

I CONGRESO IBEROAMERICANO DE DOCENTES

CONGRESO VIRTUAL DEL 26 NOVIEMBRE AL 08 DICIEMBRE DE 2018

ALGECIRAS (CÁDIZ) DEL 06 AL 08 DICIEMBRE DE 2018

Actas del Congreso Iberoamericano de Docentes

El pensamiento computacional a través de la
programación y la robótica en estudiantes de Media
General de IEA

Alejandro Del Mar

Lizet Mujica

ISBN: 978-84-948417-0-5

Edita **Asociación Formación IB.**

Coordinación editorial: **Joaquín Asenjo Pérez, Óscar Macías Álvarez, Patricia Ávalo Ortega y Yoel Yucra Beisaga**

Año de edición: **2018**

Presidente del Comité Científico: **César Bernal.**

El I Congreso Iberoamericano de Docentes se ha celebrado organizado conjuntamente por la Universidad de Cádiz y la Asociación Formación IB con el apoyo del Ayuntamiento de Algeciras y la Asociación Diverciencia entre otras instituciones.

<http://congreso.formacionib.org>



red
iberoamericana
de docentes



formaciónib))

“El pensamiento computacional a través de la programación y la robótica en estudiantes de Media General de IEA”.

Autores:

Alejandro Del Mar adelmar@iea.edu.ve

Lizet Mujica limujica@iea.edu.ve

INSTITUTOS EDUCACIONALES ASOCIADOS
ASOCIACIÓN CIVIL

Índice

Págs.	
Resumen
Abstract
Introducción
Objetivos
Capítulo I
Marco Teórico
Capítulo II
Metodología
Análisis de
resultados	
Capítulo III
Conclusiones
Referencias

Tablas No.

Título

No.1	Operacionalización de variables.
No.2	Items que componen el test de inicio.
No.3	Habilidades del pensamiento computacional
No.4	Entornos de programación orientada a objetos
No.5	Aspectos asociados al desarrollo del pensamiento computacional.

Figuras No.

Título

No.1	Distribución de calificaciones obtenidas en la prueba .Octubre 2016
No.2	Distribución de calificaciones Febrero 2018

Resumen

Este trabajo se realizó en Institutos Educativos Asociados, con estudiantes de Educación Media General en edades de 14 a 16 años. Se pretendió estudiar las características del pensamiento computacional presentes en la cohorte de alumnos de 3er. año que cursaron la cátedra de robótica en 1ero y 2do año. Se buscaron las

estrategias de enseñanza más idóneas para trabajar el desarrollo del pensamiento computacional. Se aplicó una prueba o pre-test al culminar el 1er. año de bachillerato y una segunda prueba al mismo grupo, cursando el 3er. año. El análisis de resultados, arrojó evidencias de la existencia de cierto nivel hacia la consolidación de aspectos fundamentales del pensamiento computacional: la Resolución de problemas, el análisis, la abstracción y la Secuenciación. La programación es una vía idónea para desarrollar este tipo de pensamiento pero requiere que el docente identifique bien los procesos cognitivos vinculados que desea potenciar. Quedó claro que la manipulación y construcción de objetos tecnológicos ayuda a comprobar muchas de las abstracciones matemáticas o físicas. Se recomienda, acompañar las evaluaciones de este tipo de pensamiento con otras estrategias de evaluación que corroboren a través de portafolios o entrevistas los avances de los estudiantes.

Palabras claves. Pensamiento computacional, Resolución de problemas, Programación, Robótica.

Abstract

This work was carried out in Associated Educational Institutes, with students of General Media Education in ages of 14 to 16 years. The aim was to study the characteristics of computational thinking present in the cohort of 3rd graders. year they studied the robotics department in 1st and 2nd year. The most suitable teaching strategies were sought to work on the development of computational thinking. A test or pre-test was applied at the end of the 1st. year of baccalaureate and a second test to the same group, studying the 3rd. year. The analysis of results, showed evidence of the existence of a certain level towards the consolidation of fundamental aspects of computational thinking: problem solving, analysis, abstraction and sequencing. Programming is an ideal way to develop this type of thinking but it requires the teacher to identify well the linked cognitive processes that he wants to promote. It became clear that the manipulation and construction of technological objects helps to check many of the mathematical or physical abstractions. It is recommended to accompany the evaluations of this type of thinking with other evaluation strategies that corroborate the progress of the students through portfolios or interviews.

Keywords. Computational thinking, problem solving, programming, robotics.

Introducción

La riqueza de experiencias que se viven en la escuela son origen inagotable de datos para la investigación. Esa información a veces se pierde. Es por ello que los docentes toman esa fuente que es el aula de clase, para nutrirse y desarrollar estrategias de enseñanza que mejoren su práctica.

Recientemente IEA creó la cátedra de robótica. Al convertirse en docentes de esta área los autores de esta investigación, buscando estrategias idóneas para la enseñanza de esta disciplina, encontraron que posee una teoría en desarrollo.

Cuenta con un movimiento promotor del pensamiento computacional: la Academia de las Ciencias en Francia, la Sociedad Científica Informática en España, la Asociación Europea de Computación o el movimiento Code.org de EEUU. (INTEF,

2018). No hay consenso sobre lo que abarca el pensamiento computacional ni sobre cómo evaluarlo (Brenann y Resnick, 2012). De allí, surgió el interés.

Objetivo General:

Estudiar el estado actual del pensamiento computacional en alumnos de 3er año a través de estrategias didácticas ligadas a la robótica y a la programación.

Objetivos Específicos:

1. Determinar cuáles son las experiencias didácticas idóneas para el desarrollo del pensamiento computacional vinculadas a la robótica y la programación.
2. Indagar sobre los rasgos del pensamiento computacional presentes en estudiantes de 3er año de Media General de IEA luego de trabajar dos (2) años escolares con robótica.

Capítulo I

MARCO TEÓRICO

Papert (1980) y su equipo del Tecnológico de Massachusetts, crearon el primer lenguaje de programación para niños orientado a objetos (Logo) para convertirlos en creadores de tecnología.

Wing (2006), considera el pensamiento computacional como “un proceso que implica resolver problemas, diseñar sistemas y entender el comportamiento humano haciendo uso de los conceptos de la informática”(pp.33). Papert (1980), se adelantó a esta idea pero no en los términos de Wing.

Según Merelo (2014), los rasgos de este pensamiento lógico son: la búsqueda de patrones, la abstracción y la identificación de propuestas de solución. Todos presentes en la programación.

Wing (2006), citado por Basogain, Olabe y Olabe (2015) describió algunas de las principales características del pensamiento computacional:

Reformular el problema

Pensar recursivamente

Procesar en paralelo

Interpretar códigos como datos y datos como códigos

Elegir una correcta representación o modelo para hacer tratable el problema

Utilizar el pensamiento heurístico para encontrar la solución

Planificar y aprender en presencia de incertidumbre

El interés por enseñar a programar está en los siete pilares de la Agenda Digital Europea (2016). Open Roberta lab, software abierto creado por el Instituto Fraunhofer de Análisis Inteligente y Sistemas de Información promueve el aprendizaje de la programación, posee un simulador y reproduce lo programado en línea.

ANTECEDENTES DE LA INVESTIGACIÓN

Brennan y Resnick (2012), evaluaron el desarrollo del pensamiento computacional en usuarios de Scratch con una matriz de tres dimensiones: conceptos (con pruebas de conocimiento), prácticas (Observación directa) y perspectivas computacionales (entrevistas)

Ceaterra (2015), creó un instrumento para medir el pensamiento computacional basado en las categorías establecidas por Brennan y Resnick (2012) correlacionando los resultados con la opinión de los docentes. Arranz y Pérez (2016) estudiaron niños de Primaria que trabajaron con Scratch usando el instrumento de Ceaterra (2015). Señalan que el uso de Scratch beneficia el pensamiento computacional.

Fuentes (2017) propuso una plataforma Web y una serie de herramientas de programación empleando Blockly, Scratch, Snap, Gamefroot y App inventor.

Finalmente todos apuntan hacia la estimulación del pensamiento computacional a través de la programación.

CAPÍTULO II

Marco Metodológico

Tipo de estudio

Descriptivo. Trabajo de campo exploratorio.

Población

104 estudiantes que recibieron clases de robótica en 1ero y 2do año de Educación Media General en IEA durante los años 2015-2016 y 2016-2017. Actualmente cursan 3er. año con una edad promedio de 15 y 16 años.

Muestra

Aleatoria cercana al 40% de la población estudiantil de 3er. año con 61 estudiantes.

Variable independiente:

Metodología seguida durante las clases de Robótica con los estudiantes de 1ero y 2do año.

Variable dependiente:

Presencia de rasgos del pensamiento computacional en estudiantes de 3er. año de Educación Media General. Se tomaron algunas características propuestas por Wing (2006), agrupadas según las dimensiones de Brenann y Resnick (2012) conceptos básicos de programación, prácticas y perspectivas computacionales.

Opracionalización de Variables

VARIABLES	DIMENSIONES	INDICADORES
Conceptos básicos de programación.	Hardware y piezas	Definición de programación Conocimiento de los dispositivos electrónicos Identificación de piezas
Prácticas y perspectivas computacionales.	Programación y uso del software	Identificación del código Interpretación del código mostrado Estructuras de control <i>Condicionales Bucles Espera</i> Uso de sensores Uso de actuadores <i>Motores lámparas.</i>
	Resolución de problemas: Análisis, abstracción, secuenciación.	Pensar recursivamente, Procesar en paralelo. Juzgar un programa por simplicidad de diseño Diseñar algoritmos Proponer alternativas de solución.

Tabla No 1 Operacionalización de variables Elaboración propia.

Procedimientos:

1. Test de entrada sobre rasgos del pensamiento computacional en las dimensiones identificadas por Brenann y Resnick (2012). Se diseñaron 20 ítems basados en dos áreas de la robótica construcción y programación.

Componente	Categorías	Ítems
Hardware y piezas	Conocimiento de los dispositivos electrónicos	1 2 3 4 y 5
	Identificación de piezas	6 7 y 9
	Uso adecuado de piezas	8 y 10
Programación y uso del software	Identificación e interpretación del código	11 12 13 14 15 y 16
	Estructuras de control <i>Condicionales</i>	12 y 20
	<i>Bucles</i>	14 16 17 18 10 y 20
	<i>Esperar</i>	13 14 17 18 y 19
	Uso de sensores	12 y 20
	Uso de actuadores <i>Motor, Lámpara y Sonido</i>	12 13 14 16 17 18 19 20 14 19
	Uso de actuadores <i>Motor, Lámpara y Sonido</i>	12 13 14 16 17 18 19 20 14 19


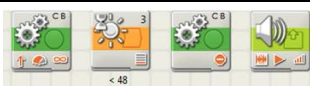






Tabla No.2 ítems que componen el test de inicio. Elaboración propia.

Se incluyeron ítems vinculados con habilidades del pensamiento computacional.

Habilidades del pensamiento computacional	Resolución de problemas	Items 8 y 10
	Análisis	Items 12 y 16
	Abstracción	Items 17 18 19 y 20
	Secuenciación	Items 11 13 y 14

Tabla No3 Habilidades del pensamiento computacional. Arranz, H. y Pérez, A. (2017) Evaluación del pensamiento computacional en educación. *RIITE. Revista Interuniversitaria de Investigación en Tecnología Educativa* 3, 25-39

Se creó una matriz con cuatro aspectos vinculados al Pensamiento Computacional que pueden desarrollarse en los entornos utilizados.

1er año		2do año			
	Scratch	NXT-G	Code.org	Open-Roberta	
Secuencia					
Repetición					

Decisión				
Espera				

Tabla No 4 Entornos de programación orientada a objetos. Elaboración propia

2. Organización de las sesiones de clase

2 horas pedagógicas semanales durante el año. Trabajo con Lego Mindstorms con una problemática y una guía didáctica. Dado el reto, los estudiantes consultan los materiales y realizan las actividades con la mediación del docente trabajando con 2 entornos de programación NXT-G y Open-Roberta.

Análisis de resultados

Para ver los resultados del test completo puede visitar este enlace

<https://drive.google.com/file/d/0B8W5cvx69L1vdGEwdTNQZ3hEZhdweGVSa2IMNkIWY2VsTUVN/view?usp=sharing>

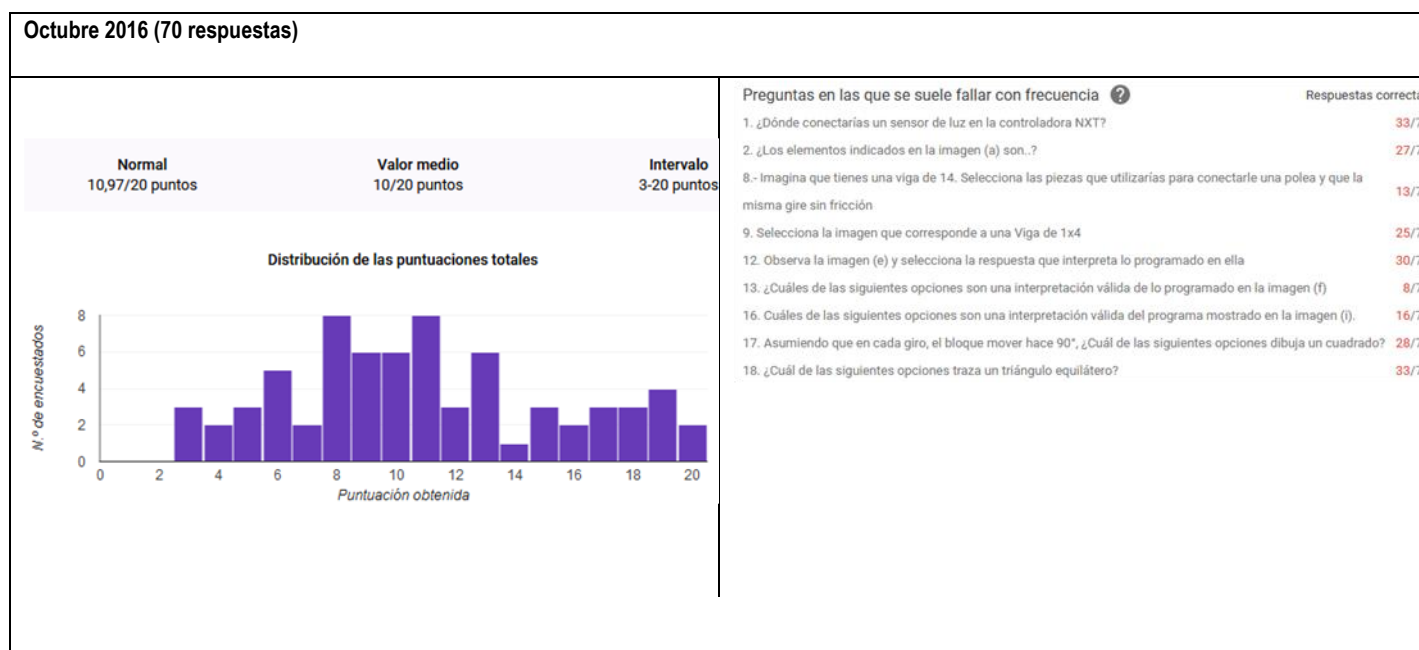


Figura No.1 Distribución de calificaciones obtenidas. Octubre 2016

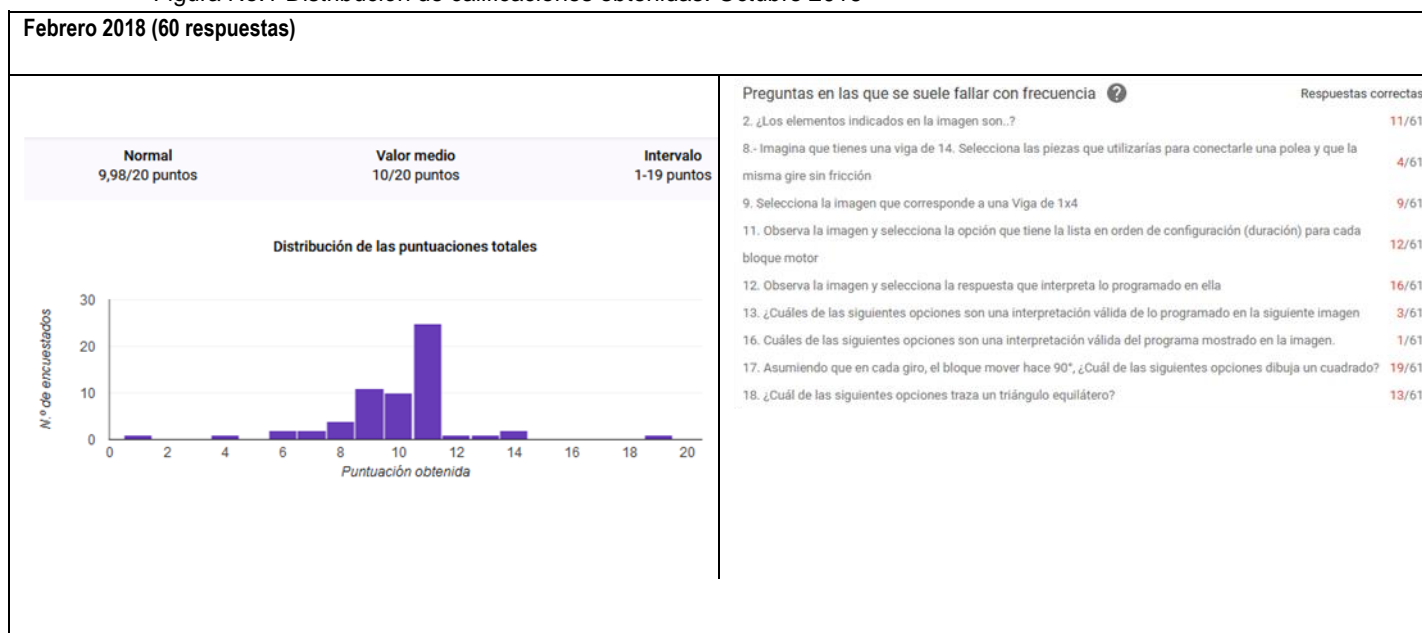


Figura No.2 Distribución de calificaciones obtenidas. Febrero 2018

El 45% del total de las preguntas en ambas ejecuciones tuvieron respuestas correctas cercanas al 50%. La segunda ejecución se realizó tipo sorpresa, sin aviso y luego de finalizadas las clases.

Seguidamente conceptos de la prueba que se asocian a procesos cognitivos que permiten identificar problemas y patrones para resolverlos aplicando la programación.

Aspectos asociados en el desarrollo del pensamiento computacional			
Componente	Categorías	Ítems asociados a la categoría	Comentarios de análisis relacional entre 1ra y 2da ejecución
Hardware y piezas	Manejo de dispositivos electrónicos	Respuestas correctas en 1ra y 2da ejecución: ítem 1: 47,1% y 55,%, ítem 2 38,6% y 18%, ítem 3 55,7% y 59%, ítem 4 82,9% y 95,1%, ítem 5 87,1% y 91,8%	El 80% de las respuestas mejoraron en la segunda versión. Se considera que los estudiantes tienen conocimiento de los dispositivos electrónicos
	Identificación de piezas	Respuestas correctas en 1ra y 2da ejecución Ítem 6 61,4% y 85,2% Ítem 7 65,7% y 85,2% Ítem 9 35,7% y 14,8%	El 67% aumentó la cantidad de respuestas correctas en la segunda ejecución por tanto, puede afirmarse que los estudiantes identifican de manera general las piezas. Puede mejorar

	Uso adecuado de piezas	Respuestas correctas en 1ra y 2da ejecución Ítem 8 62,9% y 59% Ítem 10 67,1% y 70,5%	El 100% de los ítems tuvieron más respuestas correctas en la segunda ejecución y en ambos casos el porcentaje de logro fue superior a 59%. Se concluye que los estudiantes dominan el uso de las piezas
Programación y uso de software	Identificación del código	Respuestas correctas en 1ra y 2da ejecución Ítem 11 65,7% y 60,7% Ítem 14 78,6% y 72,1% Ítem 15 68,6% y 75,4%	En los 3 ítems, 2 de ellos bajaron su desempeño en la segunda ejecución; pero mantienen un porcentaje de respuesta correcta por encima del 60% sugiere dominio en identificación del código
	Estructuras de control y condicionales	Respuestas correctas en 1ra y 2da ejecución Ítem 12 42,9% y 26,2% Ítem 20 48,6% y 63,9%	De los 2 ítems, uno bajó casi la mitad de puntos en respuestas correctas llegando a ser 26%. Aunque el segundo ítems mejoró es importante reforzar

Tabla No.5 Análisis de los resultados hardware, piezas y programación

CAPÍTULO III

Conclusiones y Recomendaciones

Se evidenció en los sujetos de estudio, la existencia aunque no consolidada, de aspectos del pensamiento computacional: la resolución de problemas, el análisis, la abstracción y la secuenciación, al registrar más del 40% de las respuestas correctas sobre la resolución de problemas que implicaban la identificación del código, programación y uso del software.

Los resultados no son desfavorables pues la 2da prueba se efectuó luego de varios meses sin previo aviso.

Vincular la práctica con actividades de reflexión y análisis; para que los estudiantes ensayen y reflexionen sobre sus planteamientos para resolver un problema, así como la construcción de objetos tecnológicos, para comprender abstracciones matemáticas, físicas o electrónicas son estrategias didáctica recomendables.

REFERENCIAS

Arranz, H. y Pérez, A. (2017). *Evaluación del pensamiento computacional en educación. RIITE*. Revista Interuniversitaria de Investigación en Tecnología Educativa, 3, 25-39

Basogain, X., Olabe, M. y Olabe, J. (2015). *“Pensamiento computacional a través de la programación: Paradigma de Aprendizaje”*. RED-Revista de Educación a Distancia, 46(6). 15-Sept.-2015. PDF.

Brennan K. y Resnick M.(2012) *Nuevos marcos de referencia para estudiar y evaluar el pensamiento computacional*. Recuperado en marzo de 2018 en

<http://www.eduteka.org/EvaluarPensamientoComputacional.php>

Cearreta,I. (2015) *Scratch recurso didáctico para el desarrollo del Pensamiento Computacional de los alumnos de Secundaria y Bachillerato en la asignatura de Informática*. Universidad Internacional de la Rioja, Zumaia (Guipuzcoa). Recuperado en marzo de 2018 en:

https://es.slideshare.net/search/slideshow?searchfrom=header&q=herramienta+para+evaluar+el+pensamiento+computacional+%28cearreta%2C+2015%29&ud=any&ft=all&lang=**&sort=

Fuentes P., Alexis D.(2017). *Desarrollo y evaluación del pensamiento computacional. Una propuesta metodológica y una herramienta de apoyo*. Universidad de La Laguna. Escuela Superior de Ingeniería y Tecnología. pdf.

García, José M. (2015). Robótica educativa. *“La programación como parte de un proceso educativo”*. RED-Revista de educación a distancia. 46(8).15-septiembre 2015. Facultad Latinoamericana de Ciencias Sociales. FLACSO. Uruguay. pdf.

INTEF. (2018) *Programación, robótica y pensamiento computacional en el aula. Situación en España*. Recuperado en noviembre de 2017 en <http://educalab.es/-/informe-sobre-programacion-robotica-y-pensamiento-computacional-en-el-aula-situacion-en-espana>

Lopes, A., Lopes, F. y Guedes. A. (2015). *Experiencias de robótica educativa*. Revista internacional de Tecnología, Ciencia y Sociedad. Volumen 4, No.2. pp 192-204. Pdf. Recuperado en enero de 2018 en: <http://tecnociencia-sociedad.com>

Merelo, J. (2014). *“La enseñanza del pensamiento algorítmico debe empezar en primaria”*. Recuperado en mayo 2017 en

<https://www.genbetadev.com/entrevistas/la-ensenanza-delpensamiento-algoritmico-debe-empezar-en-primariaentrevista-a-juan-julian-merelo>

Open Roberta lab. Recuperado en mayo 2017 en <https://lab.open-roberta.org/lio>

<https://www.iais.fraunhofer.de/>

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*, Nueva York: Basic Books, Inc. Recuperado en enero 2018 en:

<http://worrydream.com/refs/Papert%20-%20Mindstorms%201st%20ed.pdf>

Wing, J. (2010). *Pensamiento computacional:¿Qué y Por qué?.pdf*. Recuperado en febrero de 2018 en:

<https://es.scribd.com/doc/129100891/Pensamiento-Computacional-Que-y-Por-Que-Jeannette-Wing>