

DESARROLLO DEL PENSAMIENTO COMPUTACIONAL COMO HERRAMIENTA FUNDAMENTAL PARA EL DESARROLLO DE LAS TECNOLOGÍAS EMERGENTES

Isabel Lafuente Reboredo

Profesora de Tecnología, IES José García Nieto (Las Rozas)



Abstract

The term Computational Thinking was coined by Jane Wing in 2006. Making it a term applicable to all fields of knowledge and not only to programming. The term together with Critical Thinking is the basis of numerous educational research, due to the benefits of its development at an early age.

In this research, a large sample of students in 1st, 4th ESO and first year of Baccalaureate have been compared using average grades, proving that the development of this basic competence is more significant in students in the first year of ESO. It could be said that a generational gap is established already in ESO itself and that there are differences in programming skills. In addition, the non-existence of the middle ground in the works is verified. That is, either they are particularly good, with grades above 8 or they are so bad that they do not even deliver.

In both 1st and 4th grades, there are approximately 20% of students whose computational thinking development is so high that the tasks are very simple for them, and they finish them quickly and efficiently, often exceeding the teacher's expectations. In singing, the learning curve is much higher in the first year of ESO, reaching stabilization values when the task becomes complicated.

This does not happen in the first year of Baccaulaureate, where it is very difficult to quantify the impact that teaching during the pandemic has had on these students, not only academically, but also on a human level.

Resumen

El término Pensamiento Computacional fue acuñado por Jane Wing en el año 2006, por lo que es un término aplicable a todos los campos del conocimiento y no sólo a la programación. El término junto con el Pensamiento Crítico es la base de numerosas investigaciones educativas, debido a los beneficios de su desarrollo en edades tempranas.

En esta investigación se ha comparado una amplia muestra de alumnos de 1º, 4º de ESO y primer curso de Bachillerato utilizando las notas medias, demostrando que el desarrollo de esta competencia básica es más significativo en los alumnos de primer curso de ESO. Se podría decir que ya en la propia ESO se establece una brecha generacional y que existen diferencias en las habilidades de programación. Además, se constata la inexistencia de término medio en las obras. Es decir, o son particularmente buenos, con notas superiores a 8, o son tan malos que ni siquiera cumplen.

Tanto en 1º como en 4º grado, hay aproximadamente un 20% de estudiantes cuyo desarrollo del pensamiento computacional es tan alto que las tareas les resultan muy sencillas y las terminan rápida y eficientemente, superando muchas veces las expectativas del profesor. En canto la curva de aprendizaje es mucho mayor en primer año de la ESO, alcanzando valores de estabilización cuando la tarea se complica.

Esto no ocurre en primer curso de Bachillerato, donde es muy difícil cuantificar el impacto que la docencia durante la pandemia ha tenido en estos alumnos, no sólo a nivel académico, sino también humano.

Keywords

Computational thinking; algorithm; compiler; debug; flowchart.

Palabras Clave

Pensamiento computacional; algoritmo; compilar; depurar; diagrama de flujo.

1. Introducción

El impacto de las nuevas tecnologías digitales es innegable. Estamos sufriendo cambios en nuestros hábitos con gran celeridad. Y cada cambio requiere habilidades que se supone que son innatas o intuitivas. Aprendimos a conectarnos a Internet, sin saber su funcionamiento. Aprendimos a crear contenido sin saber nada de HTML. Y en un futuro próximo, posiblemente se nos exija ser buenos programadores sin apenas conocimientos de programación.

Como profesora de Tecnología he tenido que enseñar programación a mis alumnos y lo he hecho en once lenguajes de programación diferentes (Logo, Basic, Visual BASIC, C++, Python, Processing, JavaScript, Lego, Scratch, App Inventor, Arduino). Algunos de ellos sin formación previa y otros después de formarme en algún curso. Debo admitir, que escuché la palabra PENSAMIENTO COMPUTACIONAL en uno de ellos.

Se considera que el concepto fue desarrollado por Jeannette Wing, a partir de las ideas de Seymour Papert. Jeannette Wing, trata de describir con este concepto que la forma de pensar de los estudiantes de computación se podría trasladar a otros muchos campos y que el desarrollo de estas destrezas era preciso para mejorar a las generaciones venideras. ¿Y qué es el pensamiento computacional? Es una forma de resolver problemas que consta de 4 pasos principales:

- 1.- Descomponer el problema en problemas sencillos de fácil solución.
- 2.- Buscar patrones.

3.- Abstraerse.

4.- Crear un algoritmo.

Y ahí está la belleza de la teoría. Se utilizan algoritmos para resolver multitud de problemas, no sólo en programación. El desarrollo de esta destreza mejora la toma de decisiones y facilita la vida en múltiples aspectos. Los lenguajes de programación pueden cambiar con las modas o las necesidades, pero su diseño, su algoritmo, siempre será el mismo. De hecho, he utilizado los mismos algoritmos con diferentes lenguajes de programación a lo largo de mi práctica docente

Mi experiencia como profesora de programación y tecnología corrobora la importancia de esta destreza. Su correcto uso es clave para aprender programación y para desarrollar con éxito cualquier proyecto que se nos proponga.

En este artículo se recogen una serie de actividades orientadas a desarrollar el pensamiento computacional y su aplicación a diferentes campos como son la programación con Scratch, con APP Inventor y a la robótica con Arduino, en diferentes niveles dentro del mismo centro, esperando que dichas actividades y experiencia sean útiles para el resto de los profesores que quieran trabajar esta habilidad.

Dichas actividades han sido desarrolladas en el IES José García Nieto de Las Rozas. Aunque existe mucha diversidad de alumnado como en cualquier otro centro, se trata de alumnos provenientes de un entorno familiar de clase media alta y con alto nivel cultural (la mayoría de los padres tienen estudios universitarios y ejercen sus carreras). Se han seleccionado 3 grupos de alumnos: 1º de la ESO en la asignatura de Ciencias de la Computación, 4º ESO en la asignatura de Digitalización y 1º de Bachillerato en la asignatura de TEIG (Tecnología e Ingeniería). Todas las asignaturas son optativas y se puede consultar su Currículo.

2. Revisión Bibliográfica

Desde que 2006 Jane Wing definió el término Pensamiento Computacional, en una columna de opinión publicada en el número de marzo de 2006 de la revista *Communications of the ACM*, como una metodología para resolver problemas aplicables no sólo a la programación, sino también a todos los campos de la vida cotidiana. Se han realizado numerosas investigaciones para demostrar la viabilidad el procedimiento.

Ha sido el protagonista en numerosos congresos de educación, en los que algunos autores como Miguel Zapata (2020), le han establecido como competencia básica. Considerando la velocidad en la que se desarrollan las nuevas tecnologías y la importancia que el Pensamiento Computacional tiene para su desarrollo. Su importancia se ha visto incrementada con la crisis de los profesionales STEAM, consecuencia, según algunos autores, de la falta de desarrollo del pensamiento computacional en edades tempranas. Por dicho motivo, proyectos como Scratch y App Inventor, fueron desarrollados por el Massachusetts Institute of Technology (MIT) con la finalidad de mejorar dicha competencia. Fundación Telefónica ha realizado diversos estudios para verificar su correcto desarrollo en edades tempranas y las posibles repercusiones que esto podría tener.

3. Objetivos

3.1. Objetivo general

El principal objetivo de este artículo es mejorar las estrategias que permiten el desarrollo del pensamiento computacional en los niños y adolescentes, para la resolución de problemas tanto de programación como de la vida cotidiana. Desarrollar el pensamiento computacional es preciso para mejorar las habilidades de programación de nuestros alumnos, lo que parece necesario en un futuro próximo debido al desarrollo de las nuevas tecnologías.

Está claro que el pensamiento computacional compite en el asunto de resolución de problemas con otras estrategias como el método científico y el método de resolución de problemas en tecnología. Pero la gran ventaja del pensamiento computacional frente a ellos es que no les resta importancia, al contrario, les incluye como casos particulares del propio pensamiento computacional.

3.2. Objetivos específicos

1. Desarrollar el pensamiento computacional para resolver problemas tanto de computación como de otro tipo.
2. Expresar algoritmos usando los símbolos propios de los diagramas de flujo distinguiendo entre estructuras básicas secuenciales, ramificadas e iterativas.
3. Compilar los algoritmos diseñados en diferentes lenguajes de programación, estableciendo una comparativa dependiendo del lenguaje de programación.
4. Depurar los programas compilados, mejorando su funcionamiento y eliminando los posibles errores de programación existentes.

4.- Contextualización

4.1. Características del entorno

El trabajo ha sido realizado en un IES de la DAT Oeste. Geográficamente el entorno es muy bonito y cuidado. El Ayuntamiento de la localidad está muy implicado y organiza eventos con los diferentes centros de la zona, tales como desafíos de robótica, olimpiadas escolares de todo tipo y charlas sobre salud mental de los alumnos.

4.2. Descripción del centro

El centro está situado cerca de un centro comercial y en una zona residencial rodeada de tiendas y restaurantes. El alumnado proviene de las zonas cercanas al centro. El nivel social del entorno es de clase media alta, como ya se ha descrito con anterioridad.

4.3. Características del alumnado

El interés por aumentar los conocimientos es elevado y, la mayoría de los alumnos tienen planificado acceder a grados universitarios como sus progenitores. Por otro lado, son alumnos sanos en todos los sentidos. Sus aficiones principales son el deporte y las relaciones sociales, y aunque se ha detectado un excesivo número de horas del *smartphone*, no se ha discriminado si en dicho número de horas (unas 4 h al día) se incluye la posibilidad de uso para realizar tareas académicas.

En cuanto a nivel disciplinario, como en todos los centros, existen alumnos cuyo comportamiento en clase es disruptivo y con los cuales se deben tomar las correspondientes medidas disciplinarias.

5. Metodología

La gran aportación de la LOMLOE a las enseñanzas anteriores es el cambio metodológico que supone dejar de enseñar saberes y comenzar a enseñar competencias. Si partimos de que el Pensamiento Computacional es una sub-competencia de la propia Competencia Digital, podremos plantearnos qué actividades pueden ayudar a mejorar dicha competencia. Pero, además, dado la gran celeridad de la evolución de las nuevas Tecnologías, también podríamos plantearnos si existe una brecha generacional en el desarrollo de esta habilidad, y las consecuencias que esto puede tener de cara futuras generaciones.

Para evaluar todo esto, se han establecido comparativas en el aprendizaje de dicha habilidad a diferentes niveles. Es cierto, que también se han utilizado diferentes herramientas para trabajar el Pensamiento Computacional en cada nivel. Sin embargo, también es cierto que, en todas ellas, se han utilizado diagramas de flujo y algoritmos semejantes. Por lo que, aunque se pueda cuestionar si son comparables, en mi criterio lo son.

Para fomentar el cambio metodológico es preciso el uso de metodologías activas. La organización de las aulas de informática es la tradicional. Los alumnos trabajan en parejas. Se les suele proponer un reto, que debe ser explicado en unos 15 minutos al comienzo de la clase. Generalmente también se les proporciona tutoriales que les facilita la tarea. Durante cada sesión los alumnos desarrollan la competencia de aprender a aprender mediante un aprendizaje por descubrimiento basado en el método de ensayo y error. Existe, además, una metodología mezcla del aprendizaje basado en proyectos y el de servicio. En general, las producciones que se han obtenido son de gran calidad. Una vez corregidas se hace un *feedback* para que el alumnado sepa cuáles han sido sus errores.

Los materiales proporcionados a los alumnos cumplen los requisitos del Diseño Universal de Aprendizaje (DUA). Los videos (tutoriales) están subtítulos, y se les proporcionan diferentes opciones en cuanto al cambio de fuente y su tamaño. Además, se han cuidado los agrupamientos para que los alumnos menos favorecidos se vean apoyados y ayudados por otros alumnos con mayor capacidad. De hecho, el ritmo de aprendizaje del alumnado es muy diferente. Se ha observado que hay alumnos con habilidades de programación avanzadas y que rápidamente resuelven los retos propuestos frente a otros alumnos que precisan de más tiempo. Para homogeneizar el ritmo de la clase, dichos alumnos ayudan a sus compañeros más lentos en la resolución de las posibles dificultades que les pudieran suponer un freno en su avance.

6. Resultados y discusión

Lo primero que deberemos de tener en consideración es qué parámetros podremos utilizar para realizar una comparativa en lo referente al tema que estamos tratando, el desarrollo del pensamiento computacional en nuestros alumnos. Obviamente, uno son las calificaciones obtenidas por los alumnos. Pero los resultados de dichas calificaciones pueden estar afectados por otros factores, cuya consideración no es fácil de analizar. Entre otros, destaca el número de alumnos del grupo. Un grupo de 20 alumnos es más sencillo de manejar y esto se verá reflejado en los resultados. Además, si comparamos a los alumnos de 1º de la ESO con alumnos de 1º de Bachillerato hay otro factor que deberíamos de considerar. Los alumnos de 1º de Bachillerato han visto alterada su enseñanza por la Pandemia y la metodología que se utilizó. Por un lado, debieron ser capaces de desarrollar estrategias en el uso de herramientas electrónicas, pero por otro, la calidad de los conocimientos teóricos y, como consecuencia, la adquisición de un adecuado desarrollo del pensamiento computacional es cuanto menos, cuestionable. Teniendo en consideración que la adquisición de nuevo conocimiento requiere el silencio psicológico y una buena concentración. En una situación de estrés resulta muy complicado adquirir nuevo conocimiento. En cualquier caso, para un mejor análisis se han agrupado los resultados por el tipo de algoritmo (tabla 1):

Tabla 1

Calificaciones medias, para diferentes estructuras de algoritmos

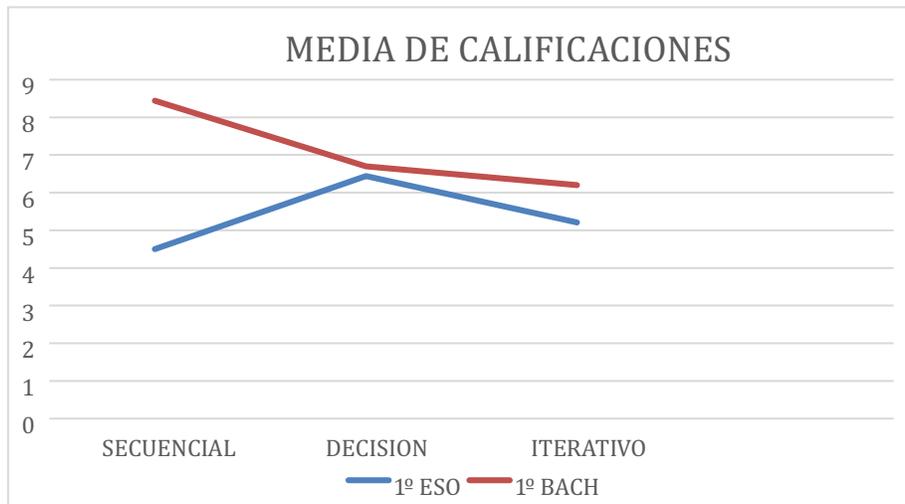
	Secuencial	Ramificado	Iterativo	Media
1º ESO	4,5	6,44	5,21	5,6
1º BACH	8,44	6,7	6,2	7,11

Nota: La tabla muestra la media aritmética de las calificaciones de los alumnos para diferentes estructuras algorítmicas, así como la media de dichas calificaciones.

Pariendo de dichos datos, se ha realizado una gráfica sobre las diferentes curvas de aprendizaje en diferentes niveles (figura 1).

Figura 1

Curvas de aprendizaje dependiendo de la curva del algoritmo



Nota: El gráfico muestra las calificaciones medias obtenidas por los alumnos de 1º de la ESO y los de 1º de Bachillerato en el aprendizaje de algoritmos secuenciales, ramificados e iterativo.

A la vista de estos resultados, se puede establecer que el alumnado de 1º de Bachillerato tiene adquirida la estructura secuencial, pero el resto de las estructuras no están tan claras. En cuanto a la velocidad de aprendizaje, es mayor en 1º de la ESO que en 1º de Bachillerato. Esto podría hacer cuestionable los conocimientos previos en este tema de los alumnos de primero de Bachillerato, que podrían haber sido afectados por la pandemia. Con la finalidad de corroborar estos posibles datos, se pueden considerar los obtenidos en 4º de la ESO para estructuras secuenciales dando una media de 5,7 que es intermedia entre los valores de 1º de la ESO y de 1º de Bachillerato.

Para realizar la calificación de las actividades se han utilizado rúbricas, que son semejantes en todos los casos y valoran aspectos parecidos para el caso de 1º ESO y de 4º de la ESO. La

DESARROLLO DEL PENSAMIENTO COMPUTACIONAL COMO HERRAMIENTA FUNDAMENTAL PARA EL DESARROLLO DE LAS TECNOLOGÍAS EMERGENTES

primera rúbrica se refiere a la programación con Scratch en 3 actividades recogidas en los anexos 1, 2, y 3 y en 4º con APP Inventor. Se consideran los siguientes aspectos (figura 2):

Figura 2

Rúbrica de calificación utilizada en algoritmos secuenciales en 1º de la ESO

ESCENARIO Y PERSONAJES	NO HAY <i>0 puntos</i>	FALTA ALGÚN ELEMENTO <i>1 puntos</i>	TODOS LOS ELEMENTOS BIEN <i>2 puntos</i>
DISFRACES	NO HAY DISFRACES <i>0 puntos</i>	FALTAN DISFRACES <i>1 puntos</i>	TODOS CORRECTO <i>2 puntos</i>
CONDICIONES INICIALES	NO EXISTEN <i>0 puntos</i>	FALTAN ELEMENTOS <i>1 puntos</i>	ESTAN TODOS LOS ELEMENTOS <i>2 puntos</i>
PROGRAMACIÓN	NO HAY PROGRAMACIÓN <i>0 puntos</i>	NO ES TOTALMENTE CORRECTA <i>1 puntos</i>	ES CORRECTA <i>2 puntos</i>

En el caso de primero de Bachillerato se ha trabajado además la competencia lingüística, por lo que además de aspectos puramente técnicos, semejantes a los anteriores, se ha considerado la posible redacción del informe (figura 3).

Figura 3

Rúbrica de calificación para algoritmos secuenciales en 1º de Bachillerato

Nº DE PRÁCTICAS	NINGUNA <i>0 puntos</i>	UNA <i>1 puntos</i>	LAS 2 <i>2 puntos</i>
DIAGRAMAS DE FLUJO	NINGUNO <i>0 puntos</i>	UNO <i>1 puntos</i>	2 <i>2 puntos</i>
PROGRAMA ARDUINO	NINGUNO <i>0 puntos</i>	UNO <i>1 puntos</i>	2 <i>2 puntos</i>
RESULTADOS	NINGUNO <i>0 puntos</i>	UNO <i>1 puntos</i>	2 <i>2 puntos</i>
REDACCION Y EXPLICACIONES	NINGUNA <i>0 puntos</i>	POCO DETALLADA <i>1 puntos</i>	DETALLADA <i>2 puntos</i>

Otro aspecto resaltable, es que las tareas o no se mandan o están perfectas. En muy pocas ocasiones se ha enviado una tarea cuya calificación sea 7 u 8. En general, los fallos son muy frecuentes y la mayoría de las tareas enviadas tienen una calificación de 10. Esto podría indicar una brecha cognitiva en el alumnado. Alrededor del 50% de los alumnos de primero tienen muy desarrollada esta habilidad, frente al otro 50% que están en proceso de desarrollo. Durante la etapa de Educación Secundaria Obligatoria (ESO) dicho conocimiento se va desarrollando muy lentamente, lo que corroboraría la hipótesis que es preciso adquirir dicha habilidad cuando la plasticidad del cerebro es mayor, es decir durante la infancia, mejor que en la adolescencia.

7. Conclusiones

Revisando los objetivos planteados en esta investigación es de resaltar, que en todos los casos se ha detectado una curva de aprendizaje satisfactoria en todos los niveles para un desarrollo del pensamiento computacional.

En todos los casos se han creado algoritmos que han sido expresados como diagramas de flujo para realizar programas en diferentes lenguajes de programación (todos ellos de alto nivel), que han sido correctamente compilados y depurados.

Analizando las calificaciones obtenidas por los alumnos en diferentes cursos, se ha detectado que alrededor del 50% del alumnado tiene muy desarrollado el pensamiento computacional, lo que les ha ayudado a resolver los problemas planteados con gran celeridad y sin excesivas complicaciones. Del otro 50%, aproximadamente la mitad ha terminado la tarea, satisfactoriamente y con éxito, aunque con las dificultades propias de un aprendizaje. Hay un pequeño porcentaje que no ha realizado la tarea o que no la ha entregado.

La gran mayoría de las tareas entregadas tienen calificaciones superiores al 8 y de ellas más del 70% son 10. En muy pocas ocasiones las tareas entregadas se han calificado con un 5.

El desarrollo del pensamiento computacional, reflejado a través de las calificaciones no depende del lenguaje de programación utilizado, aunque sí parece depender de la edad. Los alumnos de 1º de la ESO presentan una mejor curva de aprendizaje que los de 1º de Bachillerato.

Esto avalaría la teoría de que cuanto antes se comience a trabajar esta competencia digital mejores son los resultados que se obtengan, en concordancia con la teoría de la plasticidad cerebral al igual que ocurre con los idiomas.

8. Limitaciones y prospectiva

Aunque se ha realizado un estudio del aprendizaje del pensamiento computacional mediante actividades relativas a la programación, sería muy interesante aplicar el pensamiento computacional a otras actividades y verificar su eficiencia en esos casos. Por ejemplo, en la realización de un proyecto complejo, en donde además del pensamiento computacional, el alumnado debería de utilizar el pensamiento crítico y la creatividad como herramientas importantes para su elaboración.

Por otro lado, las claras diferencias entre los alumnos de 1º de la ESO con los de 4º y 1º de Bachillerato deberían de tener en consideración las enseñanzas no presenciales y semipresenciales que tuvieron lugar durante la pandemia y que lógicamente debieron de influir en la calidad de aprendizaje de nuestros alumnos.

9. Consideraciones finales

Personalmente las actividades descritas en este artículo me han sido de gran ayuda para entender las capacidades de mi alumnado y mejorar mi propia competencia digital. Soy consciente de que el grado de dificultad no es muy alto y que se podrían haber completado y actualizado utilizando IA (Inteligencia Artificial), pero creo que se han adaptado a las programaciones propuestas por el departamento y a los actuales currículos.

10. Agradecimientos

Agradezco a mis alumnos su colaboración para realizar las tareas propuestas, al centro su interés por proveer de los medios necesarios y todos los miembros del departamento por su ayuda en la coordinación de las actividades.

11. Referencias bibliográficas

Zapata-Ros, M. y Pérez-Paredes, P. (2020) *El pensamiento computacional. Análisis de una competencia clave*. Virtualidad, Educación y Ciencia.

Ramírez Benavides, K (2018). *Importancia del pensamiento computacional en la educación*. Gobierno de Canarias (día de la libertad del software.)

Cañas Encinas, M., Pinedo González, R. y Palacios Picos, A. (2022). *Prácticas de aula para promover el pensamiento crítico en el uso de los medios digitales*. En *Revista de Educación*, 398, 193–217. <https://doi.org/10.4438/1988-592X-RE-2022-398-557>

12. Anexos

Anexo 1

<https://scratch.mit.edu/projects/955138327/>

Anexo 2

<https://mediateca.educa.madrid.org/video/wgddff7qb5sa1ls4?jwsourc=cl>

Anexo 3

<https://scratch.mit.edu/projects/968901533/>