

Artículo de Investigación



Implementación del Cómputo Físico para enseñar los fundamentos de la programación en contextos laborales

Implementation of Physical Computing to teach the fundamentals of applications programming in work contexts

Juan Riquelme Odi

Universidad Autónoma de Querétaro, México. jriquelme02@alumnos.uaq.mx
<https://orcid.org/0000-0003-0673-9685>

Claudia Marina Vicario Solórzano

Instituto Politécnico Nacional-UPIICSA, México. cvicario@ipn.mx
<https://orcid.org/0000-0003-0144-3607>

Ma. Teresa García Ramírez

Universidad Autónoma de Querétaro, México. teregar@uaq.mx
<https://orcid.org/0000-0002-5524-2002>

DOI: <https://doi.org/10.56162/transdigital26>

Sección: **Artículo de investigación**

Fecha de recepción: **13/05/2020** | Fecha de aceptación: **06/04/2020**

Referencia del artículo en estilo APA 7^a. edición:

Riquelme, J., Vicario, C., & García, M.T. (2020). Implementación del Cómputo Físico para enseñar los fundamentos de la programación en contextos laborales. *Revista Transdigital*, 1(1).
<https://doi.org/10.56162/transdigital26>



Licencia [Creative Commons Attribution 4.0](https://creativecommons.org/licenses/by/4.0/)
International License (CC BY 4.0)

Resumen

La enseñanza de la programación de aplicaciones es una tarea difícil, ya que se necesita comprender diferentes elementos característicos de las Ciencias Computacionales, así como el estudio de la sintaxis y la gramática de los lenguajes de programación. Debido a esta dificultad, existen actualmente diferentes metodologías, técnicas y enfoques para la enseñanza efectiva de los fundamentos de la programación, y uno de estos enfoques es conocido como Cómputo Físico, que utiliza tanto componentes electrónicos (tarjetas programables, circuitos integrados, etc.) como componentes de software (programas, códigos y aplicaciones). Con el Cómputo Físico se han obtenido resultados satisfactorios para la enseñanza de los fundamentos de la programación. Este escrito expone los resultados de la implementación del Cómputo Físico mediante la tarjeta BBC Micro:Bit en un seminario implementado con 16 trabajadores del departamento de Informatización y del departamento de Dirección de Innovación y Tecnologías de la Información de la Universidad Autónoma de Querétaro, en una prueba de concepto enfocada en la consolidación de los fundamentos de la programación mediante el Cómputo Físico. Esta investigación fue de tipo cuantitativo para el análisis de la información a través de cuestionarios pre-test y post-test y mediante un cuestionario con Escalamiento Likert para obtener la percepción de los participantes del seminario.

Palabras clave: BBC Micro:Bit, ciencias computacionales, cómputo físico, fundamentos de programación.

Abstract

Teaching applications programming is a difficult task, since it is necessary to understand different characteristic elements of Computer Sciences, as well as the study of the syntax and grammar of programming languages. Due to this difficulty, there are currently different methodologies, techniques and approaches for the effective teaching of the fundamentals of applications programming, and one of these approaches is known as Physical Computing, which uses both electronic components (programmable cards, integrated circuits, etc.) and software components (programs, codes and applications). With the Physical Computation, satisfactory results have been obtained for teaching the fundamentals of programming. This document presents the results of the implementation of Physical Computing using the BBC Micro:Bit pocket-sized computer in a seminar implemented with 16 workers from the

Computing Department and the Department of Innovation and Information Technologies of the Autonomous University of Querétaro, in a concept test focused on the consolidation of the fundamentals of applications programming, through Physical Computing. This research was of a quantitative type for the analysis of the information through pre-test and post-test questionnaires and through a Likert Scaling questionnaire to obtain the perception of the seminar participants.

Keywords: BBC Micro:Bit; Computer Sciences; Fundamentals of Applications Programming; Physical Computing.

1. Introducción

Algo fundamental que se requiere para facilitar el trabajo en cualquier área, y que se puede considerar una competencia clave en la actualidad es la programación de aplicaciones, pero el aprendizaje efectivo de los fundamentos de la programación se considera una actividad compleja por diferentes factores propios de las Ciencias Computacionales, con situaciones que van desde el aprendizaje adecuado de un lenguaje de programación, así como elementos característicos propios de la programación de aplicaciones. Esta problemática se debe considerar desde un factor computacional como desde una perspectiva tecnológica para la correcta enseñanza de los fundamentos de la programación.

Para apoyar el proceso de enseñanza de los fundamentos de la programación existen actualmente diferentes técnicas, métodos y enfoques, que auxilian en la adquisición de los conocimientos de la programación de aplicaciones y de las Ciencias Computacionales, tanto en el entorno educativo como en contextos profesionales para reducir la dificultad en el aprendizaje de la programación. Uno de estos enfoques es el Cómputo Físico, que busca que los usuarios consigan una interrelación y manipulación de los componentes físicos (hardware) y de los componentes lógicos (software) para lograr una interrelación con el mundo real y no limitarse únicamente a entornos virtuales (Tyncan, 2016).

1.1 La necesidad del aprendizaje de la programación

El aprendizaje de la programación de aplicaciones es una tarea que se considera muy compleja desde su inicio hasta su finalización. Debido a esto existen algunos elementos a considerar sobre la enseñanza de la programación de aplicaciones, y en vista de la naturaleza del aprendizaje de la programación se considera una habilidad abstracta y el producto que

se obtiene es de tipo intelectual, lógico e intangible, ya que se consigue un producto de software o una aplicación.

A causa de la necesidad que actualmente se tiene del aprendizaje de la programación de aplicaciones es necesario identificar que aprender programación no es como seguir una fórmula o una serie de pasos específicos, ni tampoco memorizar diferentes elementos (como la sintaxis y la gramática de un lenguaje de programación), sino que se requiere pensar una forma de solucionar problemas que se plantean para resolverse mediante el uso del equipo de cómputo a través de un lenguaje de programación, y cada persona resuelva el problema desde sus propias competencias (Fuentes-Rosado y Moo-Medina, 2017).

Con respecto a esta problemática, que no solo se presenta en el ámbito académico, sino también en el ámbito profesional, han surgido algunas estrategias para facilitar la enseñanza de la programación, pero tal como especifica Gottfried (1997) en varios entornos de enseñanza de esta disciplina continúan en uso los mismos métodos que hace varios años, sin cambio alguno, a pesar de los avances que han tenido tanto los entornos computacionales como las herramientas para programar, lo que hace más compleja la enseñanza de la programación.

Aun con el dominio de un lenguaje de programación a nivel de sintaxis, gramática y palabras reservadas, una persona no puede considerarse un "experto" en programación pues se requiere además trasladar la problemática a un campo abstracto (la codificación correcta) para dar solución al problema planteado. Para que una persona sea considerada eficiente en la programación de aplicaciones se requieren un conjunto específico de habilidades en el área de la programación de aplicaciones, como son el aprendizaje de un lenguaje de programación, la depuración de un programa generado, crear programas nuevos que resuelvan realmente las problemáticas planteadas, reutilización de código y de componentes y otra serie de elementos característicos de la programación (Isong, 2014). Además, existen algunas otras consideraciones a tener en cuenta al momento de enseñar la programación de aplicaciones:

- Se requieren diferentes habilidades.- Cada individuo programa de forma diferente, y existe divergencia bajo el estilo propio de cada individuo.
- Existen varios procesos.- No solamente basta con memorizar las sentencias de un lenguaje de programación, se requiere generar un algoritmo, trasladarlo a código y posteriormente ejecutar pruebas para depurar el código generado.

- Los lenguajes de programación.- Se busca que los individuos desarrollen más la lógica de la programación que el aprender un lenguaje específico.

Diferentes investigadores han abordado la problemática del aprendizaje de la programación, mediante la implementación de diferentes técnicas, métodos y estrategias, así como con la utilización de herramientas de software novedosas seguidas de procesos pedagógicos adecuados para facilitar la enseñanza de los fundamentos de la programación en diferentes contextos, como lo reflejan las investigaciones de Vera (2017), Luján-Mora y Aragonés (2007) y la de Pérez y Roig-Vila (2015) en donde además de identificar la dificultad de la problemática del aprendizaje de la programación, se implementaron estrategias que iban desde el uso de técnicas didácticas, hasta el uso del entorno Scratch para facilitar en los estudiantes el aprendizaje de los fundamentos de la programación, y la programación de aplicaciones.

Con respecto a la dificultad en la enseñanza y en el aprendizaje de los fundamentos y la programación de aplicaciones han aparecido diferentes medios, técnicas, metodologías, herramientas y enfoques para facilitar esta enseñanza, y uno de estos enfoques es conocido como **Cómputo Físico**, que utiliza medios físicos (electrónicos) y medios lógicos (software) para que las personas puedan comprender mejor a la programación y a las Ciencias Computacionales en general, todo ello basado en un medio físico de interacción (Igoe, 2004).

1.2 El Cómputo Físico

El **Cómputo Físico** es considerado un enfoque pedagógico para la enseñanza de las Ciencias Computacionales y, en este caso, para facilitar el aprendizaje de los fundamentos de la programación, lo que permite que las personas interactúen con el equipo de cómputo a través de un medio físico de comunicación (Igoe, 2004).

Este enfoque tiene un origen a través de Dan O'Sullivan y Tom Igoe, para ayudar a las personas que no tienen conocimientos avanzados de electrónica y así puedan generar proyectos de este tipo. Básicamente consiste en la interconexión de componentes electrónicos de fácil utilización, en conjunto con actuadores y sensores para generar una interrelación entre el mundo físico y el mundo virtual, lo que permite obtener elementos tangibles para que las personas interactúen con ellos, respondan al entorno y se adapten al mismo (Przybylla y Romeike, 2014a).

Desde una perspectiva educativa, el Cómputo Físico utiliza un aprendizaje basado en el Construccionismo donde se busca que las personas se comprometan en la generación de objetos que sean tangibles, que sean expuestos a los demás, y que además puedan evaluarse por las demás personas y con esto obtener un aprendizaje significativo y que exista un compromiso con el proyecto o artefacto que la misma persona crea y construye (Blikstein, 2015).

El Cómputo Físico ha sido utilizado exitosamente en diferentes contextos para la enseñanza de las Ciencias Computacionales y la programación de aplicaciones, tal y como lo sustentan las investigaciones de Schulz y Pinkwart (2015), Przybylla y Romeike (2014b) y la investigación de Rubio, Mañoso y Pérez (2013), quienes demuestran que este enfoque es útil para la enseñanza de la programación y la enseñanza de las Ciencias Computacionales, incluso para introducirlo en la currícula de diferentes asignaturas relacionadas con las Ciencias Computacionales.

El Cómputo Físico ha estado presente en contextos de enseñanza de los fundamentos de la programación y de las Ciencias Computacionales desde hace cierto tiempo, como por ejemplo mediante el uso de LEGO/Logo, el kit LEGO Mindstorms, la tarjeta programable Arduino, la mini-computadora Raspberry Pi y otros componentes electrónicos (Blikstein, 2013). Y actualmente uno de los dispositivos del Cómputo Físico que ha obtenido un gran impacto en la enseñanza de la programación es la tarjeta BBC Micro:Bit, la cual tiene un origen en la iniciativa "Make it Digital" de Reino Unido para motivar a las personas a ser creadores digitales (y no solo consumidores digitales) y desarrollar habilidades propias de ciencias, tecnología, ingeniería y matemáticas (Sentance, Waite, Hodges, MacLeod y Yeomans, 2017).

1.3 La tarjeta programable BBC Micro:Bit

La tarjeta programable BBC Micro:Bit, como dispositivo del Cómputo Físico, tiene su origen en la enseñanza de las Ciencias Computacionales desde los entornos físicos y reales, bajo una concepción de dispositivo electrónico y digital que auxiliara en la enseñanza de la programación a cualquier persona que quisiera aprender la programación de aplicaciones.

Este dispositivo surge con la iniciativa llamada "Make it Digital" en Reino Unido, la cual buscó inspirar a los niños y jóvenes a convertirse en creadores digitales y no solamente consumidores digitales, para el desarrollo de habilidades requeridas actualmente como lo son la ciencia, la tecnologías, la ingeniería y las matemáticas, ya que se había detectado una

disminución de egresados en las escuelas con carreras relacionadas con las Ciencias Computacionales y la programación de aplicaciones (Sentance et al., 2017). Debido a esto se impulsó retomar asignaturas relacionadas con las Ciencias Computacionales de manera obligatoria en las escuelas de Reino Unido, para promover tanto el Pensamiento Computacional como la programación en los estudiantes.

La tarjeta BBC Micro:Bit fue anunciada para su utilización en entornos educativos en julio del 2015, y después se empezó a capacitar a los docentes para que aprendieran a utilizarla e incorporarla en sus asignaturas en este mismo año. Se finalizó este proceso con la entrega de los dispositivos en varias ciudades de Reino Unido (Inglaterra, Irlanda del Norte, Wales y Escocia) en febrero del 2016 con la entrega a profesores y a alumnos de un millón de dispositivos (Ball et al., 2016).

El dispositivo dispone de diferentes características tecnológicas que lo hacen ideal para su utilización en contextos de enseñanza de la programación y de la electrónica, pues cuenta con un display con 25 LEDs, dos botones táctiles, 19 terminales de propósito general, diferentes sensores (acelerómetro, compás, luz, temperatura), y diferentes mecanismos de comunicación con otros dispositivos ya que cuenta con un transmisor/receptor de radiofrecuencia, Bluetooth e interfaz USB (Monk, 2018) (Figura 1).

El dispositivo ha sido utilizado exitosamente en diferentes contextos de enseñanza de las Ciencias Computacionales, como lo reflejan las investigaciones de BBC (2017), Marí (2017), Sentance et al. (2017) y el estudio de Gibson y Bradley (2017), específicamente sobre el uso de la tarjeta en contextos donde se requiere enseñar las Ciencias Computacionales, la programación de aplicaciones y el desarrollo del Pensamiento Computacional. Es por sus características que la tarjeta es ideal para facilitar la enseñanza de los fundamentos de la programación, además del aprendizaje de componentes electrónicos.

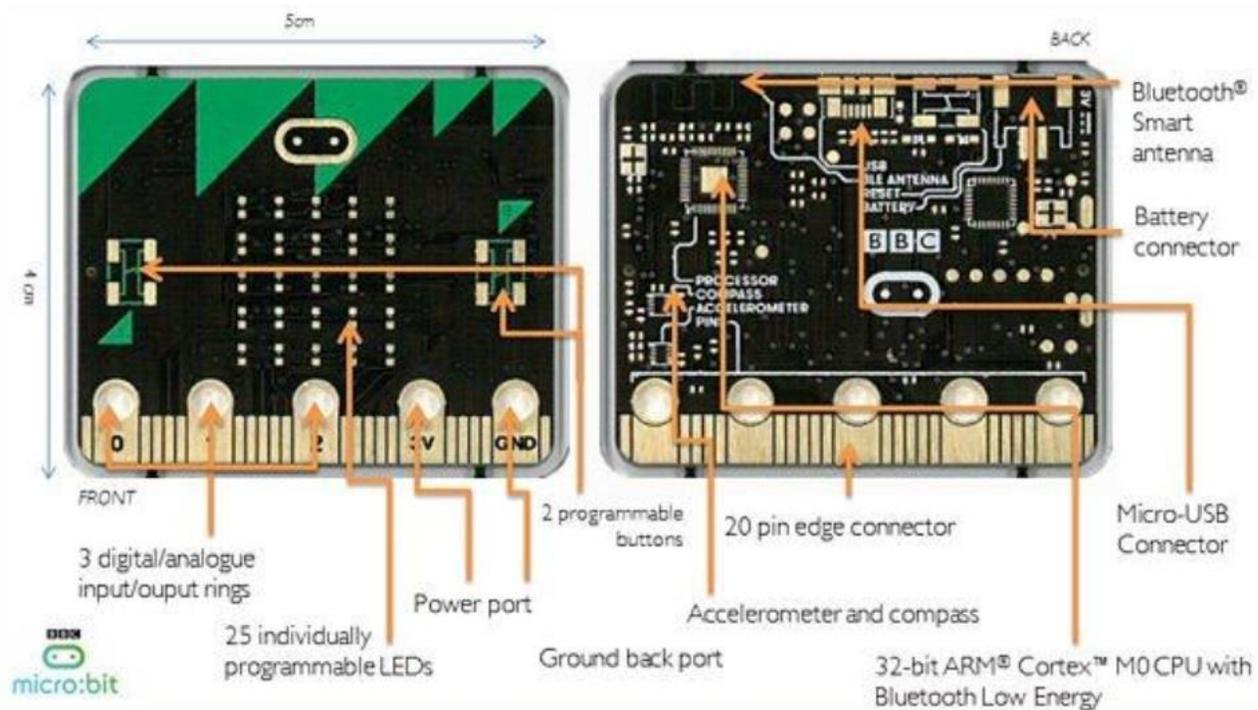


Figura 1.
 La tarjeta programable BBC Micro:Bit.
 Fuente: Ball et al. (2016).

No basta con tener una herramienta tecnológica para facilitar la consolidación de los conocimientos de los fundamentos de la programación, sino también, un sustento pedagógico que avale la necesidad de su enseñanza en entornos laborales y/o contextos donde son requeridos como a través de un seminario impartido a 16 trabajadores del departamento de Dirección de Innovación y Tecnologías de la Información y del departamento de Informatización de la Universidad Autónoma de Querétaro (UAQ).

1.4 Los seminarios para trabajadores de la UAQ

Los diferentes cursos, seminarios y talleres que se imparten al personal administrativo de la UAQ son promovidos por el departamento de Capacitación de Personal de dicha institución para que los trabajadores cuenten con las competencias tecnológicas necesarias para estar a la vanguardia en los conocimientos que le exige su campo laboral.

Riquelme, J., Vicario, C., & García, M.T. (2020). Implementación del Cómputo Físico para enseñar los fundamentos de la programación en contextos laborales. *Revista Transdigital*, 1(1). <https://doi.org/10.56162/transdigital26>

De manera detallada, la Dirección de Recursos Humanos de la Universidad Autónoma de Querétaro tiene como objetivo mejorar la capacitación y el desarrollo del personal que labora en la institución con la misión de desarrollar integralmente al personal, con la visión de desarrollar al personal de manera competitiva y lograr un compromiso permanente en el aprendizaje del personal de la Universidad (Dirección de Recursos Humanos UAQ, 2020a).

Con respecto a la finalidad de la capacitación de los trabajadores la Coordinación de Capacitación de Personal de la Dirección de Recursos Humanos de la UAQ tiene el objetivo de "asegurar la formación del personal con base en necesidades generales y específicas, para el desarrollo de competencias que permitan la mejora de los servicios y procedimientos que desempeñan en la Universidad" (Dirección de Recursos Humanos UAQ, 2020b, p. 1).

La realización de los cursos, talleres y seminarios incluyen conocimientos de diferentes áreas de estudio, como responsabilidad social, manejo de personal, salud laboral y tecnologías de información. Pero en gran medida se requieren conocimientos y habilidades del área de las Ciencias Computacionales, ya que son parte integral en el desarrollo de las actividades de diferentes departamentos que integran a la Universidad Autónoma de Querétaro, además de ser competencias claves requeridas hoy en día.

Debido al objetivo de la Coordinación de Capacitación de Personal, en conjunto con las necesidades específicas de los trabajadores administrativos de la UAQ, se requieren impartir cursos, talleres y seminarios que sean adecuados para el trabajo realizado por los mismos trabajadores, y en consideración a este punto, se aplicó un seminario para la enseñanza de los fundamentos de la programación, ya que tal y como expresan Fuentes-Rosado y Moo-Medina (2017) el aprendizaje de la programación de aplicaciones es una parte integral hoy en día para cualquier profesional que se dedica al área de las Tecnologías de Información y Comunicación.

2. Método de investigación

Esta investigación de tipo cuantitativo tuvo como objetivo identificar si los trabajadores del departamento de Informatización y del departamento de Dirección de Innovación y Tecnologías de Información de la UAQ, consiguieron la consolidación de los conocimientos relacionados con los fundamentos de la programación, mediante la implementación del Cómputo Físico como enfoque educativo con el uso de la tarjeta BBC Micro:Bit. Para poder conseguir este objetivo se generó una estrategia didáctica que buscaba precisamente la

consolidación de los conocimientos de los fundamentos de la programación en los trabajadores de la Universidad Autónoma de Querétaro.

Para la verificación del cumplimiento correcto del objetivo planteado, y con ello demostrar que a través del uso del *Cómputo Físico* se consigue una mejora con respecto a los conocimientos de las Ciencias Computacionales y de los fundamentos de la programación, se aplicó un seminario presencial como mecanismo de implementación de la estrategia didáctica que reafirmara la consolidación de estos conocimientos. Los elementos esenciales del seminario aplicado fueron los siguientes:

El seminario se centró en la consolidación de los conocimientos de las Ciencias Computacionales y los fundamentos de la programación, con el objetivo de identificar si el enfoque del *Cómputo Físico*, mediante el uso de la tarjeta BBC Micro:Bit permitiría a los trabajadores consolidar los conocimientos sobre los fundamentos de la programación. También permitiría obtener la percepción de los participantes sobre si los conocimientos que se incluyeron en el seminario fueron adecuados para poder consolidar dichos conocimientos de los fundamentos de la programación. Algunas características del seminario impartido:

- Contexto y lugar de aplicación: Universidad Autónoma de Querétaro, Campus Centro Universitario, a trabajadores del departamento de Dirección de Innovación y Tecnologías de Información, y trabajadores del departamento de Informatización.
- Duración del seminario: 30 horas, divididas en 10 sesiones de 2 horas y media cada una. Se impartió de lunes a viernes durante 2 semanas de las 2 pm a las 4:30 pm.
- Características de los participantes: 16 participantes. 8 hombres y 8 mujeres. 11 participantes indicaron tener conocimientos previos en programación y 5 participantes no tenían conocimientos en programación. El seminario fue impartido por el mismo investigador.

El seminario impartido en esta prueba de concepto centrada en la consolidación de conocimientos de los fundamentos de la programación contó con diferentes contenidos relacionados con los fundamentos de la programación basados en Norton (2006), Zapata (2013), Pressman (2010) y Aguilar (2008). Además de contenidos relacionados con el *Cómputo Físico* de O'Sullivan e Igoe (2004) y de Platt (2009) y por último, contenidos relacionados con la utilización de la tarjeta BBC Micro:Bit de Halfacree (2018) y de Monk (2018). Las unidades contenidas en el seminario fueron las siguientes:

- Unidad 1: introducción a la BBC Micro:Bit.
- Unidad 2: estructuras de decisión.
- Unidad 3: estructuras de iteración.
- Unidad 4: sensores en la BBC Micro:Bit.
- Unidad 5: introducción a Python con la BBC Micro:Bit.
- Unidad 6: programación orientada a objetos y conexión de dispositivos.
- Unidad 7: radiofrecuencia en la BBC Micro:Bit.
- Unidad 8: proyectos derivados con la BBC Micro:Bit.
- Unidad 9: proyecto final con la BBC Micro:Bit.

En cada unidad del seminario se realizaron actividades prácticas con el uso de la BBC Micro:Bit, propias del contenido teórico de los fundamentos de la programación en cada una de las unidades planteadas, y además en la última unidad los participantes generaron un proyecto, donde se incluían diferentes contenidos teóricos sobre los fundamentos de la programación para la generación de un artefacto propio del Cómputo Físico, el cual fue expuesto entre compañeros para demostrar su funcionamiento y obtener una retroalimentación apropiada.

Cabe mencionar que los contenidos digitales del seminario se desarrollaron con el uso del modelo ADDIE de diseño instruccional mediante el uso de las guías de Arshavskiy (2014) en conjunto con el Sistema Multimodal de Educación a Distancia de la UAQ (Guzmán, Escudero, Ordaz, Chaparro y García, 2016). Los contenidos digitales fueron colocados en una plataforma en línea propia del mismo investigador (Moodle) para que los participantes tuvieran acceso a los mismos vía Internet cada vez que lo requirieran, ya sea para su estudio previo antes de las sesiones del seminario o bien para su estudio posterior por si querían analizar a mayor detalle los mismos. En cada sesión de la plataforma en línea se incluían secciones con contenidos teóricos relacionados con los fundamentos de la programación, componentes electrónicos y actividades propuestas para que los participantes las llevaran a cabo. Todos los asistentes al seminario concluyeron exitosamente el mismo y no hubo deserción, y la retroalimentación de las actividades se daba de forma presencial durante las sesiones del seminario.

En el seminario implementado, se utilizaron instrumentos de recolección de datos para verificar el cumplimiento del objetivo central de esta investigación. Los primeros instrumentos contemplados fueron los cuestionarios pre-test y post-test como medios de verificación sobre el conocimiento de los participantes de la temática abordada antes y después del seminario. Estos cuestionarios fueron construidos a través de elementos

teóricos considerados de Norton (2006), Zapata (2013), Aguilar (2008) y de Rubio et al. (2013) con conocimientos característicos de los fundamentos de la programación como son las estructuras secuenciales, estructuras de decisión, estructuras de iteración, operadores aritméticos y lógicos, variables y constantes y programación orientada a objetos. El cuestionario pre-test se aplicó antes de iniciar el seminario y el cuestionario post-test fue aplicado después de que se hubiera completado la impartición total del seminario con los participantes.

Para obtener la percepción de los participantes sobre el seminario e identificar si los temas abordados fueron adecuados para consolidar sus conocimientos sobre los fundamentos de la programación, se utilizó un cuestionario con Escalamiento Likert construido a través de reactivos del instrumento de Rubio et al. (2013). El cuestionario disponía de escalas que iban desde el valor de 1 (definitivamente no) hasta el valor de 5 (definitivamente sí) con categorías de conceptos que incluyeron elementos de los fundamentos de la programación, entornos de programación, componentes electrónicos y la misma percepción del seminario. Este cuestionario fue aplicado al finalizar la impartición total del seminario con los mismos participantes.

3. Resultados y discusión

El cuestionario pre-test y el cuestionario post-test fueron aplicados a los 16 participantes del seminario, y se obtuvo un puntaje de 51.06 puntos en el pre-test y un puntaje de 71.95 puntos en el post-test, mediante el uso de una escala de 0 a 100 puntos. Con esto se identifica que hubo un incremento de 20.88 puntos con respecto a los contenidos evaluados en el seminario, lo que demuestra que para este caso en particular la estrategia implementada fue efectiva para consolidar los conocimientos de los fundamentos de la programación por el incremento de los resultados que se obtuvieron. En la Figura 2 se muestra la diferencia de promedios entre el pre-test y el post-test.

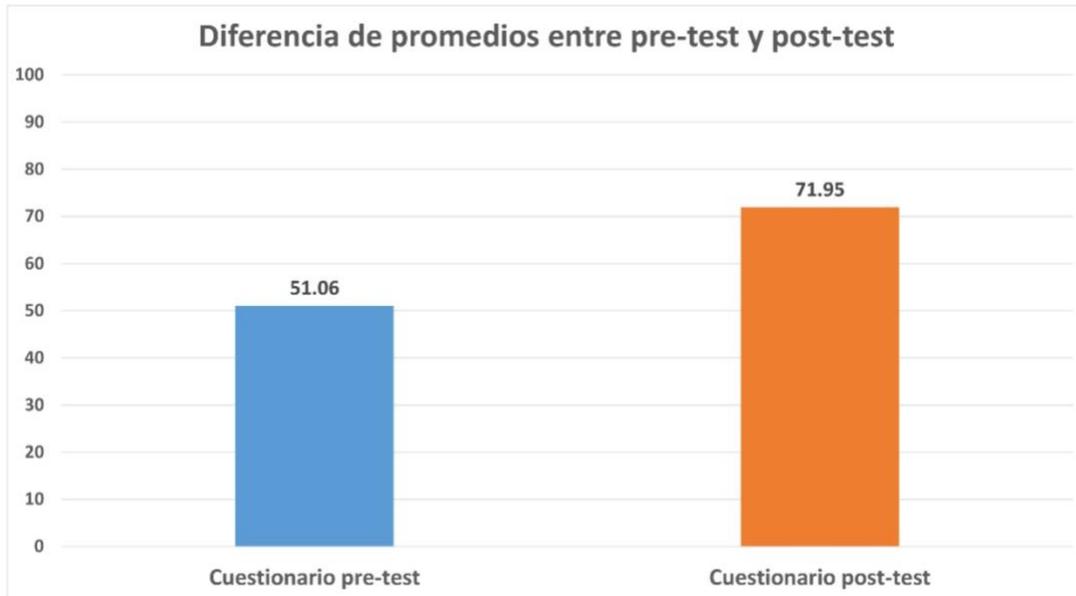


Figura 2.
Diferencia de promedios entre el pre-test y el post-test.
 Fuente: Elaboración propia.

Se muestra a continuación los puntajes por categoría con su respectiva diferencia entre el pre-test y el post-test en relación a temas específicos considerados en ambos instrumentos en la Tabla 1.

Tabla 1.
Diferencia de promedios por categoría del pre-test y el post-test.

Categoría de los instrumentos	Puntaje del pre-test	Puntaje del post-test	Diferencia
A. Estructuras secuenciales	79.68 puntos	79.68 puntos	0 puntos
B. Estructuras de decisión	53.75 puntos	61.25 puntos	+7.5 puntos
C. Estructuras de iteración	26.56 puntos	51.56 puntos	+25 puntos
D. Tipos de datos	64.28 puntos	85.71 puntos	+21.42 puntos
E. Variables y constantes	42.18 puntos	76.56 puntos	+34.37 puntos

F. Operadores en programación	50 puntos	62.5 puntos	+12.5 puntos
G. Prog. orientada a objetos	46.87 puntos	75 puntos	+28.12 puntos
H. Componentes electrónicos	42.96 puntos	75 puntos	+32.03 puntos

Fuente: Elaboración propia.

Para las 8 categorías contenidas en el instrumento hubo 7 que muestran un incremento en el conocimiento consolidado en los participantes del seminario, y cinco categorías muestran incrementos significativos superiores al 20% con respecto a los resultados obtenidos entre el cuestionario pre-test y post-test.

La categoría de "Estructuras de iteración" manifestó un incremento del 25%, la categoría de "Tipos de datos" tuvo un incremento del 21.42%, la categoría de "Variables y constantes" con un incremento de 34.37%, la categoría de "Programación orientada a objetos" obtuvo un incremento del 28.12% y la categoría de "Componentes electrónicos" manifestó un incremento del 32.03%. Con los resultados obtenidos se obtuvieron incrementos significativos con respecto a los conocimientos de las Ciencias Computacionales y los fundamentos de la programación, y con esto se afirma que para este caso en concreto el Cómputo Físico resulta un enfoque adecuado para la enseñanza de la programación de aplicaciones, lo que permite la comprensión de conocimientos relacionados con los fundamentos de la programación.

En la Figura 3 se puede ver a algunos de los participantes en la exposición de sus proyectos finales realizados al concluir el seminario, con el uso de la tarjeta BBC Micro:Bit.



Figura 3.

Participantes del seminario en la exposición de sus proyectos.

Fuente: Elaboración propia.

Tanto el cuestionario pre-test y el cuestionario pos-test fueron validados con respecto a su confiabilidad mediante la aplicación del Alfa de Cronbach donde se obtuvo un valor de 0.9127, además de aplicar la prueba KR-20 donde se obtuvo un valor de 0.9191 lo que demuestra con esto la confiabilidad de los instrumentos. Para llevar a cabo estas pruebas estadísticas de confiabilidad del instrumento se utilizó el software Minitab.

Como medio de afirmar que hubo un incremento en el conocimiento de los participantes con respecto al seminario se aplicó la prueba T de Student, precisamente para verificar la consolidación de los conocimientos de los fundamentos de programación. La aplicación de esta prueba consideró un intervalo de confianza del 95% y se obtuvo un valor de 0.00029 lo cual verifica el planteamiento de que sí hubo un incremento en el conocimiento de los participantes del seminario.

El cuestionario con Escalamiento Likert se aplicó al finalizar el seminario implementado con los trabajadores. El mismo se aplicó a los 16 participantes, y utilizó una escala que iba del número 5 (definitivamente sí) hasta el valor de 1 (definitivamente no). La finalidad de este instrumento fue el analizar la percepción de los participantes del seminario con respecto a los contenidos incluidos de los fundamentos de la programación, los contenidos del seminario, las actividades implementadas y la tarjeta BBC Micro:Bit. En la Figura 4 se muestran los promedios obtenidos en el cuestionario con Escalamiento Likert.

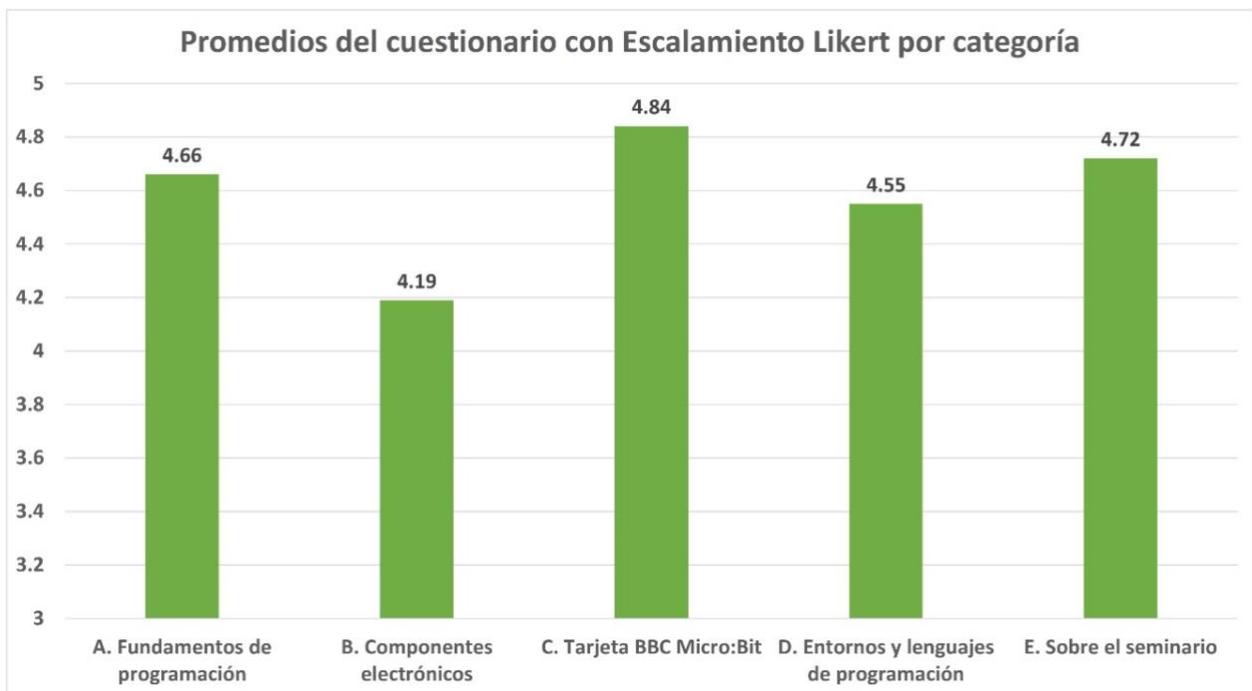


Figura 4.
Promedios del cuestionario Likert por categoría.
Fuente: Elaboración propia

En los resultados obtenidos se puede ver cómo todas las categorías del instrumento obtuvieron promedios superiores a los 4 puntos (probablemente sí) y que las categorías de "Fundamentos de programación", "Tarjeta BBC Micro:Bit", "Entornos de desarrollo y lenguajes de programación" y la categoría de "Sobre el seminario" obtuvieron promedios superiores a

los 4.5 puntos. Los resultados obtenidos demuestran que desde la perspectiva de los participantes el seminario tuvo una percepción positiva para casi todas las categorías del instrumento.

Al considerar la percepción de los participantes en la categoría "A. Fundamentos de programación", la cual incluía reactivos que evaluaban la percepción de los estudiantes sobre los fundamentos de la programación incluidos en el seminario, esta categoría obtuvo los siguientes puntajes para cada reactivo específico como se muestra en la Tabla 2.

Tabla 2.

Promedios por reactivo de la categoría "A. Fundamentos de programación" del Escalamiento Likert.

Categoría del Cuestionario Likert por reactivo	Promedio obtenido
a. Estructuras secuenciales	4.75 puntos
b. Estructuras de decisión	4.62 puntos
c. Estructuras iterativas	4.31 puntos
d. Variables y constantes	4.68 puntos
e. Tipos de datos en programación	4.87 puntos
f. Operadores aritméticos	4.93 puntos
g. Operadores relacionales	4.81 puntos
h. Operadores booleanos	4.56 puntos
i. Programación orientada a objetos	4.43 puntos

Fuente: Elaboración propia.

En la tabla anterior, se puede ver cómo todos los reactivos tuvieron percepciones con un promedio superior a los 4 puntos en el cuestionario con Escalamiento Likert, y los reactivos de "a. Estructuras secuenciales", "b. Estructuras de decisión", "d. Variables y constantes", "e. Tipos de datos en programación", "f. Operadores aritméticos", "g. Operadores relacionales" y "h. Operadores booleanos" obtuvieron promedios superiores a los 4.5 puntos, con lo que se puede especificar que, para este caso en concreto de aplicación del seminario, los participantes consideraron adecuados los contenidos enseñados sobre los fundamentos

de la programación para sus necesidades en el ámbito laboral, y para el desarrollo de aplicaciones mediante la programación.

El cuestionario con Escalamiento Likert fue validado con respecto a su confiabilidad mediante la prueba Alfa de Cronbach, donde se obtuvo un valor de 0.9462 a través el software Minitab para realizar este estadístico. Con esto queda demostrado que el instrumento fue confiable en su construcción y en su misma aplicación.

La implementación del seminario para la prueba de concepto resultó ser una estrategia adecuada para consolidar los conocimientos de los fundamentos de la programación lo cual se puede ver en los puntajes obtenidos en cada uno de los instrumentos aplicados en esta investigación. Por lo tanto, para la prueba de concepto aplicada para el contexto específico donde se llevó a cabo la prueba mediante la culminación de un seminario, sí hubo un incremento específico con respecto al conocimiento de los fundamentos de la programación en diferentes categorías consideradas como necesarias para la programación de aplicaciones.

4. Conclusiones

En los resultados mostrados se observa que el Cómputo Físico es un enfoque adecuado para la enseñanza de los fundamentos de la programación, pues permite consolidar estos conocimientos mediante su utilización, lo cual queda demostrado a partir de los resultados que se obtuvieron en los diferentes instrumentos considerados en la parte metodológica de esta investigación.

Con respecto al promedio general obtenido en esta prueba de concepto, se obtuvo un incremento significativo en este promedio en general, y sobre cada una de las categorías consideradas para su evaluación, lo que muestra que hubo diferencias significativas para casi todas las categorías consideradas en el cuestionario pre-test y el cuestionario post-test, y que solamente una única categoría la de "Estructuras secuenciales" mantuvo la misma puntuación tanto en el pre-test como en el post-test, pero que las categorías de "Estructuras de iteración", "Variables y constantes", Programación orientada a objetos" y la categoría de "Componentes electrónicos" tuvieron un incremento significativo en la comparación entre ambos instrumentos.

La estrategia implementada incluyó al Cómputo Físico como enfoque didáctico, la impartición del seminario para la enseñanza de los fundamentos de la programación, y el uso de la tarjeta BBC Micro:Bit, resultó ser adecuada para la enseñanza de conocimientos relacionados con las Ciencias Computacionales y los fundamentos de la programación, ya que permitió a los trabajadores de la UAQ consolidar conceptos y teorías de la programación de aplicaciones que pueden ser de difícil comprensión, lo que permite además enseñar de una forma amigable los conceptos de los fundamentos de la programación.

En el seminario impartido se obtuvieron tendencias positivas para todas las categorías del cuestionario con Escalamiento Likert con lo cual se especifica que, para el seminario impartido con respecto a los conocimientos incluidos de los fundamentos de la programación, los componentes electrónicos elegidos, el uso de la tarjeta BBC Micro:Bit, los entornos y lenguajes de programación así como el mismo seminario, resultaron ser una manera correcta y adecuada para la enseñanza de los fundamentos de la programación desde la perspectiva de los participantes en el seminario.

Debido a la puntuación obtenida a nivel de conocimientos, lo cual quedó expresado en el incremento de los puntajes entre el pre-test y el post-test, así como a través de la percepción de los participantes con el uso del cuestionario con Escalamiento Likert, y en la realización efectiva de todas y cada una de las prácticas consideradas durante la estrategia además del proyecto final generado, se considera que la estrategia implementada a través del seminario impartido en esta prueba de concepto, mediante el uso del Cómputo Físico y con la tarjeta BBC Micro:Bit, permitieron consolidar los conocimientos relacionados con los fundamentos de la programación, tanto a nivel de conocimientos y desde la misma perspectiva de los trabajadores de la UAQ.

6. Agradecimientos.

Se agradece al Consejo Nacional de Ciencia y Tecnología en México (CONACYT) por el apoyo brindado para llevar a término esta investigación, así como por la adquisición de las tarjetas utilizadas durante el seminario. También se agradece el apoyo financiero de la Red LaTE México, con número de proyecto 294219 de este mismo Consejo con clave SOP-2018-RE / 048 del Instituto Politécnico Nacional. Por último, se agradece a la Dirección de Recursos Humanos de la Universidad Autónoma de Querétaro, por las facilidades prestadas para la aplicación del seminario.

Referencias

Aguilar, L. (2008). *Fundamentos de programación: algoritmos, estructuras de datos y objetos*. España: McGraw-Hill/Interamericana.

Arshavskiy, M. (2014). *Diseño Instruccional para Aprendizaje en Línea Guía Esencial para la Creación de Cursos Exitosos de Educación en Línea*. Charleston, Carolina del Sur: Createspace Independent Pub.

Ball, T., Protzenko, J., Bishop, J., Moskal, M., de Halleux, J., Braun, M., Hodges, S. y Riley, C. (2016). Microsoft touch develop and the BBC micro:bit. *Proceedings of the 38th International Conference on Software Engineering Companion - ICSE '16*, 637–640. <https://doi.org/10.1145/2889160.2889179>

BBC (2017). BBC micro:bit celebrates huge impact in first year, with 90% of students saying it helped show that anyone can code. Recuperado el 05 de marzo de 2020, de <http://www.bbc.co.uk/mediacentre/latestnews/2017/microbit-first-year>

Blikstein, P. (2013). Gears of our childhood: constructionist toolkits, robotics, and physical computing, past and future. *IDC '13 Proceedings of the 12th International Conference on Interaction Design and Children*, 173–182. <https://doi.org/10.1145/2485760.2485786>

Blikstein, P. (2015). Computationally Enhanced Toolkits for Children: Historical Review and a Framework for Future Design. *Foundations and Trends in Human– Computer Interaction*, 9(1), 1–68. <https://doi.org/10.1561/11000000057>

Dirección de Recursos Humanos UAQ (2020a). Quiénes somos. Recuperado el 7 de marzo de 2020, de <https://drh.uaq.mx/index.php/conocenos/quienes-somos>

Dirección de Recursos Humanos UAQ (2020b). Coordinación de Capacitación de Personal. Recuperado el 9 de marzo de 2020, de <https://drh.uaq.mx/index.php/capacitacion-de-personal>

Fuentes-Rosado, J. I., y Moo-Medina, M. (2017). Dificultades de aprender a programar. *Revista Educación en Ingeniería*, 12(24), 76-82. <https://doi.org/10.26507/rei.v12n24.728>

Riquelme, J., Vicario, C., & García, M.T. (2020). Implementación del Cómputo Físico para enseñar los fundamentos de la programación en contextos laborales. *Revista Transdigital*, 1(1). <https://doi.org/10.56162/transdigital26>

- Gibson, S., y Bradley, P. (2017). A Study of Northern Ireland Key Stage 2 Pupils' Perceptions of Using the BBC Micro:Bit in Stem Education. *The STeP Journal*, 4(1), 15–41.
- Gottfried, B. S. (1997). Teaching Computer Programming Effectively Using Active Learning. *Age*, 2(1), 1-8.
- Guzmán, T., Escudero-Nahón, A., Ordaz, T., Chaparro, R. y García, T. (2016). *Sistema Multimodal de Educación. Principios y lineamientos de la educación a distancia, abierta y mixta de la Universidad Autónoma de Querétaro*. Recuperado el 02 de febrero de 2019, de <https://www.uaq.mx/docsgrales/informatica/Sistema-Multimodal-de-educacion-UAQ.pdf>
- Halfacree, G. (2018). *The official BBC Micro:bit user guide*. Indianapolis, Ind: John Wiley and Sons, Inc.
- Igoe, T. (2004). What Is Physical Computing? Recuperado el 07 de marzo de 2020, de <http://www.tigoe.com/blog/what-is-physical-computing/>
- Ison, B. (2014). A Methodology for Teaching Computer Programming: first year students' perspective. *International Journal of Modern Education and Computer Science*, 6(9), 15–21. <https://doi.org/10.5815/ijmeecs.2014.09.03>
- Luján-Mora, S., y Aragonés Ferrero, J. (2007). Técnicas didácticas novedosas en la enseñanza de programación: el caso de " Programación en Internet. *Actas del XV Congreso Iberoamericano de Educación Superior en Computación*, 1-9.
- Marí, J. J. (2017). *BBC Micro:Bit. Introducción a la mecatrónica en estudios preuniversitarios*. Universidad Politécnica de Valencia.
- Monk, S. (2018). *Programming the BBC micro:bit Getting Started with Micropython*. (M. G. Hill, Ed.). Nueva York: Mc Graw Hill Education.
- Norton, P. (2006). *Introducción a la computación* (6a ed.). México, D.F.: Mc Graw Hill.
- O'Sullivan, D., e Igoe, T. (2004). *Physical Computing: sensing and controlling the physical world with computers*. (Thomson, Ed.). Boston: Thomson.

- Pérez, H. O., y Roig-Vila, R. (2015). Entornos de programación no mediados simbólicamente para el desarrollo del pensamiento computacional. Una experiencia en la formación de profesores de Informática de la Universidad Central del Ecuador. *Revista de Educación a Distancia*, (46), 1-22.
- Platt, C. (2009). *Make: Electronics: Learning Through Discovery*. California: O'Reilly.
- Pressman, R. (2010). *Ingeniería del software: un enfoque práctico* (7a ed.). México: McGraw-Hill.
- Przybylla, M., y Romeike, R. (2014a). Key Competences with Physical Computing. *Proceedings of Key Competencies in Informatics and ICT 2014*, 351–361.
- Przybylla, M., y Romeike, R. (2014b). Physical Computing and Its Scope—Towards a Constructionist Computer Science Curriculum with Physical Computing. *Informatics in Education*, 13(2), 241–254. <https://doi.org/10.15388/infedu.2014.05>
- Rubio, M. A., Mañoso, C., y Pérez, Á. (2013). Using arduino to enhance computer programming courses in science and engineering. *Proceedings of EDULEARN13 conference*, 5127-5133.
- Schulz, S., y Pinkwart, N. (2015). Physical computing in stem education. *Proceedings of the Workshop in Primary and Secondary Computing Education*, 134-135.
- Sentance, S., Waite, J., Hodges, S., MacLeod, E., y Yeomans, L. (2017). “Creating Cool Stuff.” *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education - SIGCSE '17*, 531–536. <https://doi.org/10.1145/3017680.3017749>
- Tyncan (2016). Physical Computing. Recuperado el 07 de marzo de 2020, de <http://www.tyncan.com/physical-computing>
- Vera, E. E. (2017). *Estrategias de aprendizaje autónomo para disminuir los índices de reprobación en la materia de Metodología de la Programación en la FCC de la BUAP en un sistema de E-Learning*. Centro de Estudios Superiores en Educación.
- Zapata, C. A. (2013). *Fundamentos de programación, guía de autoenseñanza*. Colombia: RA-MA Editorial.