



PROGRAMAR COMPUTADORAS EN EDUCACIÓN INFANTIL PROGRAMMING COMPUTERS IN EARLY CHILDHOOD EDUCATION

María Cecilia Martínez; cecimart@gmail.com
Marcos Javier Gómez; markitosja243@gmail.com

Universidad Nacional de Córdoba y CONICET

RESUMEN

Este artículo analiza una experiencia de enseñanza de la programación con robots en un centro de educación infantil. Basándonos en observaciones, trabajos prácticos de los alumnos y grupos focales; identificamos que los niños logran construir ideas complejas sobre qué es una computadora, qué es programar y qué hace un programador. La programación de robots con lenguaje icónico también permite desarrollar conceptos computacionales como secuencia, parámetro, ciclo y condicional. Los niños de 5 años establecen correspondencia entre el código programado por ellos y el desplazamiento de los robots, apropiándose de la noción de computadora como máquina programable. Los pequeños de 3 y 4 años no logran comprender esta idea.

Palabras clave: enseñanza, pensamiento computacional, programación, educación infantil, tecnología, robótica

ABSTRACT

This article analyzes a robot programming teaching experience in an Early Childhood Education center. Based on classroom observations, students practice sheets, and focus group responses; we identified that children develop complex ideas about what a computer is, what programming is, and what programmers do. Robot programming using an iconic language also allows developing computational concepts such as sequence, parameter, loops and conditional. 5-year-old children established correspondence between the code that they programmed and the robot movements, learning the concept of computers as a programmable machine. The 3 to 4-year-old children do not master this idea.

Keywords: teaching, computational thinking, programming, early childhood, technology, robotics

Introducción

Muchos países están debatiendo sobre si las Ciencias de las Computación (CC) deben ser incluidas al contenido curricular obligatorio del sistema educativo (Bell, 2014). Quienes argumentan que este cambio curricular es necesario, sostienen que la enseñanza de las CC, específicamente a través de la programación, contribuiría a cerrar la brecha digital, mejorar la fluidez tecnológica entre la ciudadanía (Benitez Larghi, 2014), y desarrollar el pensamiento computacional (Zapata Ross, 2015). En este sentido, algunas investigaciones realizadas durante la década de los 90 han mostrado cómo aprender a programar fortalece el pensamiento lógico y matemático y por tanto sería adecuado enseñarlo en las escuelas (Clements, 2002). A pesar de estas experiencias, en los últimos 20 años las propuestas educativas se han centrado en integrar el uso de tecnologías programadas en los espacios curriculares existentes para potenciar los aprendizajes digitales y disciplinares básicos de la escuela. Este último enfoque, denominado integrador, define a la computación como una herramienta que permite representar la información de distintas maneras (Levis, 2007). Desde el punto de vista de las CC, la alfabetización computacional en cambio, no se reduce a usar herramientas sino a dominar conceptos y lenguajes que permitirían programar a esas herramientas.

Estos últimos años hemos visto varias propuestas para enseñar a programar en la escuela. Esta introducción genera preguntas sobre las consecuencias cognitivas que resultan de la exposición a las computadoras en la educación infantil. Un informe realizado para el caso de Estados Unidos indica que el 40% de los niños de entre 0 y 4 años han usado algún tipo de medios digitales, tales como teléfonos móviles, computadoras, o tabletas (Wartella, 2013). En este mismo estudio se identificó que el 30% de los padres de niños que transitan la educación infantil han “bajado” o instalado en sus computadoras hogareñas algún tipo de aplicación o juego para sus niños. Esta tendencia es creciente a lo largo del tiempo. El uso de celulares de niños de 0 a 8 años pasó de 75% en 2014 a 98% en 2017, y de 15 minutos por día en 2014 a 45 minutos por día en 2017 (Rideout, 2017). Estos datos sugieren que los niños están accediendo a medios digitales a temprana edad, más allá de que el sistema educativo decida o no introducir las computadoras en las aulas.

Este consumo masivo y temprano de juguetes computarizados puede contribuir a generar ideas erradas sobre las máquinas programables y su potencial para la creación de tecnología. En efecto, el 60% de los usos familiares de tecnología digital está destinado a aplicaciones de juegos y solo el 20% a aplicaciones educativas. Las aplicaciones educativas no abordan contenido de las CC sino de otras áreas disciplinares tales como la literatura o matemática (Genc, 2014). Generalmente, los juegos tienen complejos diseños que se adaptan a las habilidades de los niños, y en muchos casos limitan estas habilidades a identificar objetos o patrones. Los juguetes programados no ofrecen posibilidades de crear nuevas acciones, movimientos o usos alternativos. Así, los niños aprenden tempranamente a ser usuarios de juguetes pre-programados.

Lejos del uso mecánico de herramientas, las CC consisten -en parte- en resolver problemas o desafíos de automatización de tareas a través de programas que una máquina pueda ejecutar. El tipo de pensamiento que subyace al proceso de pensar problemas que puedan ser resueltos por una máquina se ha designado como “pensamiento computacional” (Román, 2015). Introducir las CC a través de la programación en la escuela obligatoria genera discusiones sobre el currículum y en particular sobre qué contenidos de programación son considerados básicos y alcanzables para los diferentes grupos etarios. Algunos países como Nueva Zelanda e Israel ofrecen contenidos de programación solamente en la escuela secundaria. El Reino Unido ha introducido la programación en la escuela primaria y Estonia lo hace desde la educación infantil (Bell, 2014). Un debate sobre los contenidos de CC ofrecidos en la escuela permitiría establecer qué nivel dentro del sistema educativo sería adecuado para introducir la programación.

Desde el punto de vista del planeamiento educativo toda modificación en el currículum es una expresión del Estado para cambiar la formación básica de sus ciudadanos. Sin embargo, es difícil tomar decisiones curriculares para el nivel de la educación infantil porque no conocemos los efectos y beneficios de aprender a programar en estos contextos. En particular, poco sabemos de cómo se relacionan el aprendizaje de la programación y la fluidez digital; y el conocimiento sobre la disciplina y el desarrollo del pensamiento computacional. Con el propósito de estudiar cómo niños de 3 a 5 años aprenden conceptos y procedimientos básicos de las CC a través de la programación y desarrollan sus creencias sobre qué hace un especialista en computación, diseñamos un estudio exploratorio en un centro de gestión privada de Córdoba, Argentina. Este artículo busca documentar el valor del aprendizaje de la programación para comprender el mundo digital nos rodea.

Referencias Teóricas

Algunos investigadores argumentan que aquellos niños que usan software educativo para aprender a programar pueden desarrollar operaciones simbólicas (Clements, 2002), y que este desarrollo traer beneficios a otras áreas de conocimiento que son esencialmente simbólicas tales como matemáticas y lenguas. De esta manera, el aprendizaje de la programación podría ser beneficioso a largo plazo más allá del efecto en educación infantil (Reynolds, 2011). En su trabajo, Clements (2002) resume una serie de estudios de los años 80 mostrando que la programación de computadoras promueve la clasificación y el pensamiento lógico. También argumenta que los estudiantes que usan Logo¹ y otros sistemas de programación tangible a través objetos físicos como cubos, pueden proponer diferentes teorías sobre cómo resolver un problema y tienen la habilidad de “depurar” (eliminar errores de programación) un programa. Esto significaría que los niños realizan un monitoreo de su aprendizaje, puesto que revisan los pasos de su programa buscando los errores; indicando el

1 Logo es un sistema informático para enseñar a programar utilizado desde la década de los 80 y desarrollado por S. Papert, quién buscaba enseñar matemática de manera constructivista en el nivel primario.

desarrollo de procesos metacognitivos. Clements (2002) concluye que en particular el programa Logo, logra vincular el saber intuitivo de los niños (sobre movimientos y arrastre) con ideas más formales y explícitas (como nociones matemáticas), promoviendo una positiva auto percepción sobre su propio potencial para crear productos tecnológicos. Es decir, a partir de programar, los niños se empiezan a pensar como creadores de tecnología. Esta línea argumentativa sugiere un gran potencial en introducir la programación en la educación infantil dado que muchos niños que provienen de familias empobrecidas, cultural o económicamente, no han desarrollado competencias básicas de pensamiento lógico al llegar al primer grado (Pascoe, Wood, Dufee, Kuo, 2016).

Algunas investigaciones previas sobre la enseñanza de conceptos básicos de la CC en niños de educación infantil se han centrado en la implementación de programas curriculares de CC en contextos reales, es decir en centros escolares. En general, este tipo de investigaciones ha tenido un diseño exploratorio y en algunas circunstancias, ha seguido un enfoque de la investigación-acción para documentar lo posible, es decir cómo sería enseñar computación en educación infantil (Gergen, 2003). Cuando los docentes introducen e integran conceptos de CC en el currículum, los niños pueden desarrollar estos saberes en todo su potencial (Clements, 2002). Leonel, Morgado y Kahn (2010), por ejemplo, han documentado cómo niños de educación infantil aprenden conceptos tales como sintaxis, parámetros, procedimientos complejos, paralelismo, concurrencia, canales de comunicación, input y servidores llevando a cabo experiencias de enseñanza de la programación con la plataforma Toon Talk. Estos autores realizaron la experiencia primero con un grupo reducido de niños y luego con un grupo más amplio que involucraba a maestras en contextos de clases escolares. Como resultado de su estudio, sugieren que es posible usar diferentes estrategias de enseñanza para abordar los conceptos mencionados. Una de las fortalezas de este estudio fue trabajar en contextos escolares reales en contraposición a intervenciones controladas de laboratorio, lo que permitió mostrar que los docentes necesitaban apoyo en la enseñanza de estos contenidos. Una buena práctica de apoyo fue crear un “Recetario” de tareas de programación que los docentes puedan utilizar.

La enseñanza de la programación en la educación infantil promueve la iniciación a las reglas de sintaxis y el desarrollo en habilidades motoras finas (Bers, Flannery, Kazacoff, 2014). Para desarrollar estas habilidades en los niños, algunos investigadores han desarrollado plataformas tangibles. En esta línea la programación de robots se presenta como una manera de ayudar a conectar a los alumnos al mundo virtual de las CC con el mundo real de las acciones concretas de los robots. Inclusive, niños que no saben leer y escribir y están en diferentes estadios del desarrollo, pueden programar con plataformas tangibles como bloques de madera o bloques virtuales (Flannery, 2013; Kazakoff, 2013). Los investigadores citados mostraron que los niños podían programar robots tanto en los contextos de su clase, como en intervenciones más controladas. Asimismo, Flannery y sus colegas encontraron relaciones significativas entre la programación de robots con plataformas tangibles y las habilidades de secuenciación. El 60% de los niños de 4 a 6 años pudieron programar un robot para hacer movimientos de una danza popular en Estados Unidos (La danza del “Hokey Pokey”, un baile que consiste en mover partes del cuerpo y agregar

recursivamente partes del cuerpo en cada iteración). Aquellos niños que no pudieron resolver el desafío del Hokey Pokey, sin embargo, podían programar desafíos pequeños. La programación de los robots disparó colaboraciones y reflexiones entre los niños inclusive después que la computadora había sido apagada.

En resumen, investigaciones previas muestran que aprender algunos conceptos de programación en educación infantil es posible y que este aprendizaje promueve el desarrollo cognitivo. El propósito de este artículo es contribuir a esta línea de indagación haciendo foco en el aprendizaje de la programación de robots y su relación con la comprensión de la disciplina y el oficio de un especialista en computación.

Método

Diseño

Este estudio tuvo un carácter exploratorio con aportes del enfoque de la investigación-acción. Nuestro equipo multidisciplinario de la Universidad Nacional de Córdoba (UNC++) trabajó en jornadas de capacitación docente en enseñanza de la programación con dos maestras que tenían 30 años de experiencia educación infantil².

Muestra

Junto con las docentes, desarrollamos una secuencia didáctica para enseñar a 46 niños de salas de 3 y 5 años a programar robots. Las docentes se acercaron a tomar nuestro curso de Didáctica de la Programación en la Universidad y se les ofreció trabajar en conjunto para desarrollar una propuesta para el nivel recuperando su experiencia.

Procedimientos

Utilizamos la plataforma “UNC++Duino³” desarrollada para programar robots que permite el uso de múltiples lenguajes de programación según las habilidades del programador. Para la educación infantil, los niños pueden utilizar un lenguaje icónico de flechas y dibujos que representan códigos o comandos para programar cada una de las acciones o movimientos del robot. Las flechas con diferentes sentidos representan las acciones de avanzar o doblar. Una imagen del robot chocando con una caja representa un obstáculo frente al robot y sirve para introducir el concepto de condicional en computación (if).

Asimismo, dentro de la misma plataforma los estudiantes más avanzados pueden programar un robot utilizando lenguajes formales de programación (Python o C++) y controlar la velocidad y distancia de los desplazamientos del robot (con el lenguaje icónico estas variables están previamente definidas). Esta plataforma fue diseñada para programar los Robots N6, de la firma Robotgroup. Este es un robot robusto,

2 Las docentes Ana María Bustos y Marcela Damonte pertenecen a la escuela Juan Mantovani.

3 Dirigió Luciana Benotti y desarrolló Marcos Gómez con la financiación de una beca BiD, UNC.

montado sobre placas Arduino, que cuenta de una estructura de tres ruedas y sensores.

Nos basamos en la experiencia de las docentes de educación infantil para diseñar una secuencia didáctica de programación de robots. La secuencia quedó definida en tres momentos:

Durante el primer momento, las docentes sugirieron trabajar los movimientos de los robots con el cuerpo. Para las maestras, los niños debían experimentar con su propio cuerpo los desplazamientos del robot antes de transferir estas acciones a objetos. Inspirados en el juego “LightBot” y las actividades para educación infantil de Code.org, las docentes diseñaron para esta etapa un juego de piso en donde marcaron una cuadrícula de 5 por 5. En algunas celdas se ubicaron los desafíos que estaban relacionados con temas de educación para salud tales como “lavado de manos”. Los niños tiraban un dado que contenía en cada cara una imagen de hábitos de higiene y debían encontrar esa imagen en la cuadrícula. Luego, los niños elegían una secuencia de flechas que el robot debía seguir hasta llegar a la imagen seleccionada. Había tres tipos de flechas: seguir derecho, doblar a la derecha y doblar a la izquierda. (Ver Imagen 1)



Imagen 1: Juego de cuadrícula en el piso con flechas

El segundo momento consistió en replicar el juego de piso en un juego de mesa. Utilizando un tablero del tamaño de la mesa en una cuadrícula de 6 por 4. Los niños debían desplazar robots realizados de botellas de plástico. Las docentes dejaron que sus alumnos eligieran este juego durante más de un mes como parte de los juegos electivos de la sala. Este fue el primer paso para pasar de hacer una serie de movimientos con su propio cuerpo a mover objetos como el robot de plástico.

El tercer momento consistió en programar un robot N6 con la plataforma UNC++Duino. Dos docentes e investigadores junto con las maestras de la sala desarrollaron este momento en el centro. Visitamos la sala de educación infantil tres veces. Durante la primera visita los niños debían programar el robot N6 para que se desplazara en la

cuadrícula del piso (se replicó el juego de piso del primer momento de la secuencia didáctica, pero con el robot N6). En esta clase se ofrecieron los conceptos de secuencia y parámetro usando un comando con el símbolo de notas musicales que representaba la función de cantar (el robot reproducía una música electrónica sencilla). Había dos opciones de canciones. Durante la segunda visita los niños programaron un robot para esquivar objetos usando el condicional (if). En esta ocasión, trabajaron en el piso sin las restricciones de la cuadrícula. En la visita final los niños usaron la función de “ciclos” (“loops”). La tarea requería que el robot avanzara múltiples veces a partir del uso de una misma orden.

Instrumentos de recolección de datos

Observaciones: Realizamos observaciones en todas las visitas del tercer momento de la secuencia didáctica.

Trabajo de los estudiantes: En cada una de las visitas los estudiantes completaron una actividad de múltiple opción o una actividad en papel similar a la propuesta de la clase. Por ejemplo, para el día en que programaron el desplazamiento secuencial los estudiantes debían completar en una cuadrícula en papel el camino para que el robot llegara a algún punto usando flechas. Para el día en que programaron con condicional, los estudiantes hicieron un ejercicio de múltiple opción que permitía elegir al ícono que representa “if” ante un dibujo con la situación de obstáculo frente al robot.

Grupo focal: Al final de toda la intervención se realizó un grupo focal con tres salas diferentes del mismo centro escolar. Solo una de las salas no había recibido la intervención en ninguno de sus momentos y fue usada a modo de grupo de comparación, puesto que los estudiantes compartían las mismas características socio económicas que las otras salas, sin las pretensiones de querer hacer un diseño experimental con un grupo control. El grupo focal indagaba sobre los tipos de trabajos que diferentes profesiones realizan, incluyendo profesiones que se relacionan con disciplinas obligatorias en el currículum de educación infantil tales como música, teatro y matemáticas. También preguntamos por oficios de otras disciplinas no obligatorias en el currículum como antropología y computación. Buscábamos entender si el acceso a una disciplina les ofrecía una visión sobre el oficio en esa área. Durante el grupo focal mostramos una aspiradora robot (marca Roomba) que se desplaza con ruedas y aspira mientras se traslada. Pedimos a los niños que describieran qué era el objeto (la aspiradora) y preguntamos además por qué pensaban que se movía y doblaba al encontrarse con un obstáculo.

Resultados

Correspondencia entre programa y máquina

En términos generales las observaciones de clases mostraron que los niños pudieron establecer la correspondencia entre los desplazamientos del robot y el programa que ellos habían escrito en la computadora. Es decir, pudieron establecer que el programa

escrito en una computadora se cargaba luego en la memoria del robot y que los movimientos del robot eran una ejecución de esa trayectoria programada por ellos. De esta manera el robot era para ellos una máquina programable. Por ejemplo, en una clase la maestra preguntó:

Maestra: Si le digo al robot “muévete para adelante”. ¿El robot se mueve?

Estudiantes: No

Niña: le tienes que dar la flechita [indicando que tenían que programar mediante el lenguaje icónico de flechas]

Niño: lo tienes que programar

Este tipo de diálogos fue frecuente en las observaciones de clases. En una oportunidad el robot no realizó el circuito que los niños esperaban que hiciera según la trayectoria que habíamos establecido. Es decir, el robot iba en otras direcciones a las pautadas y programadas. Antes esta situación, un niño apuntó a que el robot estaba haciendo exactamente lo que habíamos programado. Este niño recordaba el programa de memoria y notó que habíamos omitido una orden. El niño fue a la sala donde estaba la computadora (la prueba de desplazamiento del robot se realizaba en un pasillo adyacente a la sala), leyó el programa y le comunicó al profesor que lo habíamos programado erróneamente. No sólo que el niño entendía la secuencia de pasos programada, sino que pudo corresponder el código de la computadora con los desplazamientos del robot.

Secuencia

Una noción central en programación estructurada es la de secuencia, donde la máquina lee y ejecuta una lista de códigos en un orden. Para enseñar este concepto las docentes diseñaron varios juegos donde el objetivo era guiar a través de flechas a un robot (que en un primer momento era un niño o una maestra y luego fue un juguete). Luego de jugar el juego de piso donde los niños debían elegir una secuencia de flechas para conducir al robot a la imagen requerida, los niños replicaron esta actividad de manera individual en una hoja de papel que contenía una cuadrícula de 5 por 5 donde se encontraba en un extremo el robot y en el otro la casita a donde debía llegar. Los niños debían armar el camino con flechas que el robot debía seguir. De 24 niños de sala de 5 años, 12 seleccionaron el camino más corto sin giros innecesarios. Solo 7 de estos 12 niños completaron toda la secuencia. Los 5 estudiantes restantes se saltaron una flecha de giro en la secuencia. De la otra mitad de estudiantes que eligieron un camino sinuoso para su robot, 4 estudiantes completaron la secuencia de manera correcta. Los otros 8 estudiantes también se saltaron una flecha de giro. (Ver tabla 1)

Frecuencia de niños por tipo de camino N=44	Completaron toda la secuencia	Se saltaron una flecha
22 niños seleccionaron el camino más corto	12	10
22 niños seleccionaron un camino sinuoso	9	13

Tabla 1: Actividad de secuencia de flechas para guiar al robot a su casa en papel

Los niños de tres años mostraron resultados similares, aunque 2 de los 20 estudiantes no pudieron completar la tarea. La mitad de ellos pudo escribir una secuencia de flechas, pero solo la otra mitad recordó incluir flechas de giro. Estos resultados muestran que en general, la mitad de los estudiantes pueden escribir una secuencia correcta. Al realizar un práctico de múltiple opción para evaluar secuencia, sólo el 10% de los estudiantes eligió la opción incorrecta para llegar a la meta.

Parámetros, Ciclos y Condicionales

Para abordar el concepto de parámetros, en tanto variable que recibe una rutina o un método, se agregó la función de seleccionar una canción para que el robot ejecutara. Los niños podían elegir entre dos canciones a través de diferentes íconos de funciones, una de navidad (identificada con un símbolo navideño) y otra el feliz cumpleaños (identificada con un saxofón). En relación al concepto de parámetros, la gran mayoría de los alumnos comprendió que podía programar a un robot para que cantara una u otra canción, es decir, que la máquina ejecutara una u otra rutina que se agregaba a la secuencia de flechas para el desplazamiento. Elegir la canción fue muy particularmente emocionante para los niños.

Respecto de la noción de condicional, en tanto instrucción que se ejecuta o no dependiendo del valor que tome la variable, se les ofreció a los niños la flecha para avanzar, la flecha para doblar y un ícono donde el robot estaba frente a una caja que representaba un obstáculo frente al robot. Es decir, los niños podían agregar a su secuencia de flechas este ícono de obstáculo para que su robot no chocara frente a objetos. Fue difícil evaluar cuánto se apropiaron de este concepto, dado que algunos niños elegían la opción de seguir a través de la flecha de avance (en vez de seleccionar el ícono del obstáculo y la flecha de doblar) porque expresaron abiertamente que querían que el robot chocara. En diálogo con los niños establecimos que sí comprendían el condicional pero que sin embargo les divertía que el robot chocara y por eso no lo seleccionaban en la construcción de su código. Al seleccionar la opción incorrecta la maestra le pregunta a una de las niñas:

Maestra- ¿Estás segura de que eso es así?

Niña: Yo sé que va la caja (refiriendo al ícono del obstáculo), pero yo quiero que el robot choque y por eso no lo pongo.

Estas expresiones también dan cuenta de que comprendieron que podían aplicar una función condicional para evitar el choque.

Finalmente, nos sorprendimos al observar cómo los niños aprendieron “ciclos”. Ciclos en programación es una sentencia que se ejecuta la cantidad de veces que se designe. Tiene el mismo sentido matemático que la multiplicación en tanto cantidad de veces. Para enseñar ciclos propusimos a los estudiantes un desafío donde el robot pudiera avanzar diez veces, pero usando la menor cantidad de flechas y teniendo la opción de colocar un número que resumía la cantidad de flechas deseadas. Un alumno eligió poner dos flechas y avanzar cada una 5 veces “porque 5 veces dos flechas son 10 avances”. Otros alumnos directamente ponían una flecha e indicaban avanzar 10 veces con número. Además de escribir el código de la manera esperada, se tomó un ejercicio de múltiple opción donde a partir de una imagen que mostraba la trayectoria de un robot los estudiantes debían elegir cuántas flechas avanzaba. El 85% de los niños contestaron correctamente el práctico de múltiple opción para evaluar ciclos. Aprender ciclos podría contribuir al desarrollo necesario para aprender multiplicación.

Aprendizaje sobre la disciplina Programación

Una preocupación en la comunidad educativa es que los jóvenes no saben de qué se trata la programación ni mucho menos las CC debido a la falta de exposición a esta disciplina en la escuela obligatoria (Martínez y Echeveste, 2015). En ese sentido la computación está en desventaja con otras disciplinas como las matemáticas o la historia que se enseñan desde la educación infantil. La preocupación radica en que el acceso temprano a la disciplina y el conocimiento sobre qué consiste una disciplina es una variable que influye en las decisiones de elección de carreras. Teniendo en cuenta estos debates es que nos pareció interesante indagar qué concepciones sobre qué es una computadora y qué es programar construyeron los niños que recibieron la experiencia de programación y comparamos estos resultados con niños de la misma escuela que no recibieron la secuencia didáctica. Los resultados del grupo focal entre las salas que recibieron la intervención y la sala que no recibió la intervención mostraron claras diferencias en las concepciones que construyeron los diferentes grupos de niños sobre qué es una computadora, cómo se programa y qué hace un programador. La tabla que sigue resume las respuestas de las tres salas que participaron del grupo focal que se repitió en cada sala de manera separada (Tabla 2).

Grupo de niños	Niños de 3 y 4 años que recibieron clases de programación de robots	Niños de 5 y 6 años que recibieron clases de programación de robots	Niños de 5 y 6 años que NO recibieron clases de programación de robots
Preguntas			
¿Qué es esto? Se les muestra una aspiradora robot	Un equipo de música, una máquina para dibujar, un robot.	Una aspiradora robot	Una aspiradora
¿Por qué piensan que se mueve?	Porque tiene botones, porque fue programada, porque tiene una computadora.	Porque tiene una computadora y un programador le puso unas flechitas para que se moviera	Porque tiene ruedas
¿Por qué dobla cuando se enfrenta con objetos?	Porque limpia, porque tiene un motor (no logran relacionar con su trabajo previo)	Porque un programador lo programó con una flechita que dobla	Porque doblan las ruedas
¿Qué hace un especialista en computación?	Programa robots	Programa robots, autitos, juguetes, trenes, autos.	Arma y arregla computadoras

Tabla 2: Síntesis de los grupos focales realizados con los niños de educación infantil

Los niños que sí habían recibido las clases de programación pudieron identificar que la aspiradora era un robot y que se desplazaba porque un programador le había puesto “flechitas” es decir un código secuenciado. Al contrario, los estudiantes que no recibieron las clases de programación mencionaron sistemáticamente que la aspiradora se movía porque tenía ruedas y no la identificaron como robot. A diferencia de sus pares de 5 años, los niños de sala de 3 (3 y 4 años de edad) no reconocieron el condicional en el programa de la aspiradora robot. Los alumnos de 5 años reconocieron que la aspiradora doblaba porque había sido programada con el condicional. Quienes no recibieron la innovación mencionaron que la aspiradora doblaba porque tenía ruedas.

Al indagar sobre qué hace un programador también hubo respuestas bien diferentes entre los grupos. Los niños que habían recibido las clases de programación contestaron que un programador programa robots y los de 5 años generalizaron y dijeron que programa robots, juguetes y hasta el auto de su papá. El grupo de alumnos que no recibió la innovación contestó lo mismo que suelen decir los estudiantes de secundaria: “un programador arregla la computadora” (Martínez y Echeveste, 2015).

Una hipótesis de trabajo que nos permite sostener esta experiencia es que los estudiantes pueden cambiar sus nociones sobre la disciplina desde muy temprana edad. En este caso, se observó una diferencia cualitativamente diferente entre los niños que recibieron una sola unidad didáctica de programación relativamente corta.

Nos preguntamos sobre cuáles son las nociones sobre la disciplina que construirían los niños si tuvieran acceso a experiencias de programación a lo largo de su escolaridad.

Conclusiones y Discusión

Las respuestas de los estudiantes mostraron que aquellos niños que recibieron la intervención pudieron establecer una relación diferente con la computación que los niños que no recibieron la intervención. Comprendieron que las máquinas tales como una aspiradora, tienen una computadora y un programa. También pudieron transferir estos aprendizajes a otras máquinas de uso cotidiano (un juguete, un auto). Comprendieron una noción central de la disciplina que es la computadora como objeto programable. Se sensibilizaron además de conceptos más complejos como ciclos, condicionales, parámetros, variables, que si bien no formalizaron o pudieron definir, si pudieron usar y comprender. En términos del pensamiento computacional, se apropiaron de conceptos centrales de la disciplina.

Asimismo, desarrollaron prácticas computacionales, es decir estrategias propias de los programadores tales como depurar cuando revisaron el código, ensayar, e iterar en una actitud de cooperación entre los niños. Estos son procesos centrales del pensamiento computacional.

Estos datos muestran que con una mínima intervención es posible desarrollar nociones básicas de programación y pensamiento computacional. La formación de usuarios de tecnologías programadas por otros genera ideas equivocadas sobre qué hace un especialista en computación y estarían relacionadas con la baja selección de carreras en informática (Martínez y Etcheveste, 2015). Este estudio muestra que los mitos sobre qué hace un especialista en computación pueden ser trabajados desde muy temprana edad en la escuela. Un aspecto significativo de este estudio es que los niños logran aprender la noción de que las máquinas pueden ser programadas a pesar de no tener la forma de computadoras clásica (un robot tiene una computadora, pero una aspiradora y un tren también).

Asimismo, rescatamos el trabajo realizado con las maestras de la escuela en un contexto situado en las aulas. Poder llevar innovaciones curriculares de este tipo a las instituciones educativas requiere recuperar saberes de los distintos campos y expertos. En este caso, y en la línea de trabajo de Leonel, Morgado y Kahn (2010), recuperamos las experiencias de las maestras de educación infantil con quienes pudimos construir la propuesta y llevarla a cabo en un contexto escolar con todos los alumnos. Sostenemos que el gran desafío no es solamente enseñar a programar, sino enseñar a programar en las escuelas donde los estudiantes son heterogéneos y se agrupan en salas de 30 alumnos o más dependiendo del nivel. Esta realidad requiere construir una didáctica de la programación que tenga en cuenta estos contextos para que se traduzca en propuesta de enseñanza significativas.

Referencias

- Bell, T. (2014). Establishing a nationwide CS curriculum in New Zealand high schools. *Communications of the ACM*, 57(2), 28-32.
- Benitez Larghi, S, Aguerre, C., Calamari, M., Fontecoba, A., Moguillansky, M., & de León, J. P. (2011). De brechas, pobrezas y apropiaciones. Juventud, sectores populares y TIC en la Argentina. *Revista Versión, México*, 27, 1-24.
- Bers, M. U., Flannery, L., Kazakoff, E. R., & Sullivan, A. (2014). Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. *Computers & Education*, 72, 145-157
- Clements, D. H. (2002). Computers in early childhood mathematics. *Contemporary issues in early childhood*, 3(2), 160-181
- Flannery, L. P., & Bers, M. U. (2013). Let's dance the "robot hokey-pokey!" children's programming approaches and achievement throughout early cognitive development. *Journal of research on technology in education*, 46(1), 81-101.
- Genc, Z. (2014). Parents' perceptions about the mobile technology use of preschool aged children. *Procedia-Social and Behavioral Sciences*, 146, 55-60.
- Gergen, K (2003). Action research and orders of democracy. *Action research*, 1(1), 39-56
- Kazakoff, E. R., Sullivan, A., & Bers, M. U. (2013). The effect of a classroom-based intensive robotics and programming workshop on sequencing ability in early childhood. *Early Childhood Education Journal*, 41(4), 245-255.
- Martínez, M. C., & Echeveste, M. E. (2015). Representaciones de estudiantes de primaria y secundaria sobre las Ciencias de la Computación y su oficio. *Revista de Educación a Distancia*, (46), 2-38.
- Morgado, L., Cruz, M., & Kahn, K. (2010). Preschool cookbook of computer programming topics. *Australasian Journal of Educational Technology*, 26(3), 309-326.
- Pascoe, J. M., Wood, D. L., Duffee, J. H., Kuo, A., & Committee on Psychosocial Aspects of Child and Family Health. (2016). Mediators and adverse effects of child poverty in the United States. *Pediatrics*, 137(4). e20160340
- Reynolds, A. J., Temple, J. A., Ou, S. R., Arteaga, I. A., & White, B. A. (2011). School-based early childhood education and age-28 well-being: Effects by timing, dosage, and subgroups. *Science*, 333(6040), 360-364.
- Rideout, V. (2017). The Common Sense census: Media use by kids age zero to eight. *San Francisco, CA: Common Sense Media*, 263-283.

- Román-González, M. (2015). Test de Pensamiento Computacional: principios de diseño, validación de contenido y análisis de ítems. *Perspectivas y avances de la investigación*, 279-302
- Wartella, E., Rideout, V., Lauricella, A., & Connell, S. (2013). *Parenting in the age of digital technology: A national survey*. Evanston, IL: Report of the Center on Media and Human Development, School of Communication, Northwestern University.
- Zapata-Ros, M. (2015). Pensamiento computacional: Una nueva alfabetización digital. *Revista de Educación a Distancia*, (46).

Para referenciar este artículo:

Martinez, M., & Gomez, M. (2018). Programar computadoras en Educación Infantil. *Edutec. Revista Electrónica de Tecnología Educativa*, 0(65), 40-53 (394). doi:<https://doi.org/10.21556/edutec.2018.65.1103>