & Maday, C. (2018). Introducing Coding and Computational Thinking in the Schools: The TACCLE 3 – Coding Project Experience. In M. S. Khine (Ed.), *Computational Thinking in the STEM Disciplines. Foundations and Research Highlights* (pp. 213-226). Cham, Switzerland: Springer. doi:https://doi.org/10.1007/978-3-319-93566-9_11
- García-Peñalvo, F. J., Reimann, D., Tuul, M., Rees, A., & Jormanainen, I. (2016). *An overview of the most relevant literature on coding and computational thinking with emphasis on the relevant issues for teachers*. Belgium: TACCLE3 Consortium. doi:<https://doi.org/10.5281/zenodo.165123>
- García-Peñalvo, F. J., Rees, A. M., Hughes, J., Jormanainen, I., Toivonen, T., & Vermeersch, J. (2016). A survey of resources for introducing coding into schools. In F. J. García-Peñalvo (Ed.), *Proceedings of the Fourth International Conference on Technological Ecosystems for Enhancing Multiculturality (TEEM'16) (Salamanca, Spain, November 2-4, 2016)* (pp. 19-26). New York, NY, USA: ACM. doi:<https://doi.org/10.1145/3012430.3012491>
- González-González, C. S. (2019). State of the art in the teaching of computational thinking and programming in childhood education. *Education in the Knowledge Society*, 20, 17. doi:10.14201/eks2019_20_a17
- Graham, J. (2018 July). Meet the robots teaching Singapore's kids tech. The interactive toys reduce time children spend in front of screens. *Apolitical*. <https://bit.ly/2J2gjJO>
- IDA Singapore. (2015). *IDA supports preschool centres with technology-enabled toys to build creativity and confidence in learning*. <https://bit.ly/2FnvtrC>
- IMDA. (2017). *PlayMaker Changing the Game*. <https://bit.ly/2qxDXW1>
- Infocomm Media Development Authority (2017 November) *PlayMaker Changing the Game. IMPACT INFOCOMM MEDIA TRENDS, INSIGHTS AND ANALYSIS*. <https://bit.ly/2qxDXW1>
- Infocomm Media Development Authority (2018 October) *The game is on for PlayMaker. IMPACT INFOCOMM MEDIA TRENDS, INSIGHTS AND ANALYSIS*. <https://bit.ly/2Kw2zK0>
- Ioannou, M., & Bratitsis, T. (2017, July). Teaching the notion of Speed in Kindergarten using the Sphero SPRK robot. In *Advanced Learning Technologies (ICALT), 2017 IEEE 17th International Conference on* (pp. 311-312). USA: IEEE. doi:<https://doi.org/10.1109/ICALT.2017.70>
- Jovanov, M., Stankov, E., Mihova, M., Ristov, S., & Gusev, M. (2016, April). Computing as a new compulsory subject in the Macedonian primary schools curriculum. In *Global Engineering Education Conference (EDUCON), 2016 IEEE* (pp. 680-685). USA: IEEE. doi:<https://doi.org/10.1109/EDUCON.2016.7474623>
- KinderLab Robotics (OCTOBER 28, 2015). *KIBO resources. Curriculum units. Where the Wild Things Are*. <https://bit.ly/2L5zvIN>
- Lillard, A. S. (2011). *What Belongs in a Montessori Primary Classroom? Montessori Life*, 23(3), 18.

- Llorens-Largo, F., García-Peñalvo, F. J., Molero Prieto, X., & Vendrell Vidal, E. (2017). La enseñanza de la informática, la programación y el pensamiento computacional en los estudios preuniversitarios. *Education in the Knowledge Society*, 18(2), 7-17. doi:<https://doi.org/10.14201/eks2017182717>
- Lockwood, J., & Mooney, A. (2017). Computational Thinking in Education: Where does it Fit? A systematic literary review. *arXiv preprint arXiv:1703.07659*.
- Pérez-Paredes, P., & Zapata-Ros, M. (2018). *El pensamiento computacional, análisis de una competencia clave*. Scotts Valley, CA, USA: Createspace Independent Publishing Platform. doi:<https://doi.org/10.21585/ijcses.v2i1.26>
- Merrill, M. D. (2002). First principles of instruction. *Educational technology research and development*, 50(3), 43-59. doi:<https://doi.org/10.1007/BF02505024>
- Merrill, M. D. (2007). First principles of instruction: A synthesis. In R. A. Reiser & J. V. Dempsey (Eds.), *Trends and issues in instructional design and technology* (2nd ed., pp. 62-71). Upper Saddle River, NJ: Merrill/Prenice-Hall.
- Merrill, M. D. (2009). First principles of instruction. In C. M. Reigeluth & A. A. Carr-Chellman (Eds.), *Instructional-design theories and models: Building a common knowledge base* (Vol. III, pp. 41-56). New York: Routledge.
- Montessori, M. (1928). *Antropología Pedagógica*. Barcelona: Araluce
- Montessori, M. (1937). *Método de la Pedagogía Científica*. Barcelona: Araluce
- Montessori, M. (1935). *Manual práctico del método*. Barcelona: Araluce
- Montessori, M. (1967). *The Absorbent Mind*. 1949. Trans. Claude A. Claremont. Holt, Rinehart, and Winston.
- Montessori, M. (1991). *The Advanced Montessori Method*, Vol. 1. 1917. Trans. Florence Simmonds and Lily Hutchinson. Oxford: Clio.
- Montessori, M. (1934). *Psychogeometry*. Trans. Benedetto Scoppola. Ed. Kay Baker. Laren, The Netherlands: Montessori-Pierson Publishing Company, 2011. Retrans. of Psychogeometry Spanish ed.
- Montessori, M. (1989). *The Secret of Childhood*. Trans. Barbara Barclay Carter. Hyderabad: Orient Longman: 1963. Montessori, Maria. What You Should Know About Your Child. Oxford: Clio.
- Nurul, A. (2016). Playmaker Project. The children were introduced to the robots named Bee-Bot and KIBO for this Playmaker Project. <https://vimeo.com/179032348>
- Pérez-Paredes, P. & Zapata-Ros, M. (2018). *El pensamiento computacional, análisis de una competencia clave*. Scotts Valley, CA, USA: Createspace Independent Publishing Platform. <https://amzn.to/2KumN6N>
- Reigeluth, C. M. (1999). What is instructional-design theory and how is it changing? In C. M. Reigeluth (Ed.), *Instructional-design theories and models: A new paradigm of instructional theory* (Vol. II, pp. 5-29). Mahwah, NJ: Lawrence Erlbaum Associates.
- Reigeluth, C. M. (2016). Teoría instruccional y tecnología para el nuevo paradigma de la educación. *RED. Revista de Educación a Distancia*. 50. doi:<https://doi.org/10.6018/red/50/1a>
- Robelen, E. W. (2011). STEAM: Experts make case for adding arts to STEM. *Education week*, 31(13), 8.
- Sackett, G. (2013). *The Mathematical Mind*. <https://bit.ly/2ICjYyB>
- Sackett, G. (2014). "The Lines that Make the Clouds" The Essence of the Mathematical Mind in the First Six Years of Life. *NAMTA Journal*, 39(2).
- Sullivan, A., & Bers, M. U. (2015). Robotics in the early childhood classroom: Learning outcomes from an 8-week robotics curriculum in pre-kindergarten through second grade. *International Journal of Technology and Design Education*, 26(1), 3-20. doi:<https://doi.org/10.1007/s10798-015-9304-5>
- Sullivan, A., & Bers, M. U. (2017). Dancing robots: integrating art, music, and robotics in Singapore's early childhood centers. *International Journal of Technology and Design Education*, 1-22
- TACCLE 3 Consortium. (2017). *TACCLE 3: Coding Erasmus + Project website*. <https://goo.gl/f4QZUA>
- Thompson, D., & Bell, T. (2013, November). Adoption of new computer science high school standards by New Zealand teachers. In *Proceedings of the 8th Workshop in Primary and Secondary Computing Education* (pp. 87-90). USA: ACM. doi:<https://doi.org/10.1145/2532748.2532759>
- Velázquez-Iturbide, J. Á. (2018). Report of the Spanish Computing Scientific Society on Computing Education in Pre-University Stages. In F. J. García-Peñalvo (Ed.), *Proceedings TEEM'18. Sixth International Conference on Technological Ecosystems for Enhancing Multiculturality* (Salamanca, Spain, October 24th-26th, 2018) (pp. 2-7). New York, NY, USA: ACM. doi:<https://doi.org/10.1145/3284179.3284180>
- Velázquez Iturbide, J. Á., Bahamonde, A., Dabic, S., Escalona, M. J., Feito, F., Fernández Cabaleiro, S., ... Zapata Ros, M. (2018). *Informe del Grupo de Trabajo SCIE/CODDII sobre la enseñanza preuniversitaria de la informática. España: Sociedad Científica Informática de España, Conferencia de Decanos y Directores de Ingeniería Informática*. <https://goo.gl/dmCPgm>
- Villalba-Condori, K. O., García-Peñalvo, F. J., Lavonen, J., & Zapata-Ros, M. (2019). What Kinds of Innovations Do We Need in Education? In K. O. Villalba-Condori, F. J. García-Peñalvo, J. Lavonen, & M. Zapata-Ros (Eds.),

- Proceedings of the II Congreso Internacional de Tendencias e Innovación Educativa – CITIE 2018 (Arequipa, Perú, November 26-30, 2018)* (pp. 9-15). Aachen, Germany: CEUR-WS.org.
- Zapata-Ros, M. (2014). Coding y pre-coding. Blog Microposts, Tumblr <https://bit.ly/31Lwt2a>
- Zapata-Ros, M. (Noviembre 2014). ¿Por qué “pensamiento computacional”? (I) *Blog Pensamiento computacional y alfabetización digital / Computational thinking and computer literacy*. <https://bit.ly/2x5ENf8>.
- Zapata-Ros, M. (Diciembre 2014). Pensamiento computacional y alfabetización digital (I). Blog RED, Hypotheses. <https://bit.ly/2Ruv0bZ>
- Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. *RED. Revista de Educación a Distancia* (46), 1-47. doi:<https://doi.org/10.6018/red/46/4>
- Zapata-Ros, M. (2018a). El pensamiento computacional en la transición entre culturas epistemológicas. *Blog RED El aprendizaje en la Sociedad del Conocimiento*. <https://bit.ly/31NNc50>
- Zapata-Ros, M. (2018b). Pensamiento computacional. Una tercera competencia clave. (I) *Blog RED El aprendizaje en la Sociedad del Conocimiento*. <https://bit.ly/2L85S9x>

ANEXO 1

Primer ejemplo.- Montessori à la maison, de Delphine Gilles Cotte

Un ejemplo de diseño de actividades de este tipo nos lo da el libro *Montessori à la maison*, para padres, de Delphine Gilles Cotte, que en España se publica como *Montessori en casa* (Tu hijo y tú) y en él la actividad “la torre rosa”



Figura 11. Ejemplo de actividad Montessori: La torre rosa.



Figura 12. Habilidad que se desarrolla con “la casa rosa”: La clasificación.

El guion es sencillo y la presentación sugestiva. No se necesita más.

Título: La torre rosa

Breve descripción y justificación: “El niño trabaja la lógica e inicia su capacidad de juicio...”

Elementos materiales que se necesitan. Descripción: “Necesitarás: 10 cubos de madera rosa...”

Enunciado de las actividades: “Ejercicio 1.- Pídele al niño que vaya a buscar los diez cubos de la torre...”

Descripción más extensa y comentada de las actividades: “En tiendas puedes encontrar torres...”

Segundo ejemplo.- Actividades Montessori de Matemáticas.

Living Montessori Now. Maths Activities Primary Guide es un repositorio de actividades y recursos para matemáticas que nos parece muy interesante como modelo de repositorio de recursos y actividades para pensamiento computacional desenchufado.

Montessori Math Activities

LivingMontessoriNow.com



Fig. 13 Materiales que se utilizan en las actividades de *Living Montessori Now. Maths Activities Primary Guide*

La web *Living Montessori Now* de Deb Chitwood⁶ dedica una página a Matemáticas. En ella podemos ver una numerosa y variada colección de actividades.

De ellas elegimos, sólo a título de ejemplo significativo, la denominada *Small Bead Frame: introduction to addition, subtraction and multiplication* (**Ábaco: Introducción a la suma, la resta y la multiplicación**) en el apartado *Passage to Abstraction* (Transición a la abstracción).

Se trata de un ejemplo que se propone recurrentemente como actividad, conocida por todos, por eso la elegimos. Pero también en ese mismo apartado y en el resto hay numerosos y muy interesantes casos y ejemplos de actividades todos con un esquema y una estructura similar. También señalamos que, aunque utiliza una expresión en inglés muy elaborada, *Small Bead Frame* que es la expresión estándar utilizada en los medios Montessori (Lillard, 2011), se está refiriendo al ábaco decimal, diez cuentas por barra. El que podemos encontrar en Ikea o en Amazon, como tantos otros juegos que en este trabajo citamos.

Pues bien la estructura de estas guías didácticas es la siguiente, tomando como referencia *Small Bead Frame: introduction to addition, subtraction and multiplication*:

Título:

Materiales (Elementos materiales que se necesitan. Descripción): En este caso el ábaco (Small Bead Frame), las pegatinas con las unidades, decenas, etc y materiales de anotación, papel y lápiz.

Notas.- Este apartado hace referencia a notas metodológicas. Observaciones que se hacen al maestro sobre su trabajo con los alumnos.

Actividades.- Descripción de los ejercicios o actividades, en este caso agrupadas en racimos de actividades (a las que llama “presentaciones”). La primera incluye introducción, contando sin cero, contando con cero,...; la segunda incluye adición estática, adición, ...; y así sucesivamente.

Más notas metodológicas.- Parece ser que estas son con carácter más generales que las anteriores, en este caso da una nota que vagamente recuerda un criterio de dominio (mastery learning):

Dr. Montessori referred to this piece of material as marking the passage to abstraction.

This material allows the child to stop using the material when he no longer needs it to find the answer to the problem.

Objetivos.- Escritos de forma directa y sencilla, como propósitos. En este caso para mostrar la relación entre las categorías posicionales del sistema decimal y para aclarar el sentido de posición y valor de posición, como requisitos necesarios y de ayuda para sumar y restar.

Control de error.- Se refiere a evaluación formativa, en este caso la propia habilidad del niño y las inscripciones en las anotaciones del niño.

Este apartado es importante y junto con los elementos o criterios de *mastery learning* los incluiremos en nuestra propuesta.

Edad.- Periodo madurativo para el que de forma estándar está recomendado. En este caso desde 5 años y medio a 6 años

Preguntas y comentarios.- Apartado dedicado a los maestro para que compartan y discutan. En este caso se da la indicación de que compartan sus tus experiencias en un foro.

6. Formada en St. Nicholas Training Centre in London, ahora Montessori Centre International, y miembro y ha trabajado en Association Montessori Internationale (AMI), American Montessori Society (AMS) y National Center for Montessori Education (NCME).

ANEXO 2

Montessori Paso a Paso. El cálculo y las matemáticas. 3-6 años

Se puede adquirir en Amazon

Va dedicado al aprendizaje de las matemáticas con el método Montessori. **Tiene agrupadas y secuenciadas numerosas actividades y presentaciones para edades entre los 3 a los 6 años.**

Es significativo el índice detallado con todas las presentaciones y el índice por aprendizajes (Fig. 14)

Índice por aprendizajes	
Los números	
Los listones rojos y azules	24
Las cifras rugosas	27
Agrupación de las barras numéricas con los símbolos	29
Los puños	32
Las fichas	35
El juego de la memoria	37
El sistema decimal	
El banco de las perlas doradas y los símbolos	38
Las primeras tablas de Seguin	75
Las segundas tablas de Seguin	81
Las cadenas	84
La suma	
El descubrimiento de la suma estática con las perlas doradas	53
La suma dinámica con las perlas doradas	59
Los sellos	92
La suma con los sellos	95
La tabla de puntos	100
La suma estática con el ábaco	233
La suma dinámica con el ábaco	238
La memorización de la suma	
La serpiente positiva, principio de la memorización de la suma	133
La memorización de la suma con la tabla de regletas	150
La memorización de la suma con las tablas de los dedos	157
La memorización de la suma: las otras herramientas	167
La resta	
La resta estática con las perlas doradas	61
La resta dinámica con las perlas doradas	63
La resta estática con los sellos	107
La resta dinámica con los sellos	110
La resta estática con el ábaco	242
La resta dinámica con el ábaco	245
La memorización de la resta	
La serpiente negativa, principio de la memorización de la resta	172
La memorización de la resta con la tabla de regletas	199
La memorización de la resta con las tablas de los dedos	205
La multiplicación	
La multiplicación con las perlas doradas	65
La multiplicación por 10, 100, 1000 con las perlas doradas	113
La multiplicación con la disposición sistemática de las perlas de color	115
La multiplicación con los sellos	118
La memorización de la multiplicación	
La memorización de la multiplicación con la tabla perforada	210
La memorización de la multiplicación con las tablas de los dedos	214
La división	
La división con las perlas doradas	69
La división con los sellos (divisor de 1 cifra)	119
La división con un divisor de 2 cifras con las perlas doradas	69
La división con los sellos (divisor de 2 cifras)	124
La división con un divisor de 3 cifras con las perlas doradas	74
La división con los sellos (divisor de 3 cifras)	131
La memorización de la división	
La memorización de la división con la tabla perforada	220
La memorización de la división con las tablas de los dedos	226
Las fracciones	
Los círculos fraccionados	254
Las primeras operaciones con fracciones	261

Fig. 14 Índice de aprendizajes con la unidad "Montessori Paso a Paso. El cálculo y las matemáticas."

Un ejemplo de actividad es la de *La división*. Y de material es el del *Gabinete geométrico*⁷.

Where the Wild Things Are.

Una excelente guía para el diseño de material curricular y el diseño de actividades lo constituye el Material de para KIBO titulado *Where the Wild Things Are. A KIBO Curriculum Unit on Programming and Robots Integrated with Foundational Literacy Topics* de DevTech Research Group (2018), de Tufts University.

Un ejemplo para nuestra propuesta sería la actividad siguiente, incluida en el trabajo:

7. *Gabinete geométrico* <http://ecole-vivante.com/gabinete-geometrico.pdf>

Activity 3: Vowel Maker

Goal: Students will program their robot to travel around and create new words.

Materials: 1 KIBO set per *group* of students, one Expression Module per KIBO, index cards, pen/ marker

KIBO Concept: Sequencing with KIBOs programming blocks

Activity Preparation: Review KIBO's different blocks. Identify three letter words that students are familiar with and have a vowel as one of the letters (ex. cat, bat, jet, bus, dog, top, hen, bib, lip). Choose a handful of words (at least as many words as there are robots) and write one word on each index card; however, do not write the vowel. Instead, draw a line to indicate that the vowel belongs in that area.

Activity Description: Students will choose one vowel (either of their choosing or one that is assigned) to write on their Expression Module. Then, they will program their robot to travel from a designated spot to one of the index cards. In order to go to an index card, the vowel on the Expression Module needs to be the vowel that completes the index card to create a real word. For example, if a group has "e" written on their index card, their KIBO could travel to "h_n" and "j_t" but not "c_t." If desired, this activity can be repeated, either by having the groups change their vowel or have the robots travel to another index card.

Activity Extension: Try lengthening the words or choosing words that have the same vowel in two different places in a word.

Actividad 3: Hacedor de vocales

Objetivo: Los estudiantes programarán su robot para viajar y crear nuevas palabras.

Materiales: 1 conjunto de KIBO por grupo de estudiantes, un módulo de expresión por KIBO, fichas, bolígrafo / marcador

Concepto KIBO: Secuenciación con bloques de programación KIBOs

Preparación de la actividad: revisar los diferentes bloques de KIBO. Identifique las palabras de tres letras con las que los estudiantes están familiarizados y tenga una vocal como una de las letras (por ejemplo, gato, murciélago, avión, autobús, perro, parte superior, gallina, babero, labio). Elija un puñado de palabras (al menos tantas palabras como robots) y escriba una palabra en cada tarjeta de índice; Sin embargo, no escriba la vocal. En su lugar, dibuje una línea para indicar que la vocal pertenece a esa área.

Descripción de la actividad: Los estudiantes elegirán una vocal (ya sea de su elección o asignada) para escribir en su Módulo de Expresión. Luego, programarán su robot para viajar desde un lugar designado a una de las tarjetas de índice. Para ir a una tarjeta de índice, la vocal en el Módulo de Expresión debe ser la vocal que completa la tarjeta de índice para crear una palabra real. Por ejemplo, si un grupo tiene una "e" escrita en su tarjeta de índice, su KIBO podría viajar a "h_n" y "j_t" pero no a "c_t". Si lo desea, esta actividad se puede repetir, ya sea haciendo que los grupos cambien su Vocal o haz que los robots viajen a otra ficha.

Extensión de actividad: intente alargar las palabras o elegir palabras que tengan la misma vocal en dos lugares diferentes de una palabra.

En este caso introduce además de los apartados de título, descripción. Materiales, etc. un apartado con el objetivo efectivo de la actividad (Goal): "Los estudiantes programarán su robot para viajar y crear nuevas palabras." Que puede servir como en casos anteriores como criterio de dominio.