

Prácticas docentes de aula en la enseñanza del pensamiento computacional en escuelas medias oficiales y particulares de la región metropolitana de la Ciudad de Panamá.

Pág. 207- 230

* Delfina D'Alfonso
** Nyasha Warren
*** Estefany González
**** Abiezer Rodríguez
***** Kathia Pitti
***** Nadia De León

*Praxia Educational Consultants.
Universidad del Istmo.
delfina.dalfonso@gmail.com

**Centro de Investigación Educativa
de Panamá, Instituto de
Investigaciones Científicas y
Servicios de Alta (INDICASAT-
AIP).
warrennyasha@gmail.com

***Universidad del Istmo.
esgonzalez@udelistmo.edu

****Universidad del Istmo.
rodriguezabiezer15@gmail.com

*****Secretaría Nacional de Ciencia,
Tecnología e Innovación de Panamá.
kpitti@senacyt.gob.pa

*****Praxia Educational
Consultants. Universidad del Istmo.
Centro de Investigación Educativa de
Panamá, Instituto de Investigaciones
Científicas y Servicios de Alta
(INDICASAT-AIP).
nadiadeleonporter@gmail.com

Fecha de entrega:
septiembre 2020

Fecha de aceptación:
diciembre 2020

RESUMEN

Este estudio describe y evalúa las prácticas pedagógicas de aula de los docentes en la enseñanza de informática y el pensamiento computacional (PC) en educación media oficial y particular de la región metropolitana de la Ciudad de Panamá, en comparación con referencias internacionales, como la Sociedad Internacional de Tecnología en Educación (ISTE) y otras referencias de la literatura relevante en pedagogía. Se analizaron datos de una muestra de conveniencia conformada por 14 escuelas de nivel medio participantes en las Olimpiadas Nacionales de Informática 2019, abiertas y ampliamente divulgadas. Los datos fueron recolectados por medio de observaciones en aula. Se observó poca evidencia de la utilización de prácticas docentes reconocidas como mejores prácticas para la enseñanza de informática y PC. Hubo algo más de evidencia para el uso de prácticas construccionistas, que para las relacionadas al PC; y aún menos para el uso de prácticas relacionadas al aprendizaje por experiencias. Así mismo son notables las diferencias entre las escuelas oficiales y particulares en cuanto al uso de estas prácticas. Adicionalmente, los hallazgos indicaron un nivel importante de inequidad en el sistema, ya que los puntajes se encontraban polarizados entre escuelas con mucha evidencia y escuelas con poca o ninguna evidencia, y pocas escuelas en el medio.

Palabras clave: prácticas pedagógicas, informática, pensamiento computacional, construccionismo, aprendizaje por experiencias.

Classroom teaching practices in the teaching of computational thinking in official and private middle schools of the metropolitan region of Panama City

Abstract

This study describes and evaluates the classroom pedagogical practices of teachers in the teaching of computer science and computational thinking (CP) secondary education in the metropolitan region of Panama City, in comparison with international references. The International Society for Technology in Education (ISTE) and other references to relevant literature in pedagogy. Data from a convenience sample made up of 14 middle-level schools participating in the 2019 National Informatics Olympics were analyzed, openly publicized. The data was collected through classroom observations. Little evidence was observed of the use of teaching practices recognized as best practices for teaching computer science and CP. There was a bit more evidence for the use of constructionist practices than for those related to CP; and even less for the use of practices related to experiential learning. Furthermore, there are notable differences between official and private schools regarding the use of these practices. Additionally, the findings indicated a significant level of inequity in the system, as the scores were polarized between schools with a lot of evidence and schools with little or no evidence, and few schools in between.

Keywords: Pedagogical practices, computing, computational thinking, constructionism, experiential learning.

INTRODUCCIÓN

El estudio fue realizado en el marco del proyecto de Olimpiadas Nacionales de Informática, para el cual uno de los objetivos principales fue fortalecer las competencias de pensamiento computacional en los estudiantes de media de Panamá. Las Olimpiadas Nacionales de Informática, pretenden a través de los resultados de un concurso anual entre escuelas, aportar en la actualización de la enseñanza de la informática en el país. Al mismo tiempo, se aprovechó la oportunidad para realizar una recolección de datos que permitieran una descripción y diagnóstico de la situación, del cual este estudio forma parte. Específicamente, esta investigación se propone describir la práctica de los docentes de informática y materias afines al pensamiento computacional en las aulas de educación media panameñas y realizar un diagnóstico al compararla con estándares globales de punta.

La enseñanza de la informática y el pensamiento computacional

La historia de la enseñanza de la informática es relativamente corta, pero ha evolucionado de manera muy rápida. Hacia los años 40 surgieron las primeras computadoras programables y, subsecuentemente, las profesiones relativas a su ingeniería y a la ciencia de la computación. Algunos años después, las universidades comenzaron a ofrecer programas de ciencias de la computación. Los mismos formaban parte de departamentos de ingeniería o matemática. No fue hasta 1960 que se instalaron departamentos específicos de computación (Nwana, 1997).

Al analizar la evolución de los currículos de informática a lo largo del tiempo es relevante mencionar como caso pionero, el de Estados Unidos. En 1962 se formó el Comité de Currículo en Informática que fue definiendo a partir de reportes un marco de referencia para el diseño de los currículos de enseñanza de la informática. Al ser el primer plan de estudios, “dedicó considerable atención a justificar la informática como una disciplina nueva y distinta, y a promulgar el nombre de ‘informática’ contra otros nombres rivales, como ciencias de la

información” (Nwana, 1997, p.34). El primer reporte generado describía los cursos y presentaba sus requisitos. Estos se enfocaban principalmente en las ciencias de la computación y las matemáticas, aunque también reflejaban contribuciones de las ciencias físicas y de la ingeniería. El informe fue revisado y actualizado a medida que la misma disciplina fue madurando (Nwana, 1997).

Ahora bien, enseñar en tiempos modernos y en el marco de nuevas tecnologías de la información supone dos tipos de alfabetismo: el mediático (de los medios audiovisuales) y el transmedia (de los medios interactivos y redes digitales). El alfabetismo transmedia aspira a formar usuarios consumidores pero también creadores (Scolari, 2016). La inmersión masiva en el mundo digital demanda cierta alfabetización digital, una competencia digital que es parte esencial del aprendizaje para la vida. Las habilidades necesarias para lograr la competencia digital incluyen “el uso de computadoras para recuperar, evaluar, almacenar, producir, presentar e intercambiar información, y para comunicarse y participar en redes de colaboración a través de Internet” (Ryberg & Georgsen, 2010, p. 89). Este mismo autor advierte que si bien es posible desarrollar gran parte de la competencia digital en ámbitos informales de interacción con la tecnología, las instituciones educativas cumplen un rol importante en este proceso.

La Asociación de Profesores de Informática de Estados Unidos, (en adelante CSTA por sus siglas en inglés), es una comunidad de maestros de informática que desarrolla y actualiza estándares para la enseñanza de la informática, y ofrece recursos para la enseñanza. Tiene como objetivos principales fortalecer, defender e involucrar a los maestros de K-12 (escolarización primaria y secundaria) en todo el mundo, y fomentar y apoyar la enseñanza-aprendizaje en el sector de la informática y áreas en común (CSTA, 2017).

En 2017 publicó un Marco de Enseñanza de la Informática (*K-12 Computer Science Framework*) el cual sostiene que las computadoras son un instrumento creador:

...las computadoras son máquinas rápidas, confiables y potentes que nos permiten construir, analizar y comunicar digitalmente nuestra experiencia humana. Más que una simple herramienta, las computadoras son un medio fácilmente accesible para la expresión creativa y personal. En nuestra era digital,

las computadoras son tanto la pintura como el pincel. La educación en informática crea a los artistas (CSTA, 2016, pp.9-10).

La alfabetización digital permite a los estudiantes convertirse en autores, artistas y expresarse en forma de programas y software. En lugar de ser consumidores pasivos de tecnologías informáticas, a partir de la enseñanza de la informática, pueden convertirse en productores y creadores activos (CSTA, 2016). Este es el estándar mínimo que se aspira alcanzar en la formación actual de nuestros jóvenes.

Pensamiento Computacional

La importancia de la incorporación del pensamiento computacional como competencia universal a los aprendizajes escolares fue ampliamente estudiada en los últimos años. Jeanette Wing, publicó un artículo en el que postulaba “que el pensamiento computacional implica resolver problemas, diseñar sistemas y comprender el comportamiento humano, basándose en los conceptos fundamentales de la informática” (2006, p.716). En el mismo artículo se afirmaba que el pensamiento computacional involucra tres elementos clave: algoritmos, abstracción y automatización. Además, establecía que se trata de una competencia universal que no solo los informáticos deberían adquirir: “esta nueva competencia debería agregarse a la capacidad analítica de cada niño como ingrediente vital de su aprendizaje escolar” (Wing citado en Voogt et al., 2015, p.716). Así mismo Mishra y Yadav agregan que el pensamiento computacional aporta a la transición de consumidores de tecnologías a creadores de nuevas formas de expresión, a construir herramientas y a fomentar la creatividad (citados en Voogt et. al, 2015).

El pensamiento computacional se encuentra estrechamente vinculado al pensamiento lógico. La lógica, en tanto forma específica de pensar, de organizar ideas y representaciones, favorece las competencias computacionales (Zapata-Ros, 2015). Proporciona estructuras para la organización de datos y para la elaboración de pasos en la resolución de determinados problemas computacionales. Al mismo tiempo, la lógica se aplica especialmente a las especificaciones de un software y a su verificación. Además, facilita la lectura, escritura y

comprensión de especificaciones formales y su análisis (Mossakowski, 2010). Por otro lado, modelar escenarios utilizando formalismos lógicos e inferir nuevos conocimientos son habilidades importantes en el campo de la enseñanza de la informática (Geck et al., 2018). Por lo tanto, el aprendizaje de formalismos lógicos y, en particular, el modelado lógico es de suma importancia para los estudiantes de informática. La enseñanza del pensamiento computacional y su incorporación a los currículos escolares es verdaderamente necesaria, ya que no solo permite resolver problemas y diseñar sistemas, sino que también le permite a los estudiantes comprender el potencial y las limitaciones de la inteligencia humana y de las máquinas (Papadakis et al., 2016).

Evolución de la Enseñanza de la Matemática, relevante a Informática

A lo largo de la historia son varias las propuestas en torno a cómo alcanzar mayores aprendizajes en el campo de la matemática. Entre estas, se encuentra la incorporación de la tecnología como un recurso que aporta a la visualización de los conceptos abstractos de la matemática. Esto se debe a que “permiten que se desarrollen actividades desde más de un sistema de representación, es decir no solo desde el enfoque algebraico, sino que también logren visualizar el concepto desarrollado” (Pizarro, 2009, p.30). De esta forma, las computadoras se vuelven un medio para enseñar la matemática. Esto difiere significativamente de su uso como máquinas de cálculo que facilita y aceleran, sino reemplazan los procesos cognitivos humanos.

Así como la enseñanza de la matemática se ve facilitada por la incorporación de tecnologías; el razonamiento lógico matemático es una habilidad fundamental para el desarrollo del pensamiento computacional y conocimientos informáticos. Prácticamente todas las ramas de las matemáticas han tenido su influencia en el desarrollo de uno u otro aspecto de la informática. Por mencionar un ejemplo, los algoritmos requieren el uso de la inducción matemática, la probabilidad, el manejo de series numéricas y la lógica proposicional para verificarlos, entre otros conceptos propios del campo matemático (Sacristán, 2007). Todos los componentes fundamentales que Lee y Chan (2019), presentan para el pensamiento

computacional, tienen su raíz en las habilidades de pensamiento lógico-matemático. Ellos son: la descomposición, el reconocimiento de patrones, el pensamiento algorítmico y la generalización. He aquí la estrecha relación entre matemática y pensamiento computacional al tratarse de habilidades cuyas bases deben ser desarrolladas desde el campo de la matemática, y también en él pueden profundizarse.

Prácticas pedagógicas en la enseñanza del pensamiento computacional

Seymour Papert (1993) fue uno de los primeros en notar la urgente necesidad de un cambio masivo en los diferentes sistemas educativos que reconociera el valor de la tecnología en los procesos de enseñanza-aprendizaje. Al mismo tiempo, destacó la importancia de que dichos cambios se introdujeran de la mano de docentes bien preparados. En el campo de la enseñanza de la informática, este autor retoma la idea de Piaget de estudiantes como constructores de sus propias estructuras intelectuales; basándose en dos supuestos importantes para desarrollar su estudio sobre las computadoras y la educación. En primer lugar, parte de la idea principal de que todos los niños pueden aprender a usar una computadora de manera magistral y, en segundo lugar, que aprender a usar computadoras puede cambiar la propia manera de aprender sobre cualquier otra cosa. Según lo planteado, las prácticas pedagógicas que fomentan en los estudiantes la construcción de sus propias estructuras intelectuales en entornos de aprendizaje que sean amigables con ellos son de vital importancia.

Consecuente con lo anterior, Seymour Papert, a inicios de la década del 70 en el Instituto Tecnológico de Massachussets (MIT por sus siglas en inglés), desarrolló LOGO, un software de programación para niños enfocado en programar el movimiento de una tortuga, para dar un enfoque diferente a los dispositivos informáticos que se usaban en el campo de la educación en ese momento. En muy poco tiempo, LOGO, se transformó en el idioma internacional para la alfabetización informática de la educación primaria. Más tarde, se amplió el alcance del software LOGO para emplearlo en los kits de construcción LEGO. Estas construcciones conectan los manipulables con la programación y la electrónica, y permiten al estudiante aprender a resolver problemas reales. Más adelante, en el MIT se diseñó Scratch, un software

más reciente de programación para niños y app actualmente líder en el campo, con el fin de armar fácil y divertidamente fragmentos de programas de computador, probarlos, desarmarlos y recombinarlos, idealmente alrededor de proyectos creativos que motivan a los niños. Además, se generó una página y comunidad web para que los creadores puedan modificar o crear nuevos proyectos para compartirlos con la audiencia y así motivar a otros a ser desarrolladores (Resnick, 2007). Tras revisar 55 estudios empíricos sobre los aprendizajes que los estudiantes adquieren a través de Scratch, Zhang y Nouri (2019) identificaron: la adquisición de conceptos computacionales con los que los programadores se involucran mientras programan; prácticas computacionales que los programadores desarrollan a medida que interactúan con los conceptos; y la perspectiva computacional que los programadores visualizan sobre el mundo que los rodea y sobre ellos mismos.

Además de la programación utilizada para crear, Papert se enfocó en el uso de manipulables y objetos tangibles para la enseñanza en general en todas las áreas. Csizmadia et al. (2019) destacan la relevancia de los aportes de la teoría construccionista de Papert, en tanto modelo de aprendizaje basado en *hands-on activities*, particularmente en la enseñanza del pensamiento computacional. De igual forma advierten que en este tipo de prácticas pedagógicas se trabaja sobre la autonomía de los estudiantes, los cuales se encuentran, por ejemplo, frente al reto de escoger entre diferentes opciones a la hora de desarrollar un artefacto.

Recientemente, Romeike y Przybylla (2014) diseñaron un plan de base para el desarrollo de futuros currículos de informática que busquen incorporar una metodología constructivista a partir de lo que los autores denominan *physical computing*. La informática concreta o física involucra el diseño y la realización de objetos interactivos y permite, a partir de la creatividad de los estudiantes, el desarrollo de productos concretos y tangibles.

Prácticas derivadas de la propia definición de pensamiento computacional y de estándares para educadores

En el 2011, la Sociedad Internacional de Tecnología en Educación (en adelante, ISTE por sus siglas en inglés) en asociación con CSTA, publicaron su propia definición operacional de pensamiento computacional. En la misma es posible observar ciertas habilidades que los docentes deben promover en sus estudiantes:

Formular problemas de una manera que nos permita usar una computadora y otras herramientas para facilitar su resolución; organizar y analizar lógicamente los datos; representar datos a través de abstracciones como modelos y simulaciones; automatizar soluciones a través del pensamiento algorítmico (una serie de pasos ordenados); identificar, analizar e implementar posibles soluciones con el objetivo de lograr el máximo rendimiento; combinar de manera eficiente y efectiva pasos y recursos; generalizar y transferir el proceso de resolución de problemas a una amplia variedad de problemas (ISTE y CSTA, 2011, p.1).

La ISTE se estableció en 1979 como una organización sin fines de lucro. Actualmente es una de las comunidades de educadores globales más grandes que creen en el poder de la tecnología para transformar la enseñanza y el aprendizaje, acelerar la innovación y resolver problemas difíciles en la educación. En 2011, la organización diseñó estándares para educadores de informática y, en el 2018, publicó estándares más específicos relacionados con la enseñanza del pensamiento computacional.

Particularmente para la enseñanza del pensamiento computacional de manera transversal en todas las materias y grados, ISTE (2018, p. 1-2) presentó estándares para educadores que incluyen:

1) Desarrollar un conocimiento práctico de los componentes centrales del pensamiento computacional: (...), 2) Contrarrestar proactivamente los estereotipos que excluyen a los estudiantes de las oportunidades de sobresalir en informática y fomentar una cultura de aula inclusiva desarrollando la auto eficacia y la confianza de los estudiantes en

torno a la informática, 3) Incorporar diversas perspectivas y habilidades únicas en el desarrollo de oportunidades de aprendizaje de los estudiantes, y reconocer que las habilidades de colaboración deben ser enseñados de manera explícita con el fin de conducir a mejores resultados que los obtenidos cuando trabajan de forma independiente, 4) Reconocer que el diseño y la creatividad pueden fomentar una mentalidad de crecimiento y trabajar para crear experiencias y entornos de aprendizaje de informática significativos, 5) Facilitar el aprendizaje integrando prácticas de pensamiento computacional en el aula, desarrollando la capacidad de cada estudiante para reconocer oportunidades para aplicar el pensamiento computacional en su entorno, dado que el pensamiento computacional es una habilidad fundamental.

La resolución de problemas

Wing (2008 citado en Voogt, 2015) afirma que el pensamiento computacional es un mecanismo para resolver problemas humanos complejos. Según Voogt (2015), el desarrollo de habilidades de pensamiento computacional requiere de prácticas pedagógicas centradas en la resolución de problemas a partir del uso de las computadoras. Además, como se mencionó anteriormente, la informática consiste en *crear* herramientas para resolver problemas, en lugar de solo *usar* las herramientas. Por consiguiente, “otros problemas externos a la informática como disciplina pueden abordarse utilizando las mismas técnicas o técnicas similares. Estas técnicas pueden definirse o resumirse en un algoritmo, un conjunto de instrucciones paso a paso” (Mohaghegh et al., 2016, p.1524).

Adicionalmente, son varios los autores que sostienen que gran parte de la estrecha relación que se establece entre matemática e informática se encuentra en la resolución de problemas, siendo esta una habilidad implicada en la enseñanza de ambas materias (Baldwin et al., 2013; Mohaghegh & McCauley, 2016). Según estos autores, “la informática comparte muchas características con las matemáticas y, por lo tanto, está implícito que también compartirá problemas y técnicas de resolución de problemas” (p.1524). En el aprendizaje basado en problemas los estudiantes trabajan en grupos, de manera colaborativa para resolver problemas

reales complejos con la guía de un docente o facilitador (Allen et al., 2011, pp. 21-29). Esta metodología no se trata específicamente de los problemas que se resuelvan en el aula, sino que los mismos son un medio para ampliar o profundizar conocimientos y adquirir habilidades relacionadas al análisis, la investigación, el debate, la búsqueda y la presentación de información y el pensamiento crítico (Wood, 2003). Duch, Gorh y Allen (2001) por su parte sostienen que en el aprendizaje basado en problemas, los problemas complejos del mundo real se utilizan para motivar a los estudiantes a identificar e investigar los conceptos y principios que necesitan saber para resolver esos problemas. Savery (2015) agrega que el aprendizaje basado en problemas, además de incentivar a investigar e integrar teoría y práctica, incentiva la aplicación de conocimientos y habilidades previas para desarrollar una solución viable para un problema determinado.

El aprendizaje basado en proyectos

El aprendizaje basado en proyectos se enmarca dentro de la resolución de retos, con particular énfasis en la elaboración de un producto final. Esta metodología organiza el aprendizaje en torno a proyectos que implican tareas complejas, con frecuencia basados en preguntas o problemas desafiantes. Los proyectos “involucran a los estudiantes en el diseño, la resolución de problemas, la toma de decisiones o actividades de investigación” y dan “a los estudiantes la oportunidad de trabajar de forma relativamente autónoma durante largos períodos de tiempo, y culminar en productos o presentaciones realistas” (Thomas, 2000, p.1). Un componente importante de esta metodología es también el trabajo colaborativo, la interacción, el diálogo y las discusiones que se producen entre los estudiantes. En la construcción de conocimiento, los estudiantes hacen uso de herramientas cognitivas, de diversas fuentes de información, incluso de otras personas o de lo que las otras personas saben como recurso (Blumenfeld et al., 1991). Además, debido a que el aprendizaje ocurre en un contexto social, “los alumnos interactúan e internalizan modos de conocimiento y pensamiento representados y practicados en una comunidad” (Blumenfeld et al., 1991, p.371). Roschelle et al., (2000) afirman que las tecnologías informáticas integradas en el aprendizaje basado en proyectos son particularmente útiles para desarrollo de habilidades de orden superior como el pensamiento crítico, el análisis y la investigación.

METODOLOGÍA

Se trata de un estudio descriptivo observacional en el aula. Se desarrolló una pauta de observación partiendo de las prácticas pedagógicas efectivas en la enseñanza de la informática y el pensamiento computacional identificadas en el marco teórico y utilizando en gran parte una adaptación de la herramienta de observación de clases elaborada por ISTE. Las observaciones se realizaron con base a 15 criterios (detallados en la sección de resultados) relacionados a prácticas construccionistas, prácticas de enseñanza del pensamiento computacional y prácticas relacionadas a la resolución de problemas y el aprendizaje basado en proyectos. Cada criterio fue evaluado con una escala de Likert, pudiendo evidenciarse hasta qué punto, cada uno era: muy evidente que sucediera (4), evidente (3), poco evidente (2) o no evidente (1). Para el análisis, se calculó el promedio de observaciones que alcanzaron cada nivel de la escala, en cada uno de los criterios observados.

La muestra fue por conveniencia, y estuvo conformada por 14 escuelas de nivel medio inscritas en las Olimpíadas Nacionales de Informática 2019. De ellas, 8 son escuelas oficiales y 6 son particulares, todas pertenecientes a las regiones educativas de Panamá Centro, Panamá Este, Panamá Oeste, Panamá Norte y San Miguelito. Los bachilleratos que principalmente ofrecen son en Ciencias, Ciencias y Letras y Ciencias con Instrumentación en Informática.

RESULTADOS

Con el objetivo de describir la práctica de los docentes de informática y materias afines al pensamiento computacional en las aulas de educación media panameñas, los resultados se presentan agrupados considerando los tres grupos de prácticas que han demostrado ser efectivas en la enseñanza de la informática y el pensamiento computacional. Para cada grupo de prácticas se presentan los porcentajes de cada nivel de evidencia en cada criterio y una comparación entre el promedio de evidencia alcanzado (en una escala del 1 al 4) en todos los criterios por escuelas oficiales y particulares.

En primer lugar, se presentan las prácticas construccionistas que involucran: la manipulación física de materiales; evidencias de enseñanza a través de la tecnología; evidencias del uso de las herramientas tecnológicas por parte de los estudiantes para recolectar, evaluar o utilizar información; y evidencias de explicación por parte del docente acerca de cómo usar las herramientas digitales fuera del aula.

En segundo lugar, se observaron evidencias de prácticas de enseñanza del pensamiento computacional: organizar y analizar lógicamente los datos; representar datos a través de abstracciones como modelos y simulaciones; automatizar soluciones; identificar, analizar e implementar diferentes soluciones para resolver un problema; y generalizar el proceso de resolución de un problema.

Por último, también se recolectaron evidencias de las prácticas docentes relacionadas a las metodologías de resolución de problemas y al aprendizaje basado en proyectos: trabajo en equipos; asignación de roles; evidencias de utilización de herramientas digitales para investigar, resolver problemas y crear trabajos; evidencias de evocación del conocimiento previo y el pensamiento crítico por parte del docente; completar informes o responder preguntas de análisis; y la promoción del diálogo y el debate entre los estudiantes por parte del docente.

Prácticas construccionistas

Como indica la literatura sobre el construccionismo, para alcanzar altos niveles de efectividad en la enseñanza de la informática y el pensamiento computacional, se esperaría observar la manipulación física de materiales; la enseñanza a través de la tecnología; la utilización de herramientas tecnológicas para recolectar, evaluar o utilizar información para el aprendizaje; y explicaciones acerca de cómo utilizar la tecnología fuera del aula. La Tabla 1 muestra el promedio de observaciones que alcanzó cada nivel de evidencia de la escala en cada criterio observado.

En lo que refiere a la manipulación de materiales, se destaca que en la mitad de las observaciones no fue evidente dicha manipulación, mientras que en el 43% fue muy evidente y

en el 7% fue evidente. Por otro lado, la enseñanza a través de la tecnología, fue el criterio más evidente entre todas las observaciones. En el 71% de las mismas el docente utilizó de manera muy evidente la tecnología para enseñar a los estudiantes mientras que en el 29% restante no lo fue para nada.

La utilización de herramientas tecnológicas para recolectar, evaluar o utilizar información para el aprendizaje de los estudiantes, fue el segundo criterio observado con mayor frecuencia en un alto nivel de evidencia entre las observaciones (64%). En el 7% fue evidente y en el 29% no lo fue. La práctica menos evidente entre las observaciones realizadas, fue la de que el docente explique cómo utilizar la tecnología fuera del aula. Solo en el 29% de los casos fue muy evidente, en el 14% fue evidente, en el 21% fue poco evidente y en el 36% restante no lo fue.

Tabla 1.

Distribución porcentual de observaciones según evidencia de cada criterio observado dentro de las prácticas de enseñanza construccionistas.

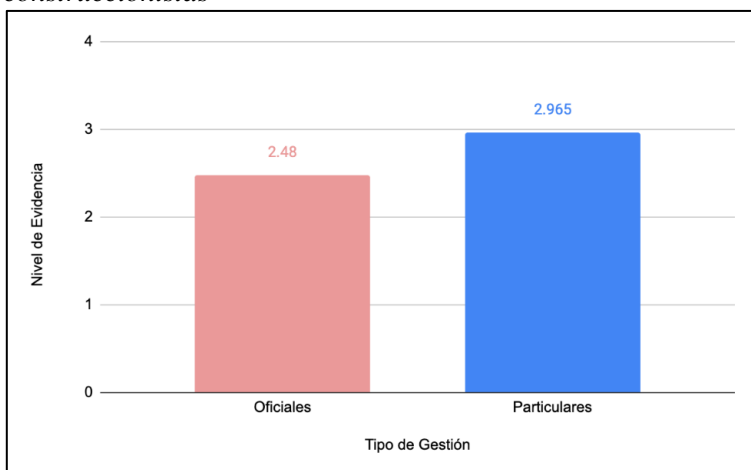
Criterio observado	Muy evidente	Evidente	Poco evidente	No evidente
Manipulación física de materiales	42.9%	7.1%	-	50%
Enseñanza a través de la tecnología	71.4%	-	-	28.6%
Utilización de herramientas tecnológicas para recolectar, evaluar o utilizar información para el aprendizaje	64.3%	7.1%		28.6%
Utilizar la tecnología fuera del aula	28.6%	14.3%	21.4%	35.7%

Fuente: elaboración propia.

Con el objetivo de contrastar la utilización de prácticas construccionistas entre escuelas oficiales y particulares, la Figura 1 muestra el promedio alcanzado por las escuelas de ambos tipos de gestión en la escala de evidencia de 1 al 4. Se observa que, en promedio, las prácticas construccionistas fueron más evidentes en las escuelas particulares que en las oficiales. Los criterios con mayor evidencia a favor de las escuelas particulares fueron que la explicación del docente acerca de cómo utilizar la tecnología fuera del aula y la manipulación física de materiales.

Figura 1

Promedio de evidencia de escuelas oficiales y particulares en la implementación de prácticas constructoristas



Fuente: elaboración propia.

Prácticas de enseñanza del pensamiento computacional

La Tabla 2 muestra el promedio de observaciones que alcanzó cada nivel de evidencia de la escala en cada criterio observado dentro de las prácticas de enseñanza del pensamiento computacional. Organizar y analizar lógicamente los datos fue el criterio más evidente dentro de este conjunto de prácticas. En la mitad de las observaciones esto fue muy evidente y en el 7% evidente. Por otro lado, la representación de datos a través de abstracciones como modelos y simulaciones fue muy evidente únicamente en el 21% de los casos y evidente también en el 21%.

La automatización de soluciones a través del pensamiento algorítmico, es decir, a partir de una secuencia ordenada de pasos, fue el criterio menos evidente. Únicamente en el 7% de las observaciones fue muy evidente. En el 21% fue evidente, en el 29% fue poco evidente y en el 43% restante no fue evidente para nada. Por otro lado, la identificación, el análisis y la implementación de posibles soluciones con el objetivo de lograr la combinación más eficiente y efectiva de pasos y recursos para resolver un problema fue muy evidente en el 14% de las observaciones. En el 43% fue evidente, en el 14% fue poco evidente y en el 29% restante no lo fue para nada.

Por último, en lo que refiere a la generalización del proceso de resolución de problemas fue el segundo criterio menos evidente. Solo en el 7% de las observaciones fue muy evidente y en el 36% fue evidente. Entre las restantes, fue poco evidente en el 29% y para nada evidente en otro 29%.

Tabla 2.

Distribución porcentual de observaciones según evidencia de cada criterio observado dentro de las prácticas de enseñanza del pensamiento computacional

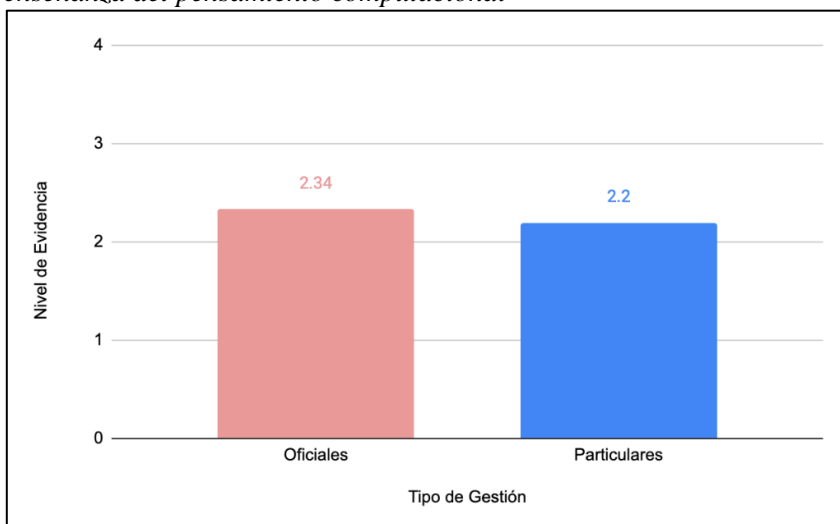
Criterio observado	Muy evidente	Evidente	Poco evidente	No evidente
Organizar y analizar lógicamente los datos por parte de los estudiantes	50%	7.1%	21.4%	21.4%
Representar datos a través de abstracciones	21.4%	21.4%	28.6%	28.6%
Automatizar soluciones a través del pensamiento algorítmico	7.1%	21.4%	28.6%	42.9%
Identificar, analizar e implementar posibles soluciones con el objetivo de lograr la combinación más eficiente y efectiva de pasos y recursos para resolver un problema	14.3%	42.9%	14.3%	28.6%
Generalizar el proceso de resolución de problemas	7.1%	35.7%	28.6%	28.6%

Fuente: elaboración propia.

Al contrastar el promedio de evidencia de escuelas oficiales contra particulares en la implementación de prácticas de enseñanza relacionadas al pensamiento computacional, se observa en la Figura 2 que éstas fueron levemente más evidentes en las escuelas oficiales. El criterio con mayor diferencia a favor de las escuelas oficiales fue la generalización del proceso de resolución de un problema.

Figura 2

Promedio de evidencia de escuelas oficiales y particulares en la implementación de prácticas de enseñanza del pensamiento computacional



Fuente: elaboración propia.

Resolución de problemas y Aprendizaje Basado en Proyectos

Como indica la literatura sobre el aprendizaje por experiencias en el campo de la enseñanza de la informática, se espera observar el trabajo en grupos; la asignación de roles dentro de esos grupos; la utilización de herramientas digitales para investigar, resolver problemas y crear trabajos; la evocación del conocimiento previo y el pensamiento crítico; la elaboración de informes o responder preguntas de análisis; y la promoción del diálogo y el debate entre los estudiantes.

La Tabla 3 muestra que en el 35.7% de las observaciones fue muy evidente que los estudiantes trabajaron en grupos o en pares y en el 7% fue evidente. En otro 7% fue poco evidente y en el 50% restante no lo fue. Por otro lado, la asignación de roles, en tanto práctica fundamental dentro de las metodologías de trabajo en equipo en la resolución de problemas y en aprendizaje basado en proyectos, no fue evidente en la mitad de las observaciones. Únicamente en el 7% fue muy evidente, en el 14% fue evidente y en el 29% fue poco evidente.

La utilización de herramientas digitales con fines de investigación, de resolución de problemas o para crear trabajos originales, fue la práctica más evidente en las observaciones. En el 21% de los casos fue muy evidente y en 36% fue evidente. Por su parte, el docente evocó el conocimiento previo, realizó preguntas o provocó el pensamiento crítico de manera muy evidente únicamente en el 14% de las observaciones. En el 21% fue evidente mientras que en la mitad fue poco evidente y en el resto no fue evidente.

En cuanto a la elaboración de informes o el responder a preguntas de análisis fue el criterio menos evidente dentro de las prácticas relacionadas al aprendizaje por experiencias. Solo en el 7% fue muy evidente y en otro 7% fue evidente. Por último, en lo que refiere a promover el diálogo y el debate entre los estudiantes durante la clase fue muy evidente en el 21% de las observaciones realizadas. En el 14% fue evidente, en el 21% fue poco evidente y en el 43% no lo fue.

Tabla 3.

Distribución porcentual de observaciones según evidencia de cada criterio observado dentro de las prácticas relacionadas al aprendizaje por experiencias

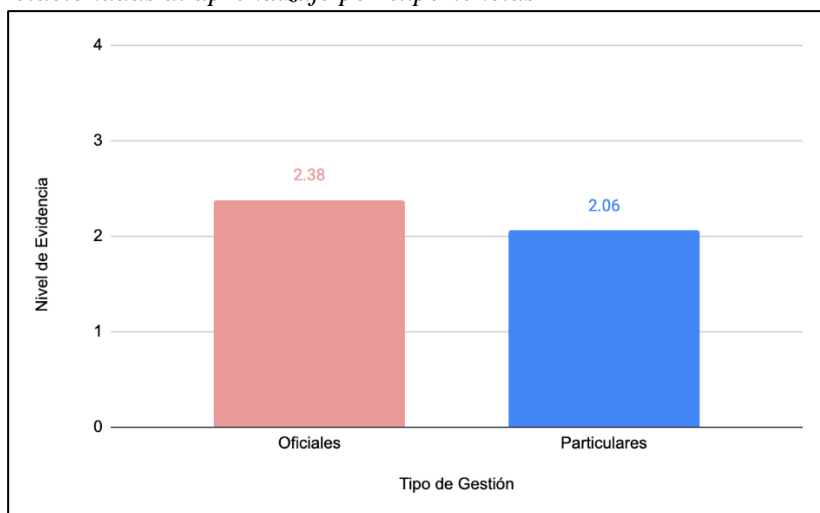
Criterio observado	Muy evidente	Evidente	Poco evidente	No evidente
Trabajo en grupos	35.7%	7.1%	7.1%	50%
Asignación de roles	7.1%	14.3%	28.6%	50%
Utilización de herramientas digitales para investigar, resolver problemas y crear trabajos	21.4%	35.7%	7.1%	35.7%
Evocación del conocimiento previo y el pensamiento crítico	14.3%	21.4%	50%	14.3%
Elaboración de informes o responder preguntas de análisis	7.1%	7.1%	35.7%	50%
Promoción del diálogo y el debate entre los estudiantes	21.4%	14.3%	21.4%	42.9%

Fuente: elaboración propia.

Con el objetivo de contrastar la utilización de prácticas relacionadas al aprendizaje por experiencias entre escuelas oficiales y particulares, la Figura 3 muestra que, en promedio, este tipo de prácticas fueron levemente más evidentes en las escuelas oficiales que en las particulares. El criterio con mayor evidencia a favor de las escuelas oficiales fue el trabajo en grupos. Sin embargo, esto podría deberse, en parte, a la necesidad de compartir computadoras entre los estudiantes. El segundo criterio con mayor evidencia a favor de las escuelas oficiales fue la elaboración de informes o el responder preguntas de análisis.

Figura 3

Promedio de evidencia de escuelas oficiales y particulares en la implementación de prácticas relacionadas al aprendizaje por experiencias



Fuente: elaboración propia.

CONCLUSIONES

En general, en las aulas panameñas observadas se encontró poca evidencia de la utilización de prácticas docentes reconocidas como mejores prácticas para la enseñanza de la informática y el pensamiento computacional. Para la mayoría de los criterios observados en torno a las prácticas de enseñanza construccionistas, las prácticas de enseñanza del pensamiento computacional y las prácticas alineadas a metodologías de resolución de problemas y de aprendizaje basado en proyectos: hubo solo cierta evidencia de su aplicación, con promedios de entre 2 y 2.5 en una escala hasta el 4. Es decir, observamos evidencia apenas mínimamente suficiente del uso de estas prácticas. En general, hubo algo más de evidencia para el uso de prácticas construccionistas (tales como el uso de manipulables), que para el uso de prácticas relacionadas al pensamiento computacional (generación de modelos o automatización de soluciones utilizando algoritmos), y aún menos para el uso de prácticas relacionadas a la resolución de problemas y el aprendizaje basado en proyectos.

En promedio, no hay clara evidencia de diferencias notables entre las escuelas particulares y oficiales en cuanto al uso de estas prácticas, con una mínima ventaja para las escuelas particulares en cuanto a prácticas construccionistas, y de resolución de problemas y aprendizaje basado en proyectos; y una mínima ventaja para las escuelas oficiales en cuanto a prácticas relacionados al pensamiento computacional. Sin embargo, hubo varios criterios de buenas prácticas en los que los resultados fueron polarizados: es decir, las escuelas se encontraban total o prácticamente divididas entre sin evidencia (1) y muy evidente (4), con ningún o casi ningún 2 o 3. Este hallazgo implica un nivel importante de inequidad en el sistema y es particularmente notable en cuanto a la manipulación física de materiales tangibles en clase, las actividades en las que los estudiantes analizan datos, y las actividades en las que los estudiantes automatizan soluciones generando algoritmos con pasos en secuencia.

Adicionalmente, es observable, tanto en la recolección de datos para este estudio, como en la experiencia previa de las autoras y colegas en el campo de la capacitación docente para la enseñanza de la informática, que la realidad en las aulas panameñas de informática podría fortalecerse con mayor utilización del aprendizaje por experiencias (construccionismo, aprendizaje basado en proyectos, y aprendizaje basado en problema) tal como se indica en la

literatura. Es recomendable llevar a cabo proyectos intensivos de capacitación masiva a docentes de materias relacionadas al pensamiento computacional, y particularmente a docentes de informática, para asegurar la maximización de su propio manejo de habilidades de programación, pensamiento analítico y computacional; y fortalecer sus competencias en cuanto al uso de mejores prácticas en las aulas de informática, y técnicas pedagógicas para el desarrollo del pensamiento computacional y analítico.

En conclusión, la enseñanza de la informática en Panamá Metro sufre de considerables carencias en cuanto a prácticas de aula enfocadas en el aprendizaje por experiencia, con colaboración, creación y manipulación. Requiere de mayor énfasis en el desarrollo del pensamiento computacional, lógico-matemático, analítico, y de resolución de problemas. Estos retos tienen un impacto negativo en la formación de recurso humano especializado en el país, así como de ciudadanos con habilidades digitales y de pensamiento lógico. Para un país con aspiraciones a *hub* o centro digital cuya economía depende ampliamente del sector servicio, y que enfrenta retos de participación civil y democrática, es de crítica importancia iniciar mejoras curriculares profundas a las carreras universitarias relevantes para la formación de docentes, y llevar a cabo intervenciones en capacitación docente en métodos de enseñanza efectivos para la informática y el pensamiento computacional. Enfrentar con pasión y compromiso este reto contribuirá a asegurar un mejor futuro para el país y todos sus ciudadanos en la era actual.

REFERENCIAS

- Allen, D.; Donham, R.; Bernhardt, S.; Buskist, W.; Groccia, J. (2011). Problem-based learning. *New Directions for Teaching and Learning*, 128, 21-29
- Baldwin, D., Walker, H. M., & Henderson, P. B. (2013). *The roles of mathematics in computer science. ACM Inroads*, 4(4), 74-80.
- Blumenfeld, P.; Soloway, E.; Marx, R.; Krajcik, J.; Guzdial, M.; Palincsar, A. (1991). *Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning. Educational Psychologist*, 26, 369-398.
- Csizmadia, A., Standl B. y Waite J. (2019). *Integrating the Constructionist Learning Theory*

- with Computational Thinking Classroom Activities. Informatics in Education - An International Journal* 1:41-67.
- Computer Science Teachers Association (2016). *K-12 Computer Science Framework*. Recuperado de <http://www.k12cs.org>
- Computer Science Teachers Association (2017). *CSTA K-12 Computer Science Standards, Revised 2017*. Recuperado de: <http://www.csteachers.org/standards>
- Duch, B. J., Groh, S. E., & Allen, D. E. (2001). *The power of problem-based learning: a practical "how to" for teaching undergraduate courses in any discipline*. Stylus Publishing, LLC.
- Geck, G., Ljulin, A., Peter, S., Schmidt, J., Vehlken, F. & Zeume, T. (2018). Introduction to Iltis: An Interactive, Web-Based System for Teaching Logic. En *Proceedings of 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education, Larnaca, Cyprus, July 2-4, 2018*
- International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA). (2011). *Operational Definition of Computational Thinking for K-12 Education*
- Lee CS., Chan PY. (2019). Mathematics Learning: Perceptions Toward the Design of a Website Based on a Fun Computational Thinking-Based Knowledge Management Framework. Kong SC., Abelson H. (eds) *Computational Thinking Education*. Springer, Singapore
- Mishra, P., & Yadav, A. (2013). Of art and algorithms: Rethinking technology & creativity in the 21st century. *TechTrends*, 57(3), 11.
- Mohagheh, D. M., y McCauley, M. (2016). Computational thinking: The skill set of the 21st century. *International Journal of Computer Science and Information Technologies*, 7, 1524-1530
- Mossakowski, T. (2010). On Teaching Logic and Algebraic Specification. *European Association of Software Science and Technology*. 26
- Nwana, H. S. (1997). *The computer science education crisis: fact or illusion?.* *Interacting with Computers*, 9(1), 27-45.
- Papadakis, Michail; Kalogiannakis y Nicholas Zaranis (2016). *Developing fundamental programming concepts and computational thinking with ScratchJr in preschool education: a case study.* *Int. J. Mobile Learning and Organisation*, 10 (3), 187-202.

- Papert, S. (1993). *Mindstorms: Children, Computers and Powerful Ideas*. Basic Books, Nueva York.
- Pizarro, Rubén (2009). *Las TICs en la enseñanza de las matemáticas*. Universidad Nacional de La Plata, Buenos Aires.
- Resnick, M. (2007). *Todo lo que Realmente Necesito Saber (Acerca del Pensamiento Creativo) Lo Aprendí (Estudiando Cómo Aprenden los niños) en el Kindergarten*. Eduteka, 1-7. Recuperado de: http://eduteka.icesi.edu.co/pdfdir/RESNICK_kindergarten-learning-approach.pdf
- Romeike R., Mareen Przybylla M. (2014) Physical Computing and its Scope – Towards a Constructionist Computer Science Curriculum with Physical Computing. *Informatics in Education - An International Journal* 2:241-254
- Roschelle, J. M., Pea, R. D., Hoadley, C. M., Gordin, D. N., y Means, B. M. (2000). Changing how and what children learn in school with computer-based technologies. *The future of children*, 76-101.
- Ryberg, T. y Georgsen, M. (2010). Enabling Digital Literacy. *Nordic Journal of Digital Literacy*, 5, 88-100.
- Sacristán, V. (2017). *Informatizar las matemáticas, matematizar la informática (una propuesta docente)*. Recuperado de: <http://dccg.upc.edu/people/vera/wp-content/uploads/2012/03/rep.pdf>
- Scolari, C. A. (2013). *Narrativas Transmedia: cuando todos los medios cuentan*. Deusto Ediciones.
- Thomas, J. W. (2000). *A review of research on project-based learning*. The Autodesk Foundation. Disponible en: https://tecfa.unige.ch/proj/eteach-net/Thomas_researchreview_PBL.pdf
- Voogt, J., Fisser, P., Good, J., Mishra, P., & Yadav, A. (2015). Computational thinking in compulsory education: Towards an agenda for research and practice. *Education and Information Technologies*, 20(4), 715-728.
- Wing, J. M. (2008). Computational thinking and thinking about computing. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 366(1881), 3717-3725.
- Wood, D. F. (2003). *Problem based learning*. *BMJ*, 326(7384), 328-330. <https://www.bmj.com/content/326/7384/328>

Zapata-Ros, M. (2015). *Pensamiento computacional: Una nueva alfabetización digital*. *Revista de Educación a Distancia*, (46).

Zhang, L. y Nouri, J. (2019). A systematic review of learning computational thinking through Scratch in K-9. *Computers & Education*. Recuperado de: https://www.researchgate.net/publication/333944299_A_systematic_review_of_learning_computational_thinking_through_Scratch_in_K-9