

APLICANDO LENGUAJES DE PROGRAMACIÓN SCRATCH

EN LA ENSEÑANZA DE MATEMÁTICA A LOS ESTUDIANTES DE NIVELACIÓN DE LA UNIVERSIDAD NACIONAL DE EDUCACIÓN

Hugo Abril Piedra

INTRODUCCIÓN

Se suele pensar que el uso de lenguajes de programación es competencia exclusiva de los estudiantes y profesionales de la informática. Un lenguaje de programación no es otra cosa más que un lenguaje en el que están escritos los procesos lógicos de

Revista de divulgación de experiencias pedagógicas MAMAKUNA
N°1 – Diciembre/marzo-2018
ISSN: 1390-9940
pp. 87-95



funcionamiento y que aprendemos con el propósito de explicarle al ordenador qué se quiere que haga por el usuario. Este procedimiento es conocido también como programar. Pero, el beneficio de aprender a programar es que el programador desarrolla el pensamiento analítico, la resolución de problemas y la creatividad, habilidades esenciales para todo ser humano.

En este trabajo, se trata de dar a conocer las experiencias vividas con los estudiantes de la nivelación de la UNAE, en el proceso de aprendizaje de programación Scratch, con el objetivo de mejorar el aprendizaje de la matemática, a través del lenguaje de programación y las habilidades asociadas a éste. De esta manera estamos fortaleciendo las

competencias para la generación de recursos educativos recurriendo a la programación sin la necesidad de complejos procesos de formación en este campo de la informática. La experticia no es una condición para que los jóvenes construyan recursos didácticos recurriendo al lenguaje de programación Scratch.

Para alcanzar el objetivo, se trabajó con los estudiantes de matemática de los paralelos 2 y 4 vespertina de la nivelación de la UNAE, del período académico octubre 2015 – marzo 2016. El proceso comenzó con un foro sobre los beneficios de aprender a programar a tempranas edades y las ventajas de aprender a través de juegos de computadoras, lo que implicó

una revisión bibliográfica inicial. Luego, se dio a conocer una introducción práctica a la programación sin código, mediante el uso de bloques, como sistema de aprendizaje de la lógica de programación utilizada en el lenguaje de programación Scratch en lo referente a la pantalla principal e interfaz del usuario, el editor gráfico y editor de audio, movimientos básicos, bucles, condicionales y operadores, variables y listas, y preguntas. A partir de esta inducción al lenguaje de programación los estudiantes comenzaron a realizar pequeños programas relacionados con los temas tratados de la materia y por último, como proyecto de aula. El objetivo del proyecto

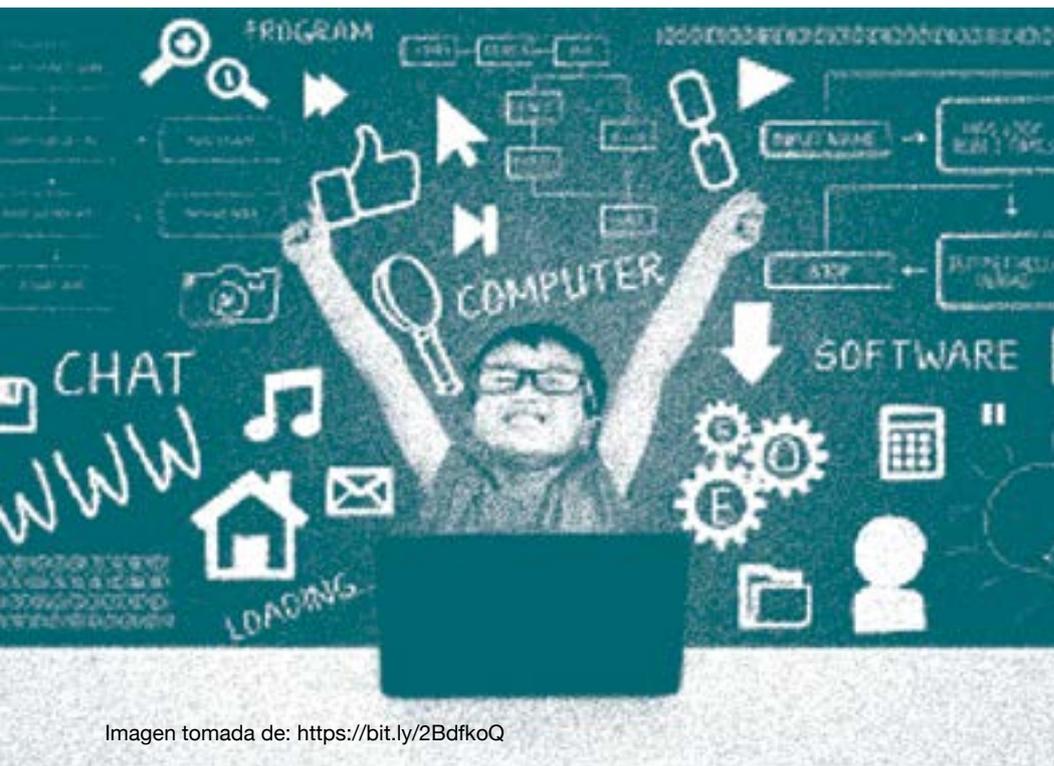


Imagen tomada de: <https://bit.ly/2BdfkoQ>

capacidades y potencialidades de los ciudadanos, objetivo del Plan Nacional del Buen Vivir de nuestro país.

La contribución de este trabajo es plantear la necesidad de incorporar a la formación docente en el contexto de la era digital, el desarrollo de

fue implementar un juego realizado a través de Scratch, que permita desarrollar cualquier tema estudiado en matemáticas en este nivel. Después se procedió a observar las calificaciones obtenidas al final de la unidad y las obtenidas en el examen final, es decir luego de aplicar Scratch, y encontramos una mejora considerable en su rendimiento.



RESULTADOS

1. EL MITO DE QUE PROGRAMAR SOLO PUEDEN HACERLO LOS PROFESIONALES DE LA INFORMÁTICA

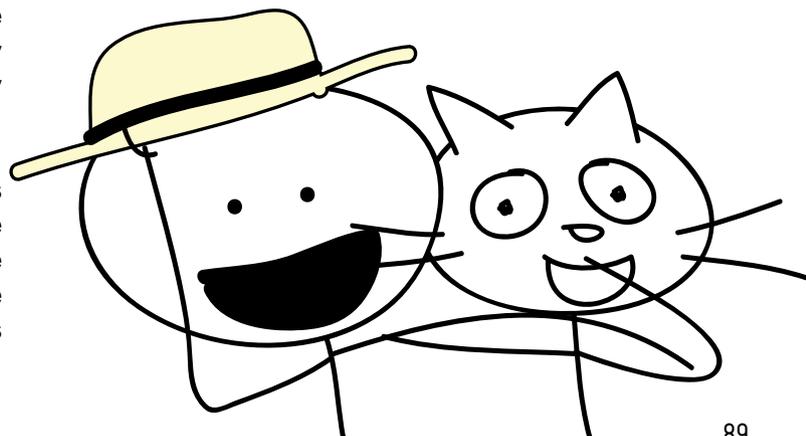
Cuando se empezó a compartir con los estudiantes la idea de aprender a programar computadoras utilizando el lenguaje de programación Scratch, todos apoyaron contentos, aunque algunos se encontraban asustados. Anteriormente se discutió en la clase sobre la importancia de aprender a programar, como también, sobre la importancia de los juegos de computadoras en el proceso del aprendizaje, entonces, se empezó a trabajar en el laboratorio de computación un taller destinado al aprendizaje del lenguaje de programación en donde muchos de ellos empezaron a tener dificultades por el hecho de que para hacer un programa hay que escribir instrucciones de una manera ordenada y sistemática para que la computadora al compilarlo y ejecutarlo entendiera.

En este momento muchos de los estudiantes manifestaban que la programación es tarea de los informáticos o de los programadores y que se hacía fácil decir que es sencillo para un ingeniero de sistemas. Pero a pesar de las primeras dificultades

los estudiantes asumieron el reto y se continuó con el desafío de programar. Y finalmente asumieron que el lenguaje de programación Scratch era una herramienta diseñada para niños de 8 a 14 años (Scratch, 2016) y que si era para niños entonces ellos ya jóvenes debería resultares más fácil el utilizarlo. Claro que no era el problema el manejo de la herramienta, el problema era que los estudiantes no podían dar las instrucciones para que la computadora les entendiera.

Se siguió trabajando, hasta que por fin empezaron a ver sus primeros resultados, ganaron confianza y ellos mismos se plantearon mayores dificultades en sus tareas, claro que siempre que cambiábamos de tema y no podían resolverlos, manifestaban que era una tarea de informáticos y que ellos no iban a ser programadores sino docentes y que si necesitaban un programa recurrirían a un profesional de la programación o informática.

Es en este momento se dio a conocer algunas de las ideas presentadas por el Dr. Mitchel Resnick, director del grupo Lifelong Kindergarten en el Media Laboratory del MIT en su Videoconferencia Aprender a Programar, Programar para Aprender, en donde empieza realizando las siguientes preguntas: “¿Realmente es importante para todos los niños aprender a escribir? Al fin y al cabo, muy pocos jóvenes serán periodistas o escritores profesionales. ¿Por qué debería aprender a escribir todo el mundo?” (Resnick, 2016, en línea), y dice que por supuesto, preguntas como estas parecen estúpidas.



Sin embargo la necesidad de aprender a escribir es indiscutible. El lenguaje de programación es el lenguaje de la era digital, y aprender su escritura es una necesidad imperativa en un mundo virtual.

Según Resnick la importancia de aprender a programar reside en “desarrollar ideas y conceptos matemáticos e informáticos (como las variables y las operaciones condicionales)” (Resnick, 2016, p. 62) así como también a la par se desarrollan



estrategias para la resolución de problemas. Esto fue muy motivante para los estudiantes y se decidió trabajar como proyecto de aula la implementación de un juego desarrollado en Scratch, que permitiera abordar temas estudiados en matemática en este nivel, el mismo que se trabajó en grupos de tres a cinco estudiantes.

A la hora de sustentar sus proyectos, todos los grupos tenían sus juegos listos, hechos por ellos, en Scratch, y sin haber sido programadores o ingenieros de sistemas. Comprendieron que ellos no aprendieron a programar en la clase de matemática, sino programaron para aprender matemática ya que al momento de realizarles preguntas que al inicio se les hacía muy difícil contestar en relación a la materia, ahora ellos lo conocían con mayor argumentación.

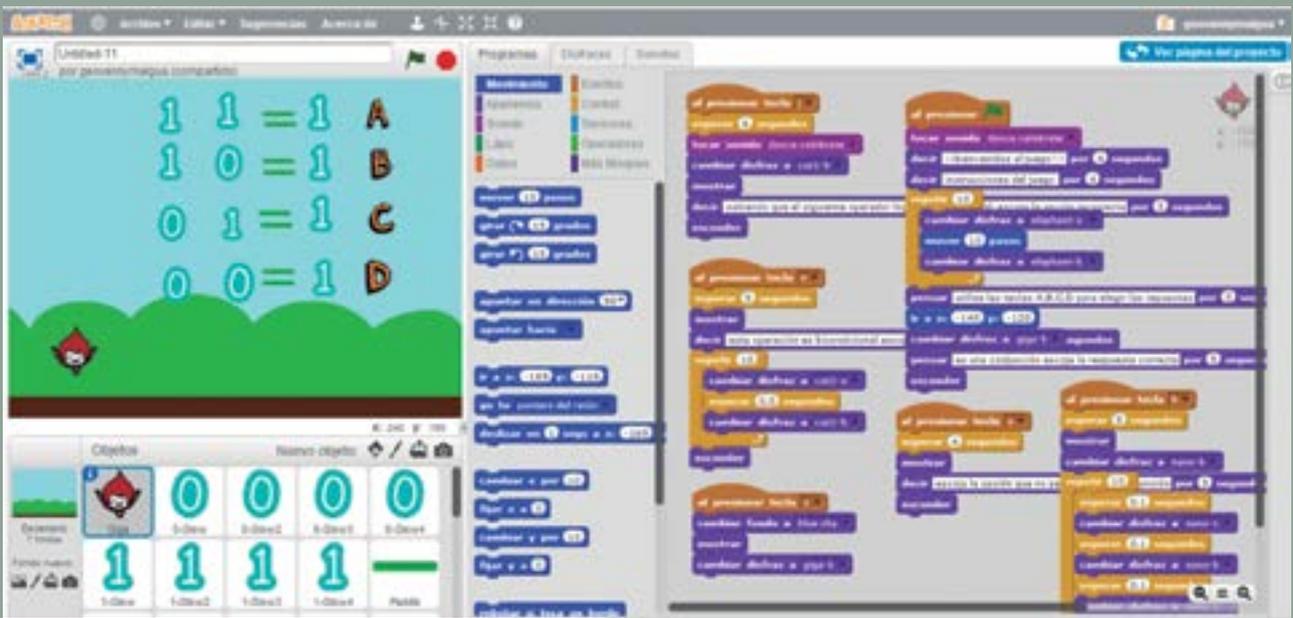
2. CON SCRATCH, DE USUARIOS A CREADORES

Scratch es un software libre desarrollado por el Grupo Lifelong Kindergarten en el Laboratorio de Medios del MIT, con el cual los estudiantes pueden desarrollar sus propios programas, sin necesidad de ser un profesional de la informática, lo que les permite desarrollar su pensamiento lógico y creativo, como también la capacidad de trabajar en equipo. Este lenguaje de programación puede ser utilizado por cualquier persona, se recomienda su uso para niños y adolescentes, en cualquier entorno.

Pero, ¿por qué el interés de enseñar a programar a los futuros docentes? Mitchel Resnick en un artículo escrito con sus colaboradores del equipo Scratch titulado “Scratch: Programación para todo”, manifiesta que los jóvenes de hoy son considerados nativos digitales por su fluidez con la tecnología digital, aunque solo interactúen con los medios digitales, sin embargo solo algunos de ellos son capaces de diseñar, crear e inventar nuevos recursos, “es como si pudieran leer pero no escribir” (Resnick, 2016, en línea). En la era digital el docente debe, además de usar tecnología educativa de vanguardia, crear sus propios recursos educativos digitales, y para ello es imperativo saber y enseñar a programar a sus estudiantes, desde tempranas edades. Actividades de aprendizaje como la maqueta del sistema solar, podrían ser desarrolladas por el niño programando con Scratch, que permite dotar de movimiento y la lógica sistémica que implica la comprensión del mundo espacial.

3. PROGRAMAR CON SCRATCH: PARA APRENDER MATEMÁTICA Y JUGANDO

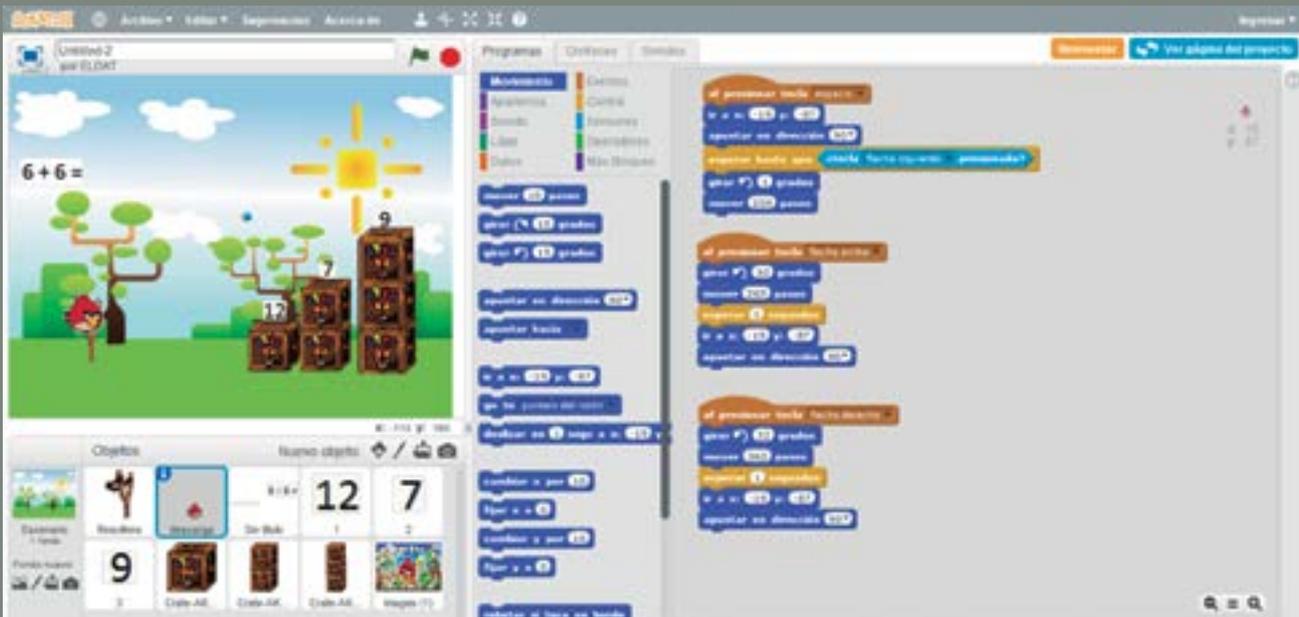
A continuación se presenta algunos de los programas desarrollados por cada uno de los grupos de estudiantes en las diferentes unidades del contenido microcurricular de la asignatura de matemática.



Interfaz del programa Scratch presentado por los estudiantes.

I. OPERADORES LÓGICOS

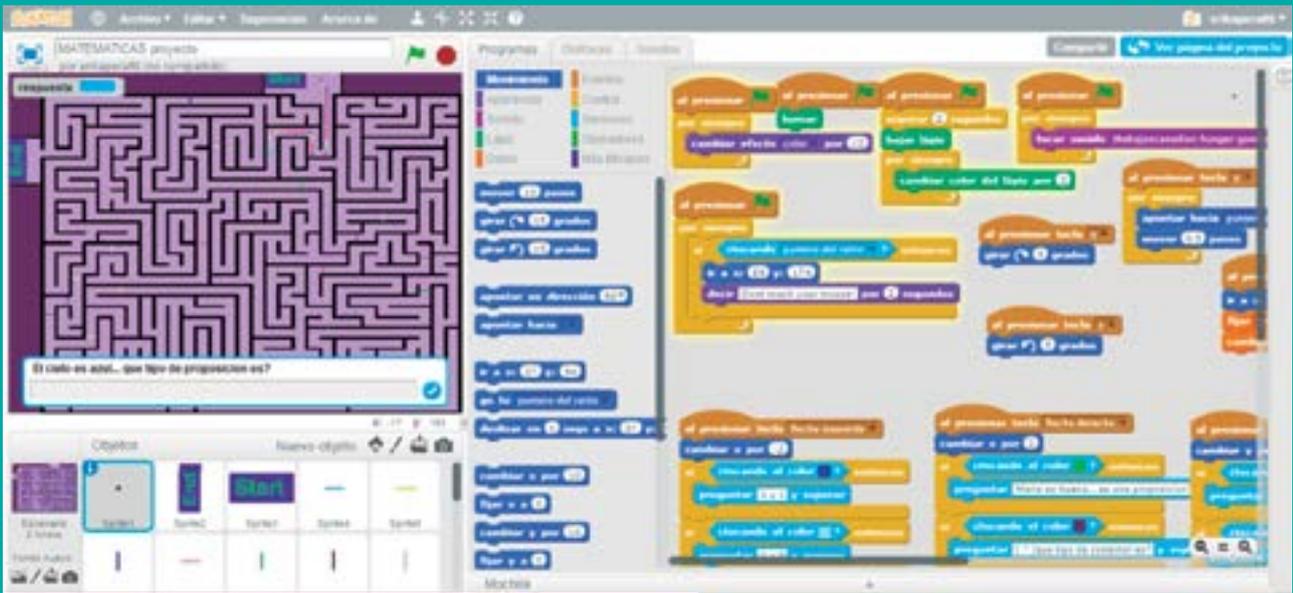
El objetivo de este programa es permitir practicar a los usuarios el tema de los operadores lógicos, en donde a través de un muñeco se hacen preguntas relacionadas a conjunciones, disyunciones, disyunciones exclusivas, condicionales o bicondicionales, el usuario tiene que escoger a través de las opciones presentadas en la pantalla la letra a la cual corresponde la respuesta correcta e inmediatamente pasa a un siguiente nivel otorgándole una puntuación. En caso de que la respuesta sea incorrecta, el programa realiza una nueva pregunta hasta que el usuario logre una puntuación correcta, para lo cual se crearon algunos objetos y en cada uno de ellos se registró instrucciones que se realizan cuando ocurra cierta acción.



Interfaz del programa Scratch presentado por los estudiantes.

II. OPERACIONES BÁSICAS

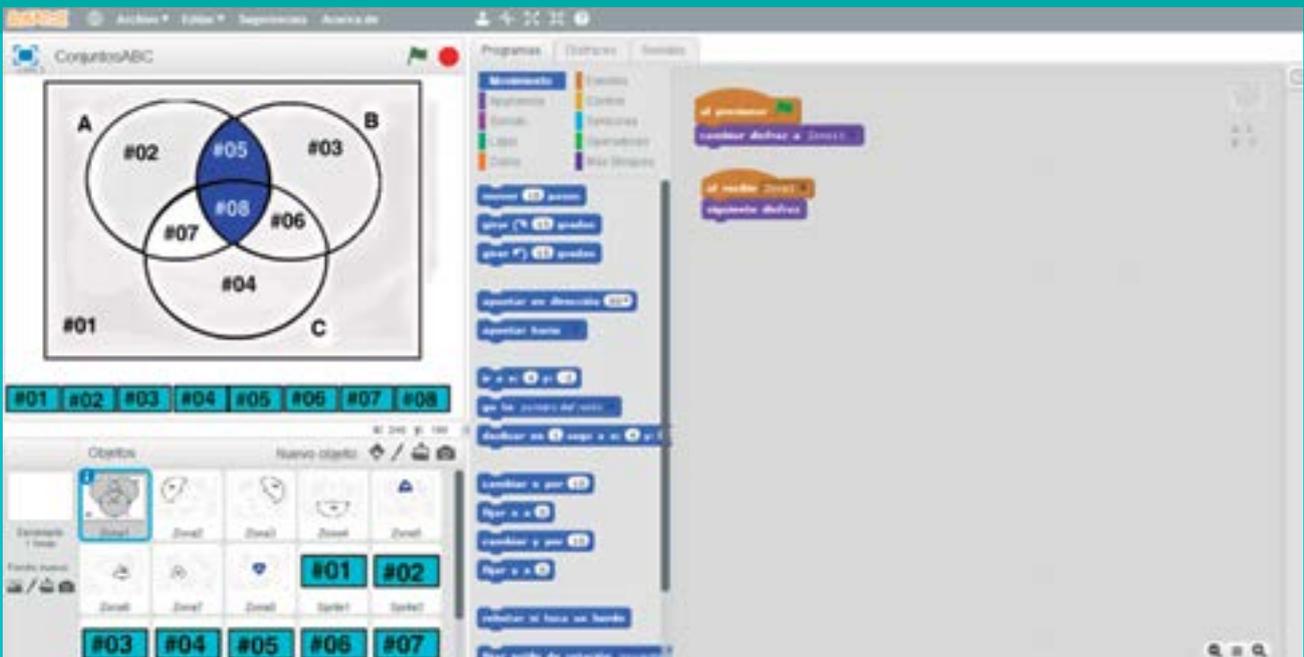
Una de las formas más divertidas de aprender a sumar, restar, multiplicar y dividir, es jugando. El usuario inicia el programa y éste le presenta una pregunta, a través de las flechas de dirección del teclado de la computadora se tiene que movilizar a un muñeco animado hasta el lugar correcto de la respuesta que se encuentran situadas en unas cajas. Se utilizó una interfaz muy flexible y divertida, aplicable para cualquier niño o joven que desee practicar las operaciones básicas.



Interfaz del programa Scratch presentado por los estudiantes.

III. PROPOSICIONES

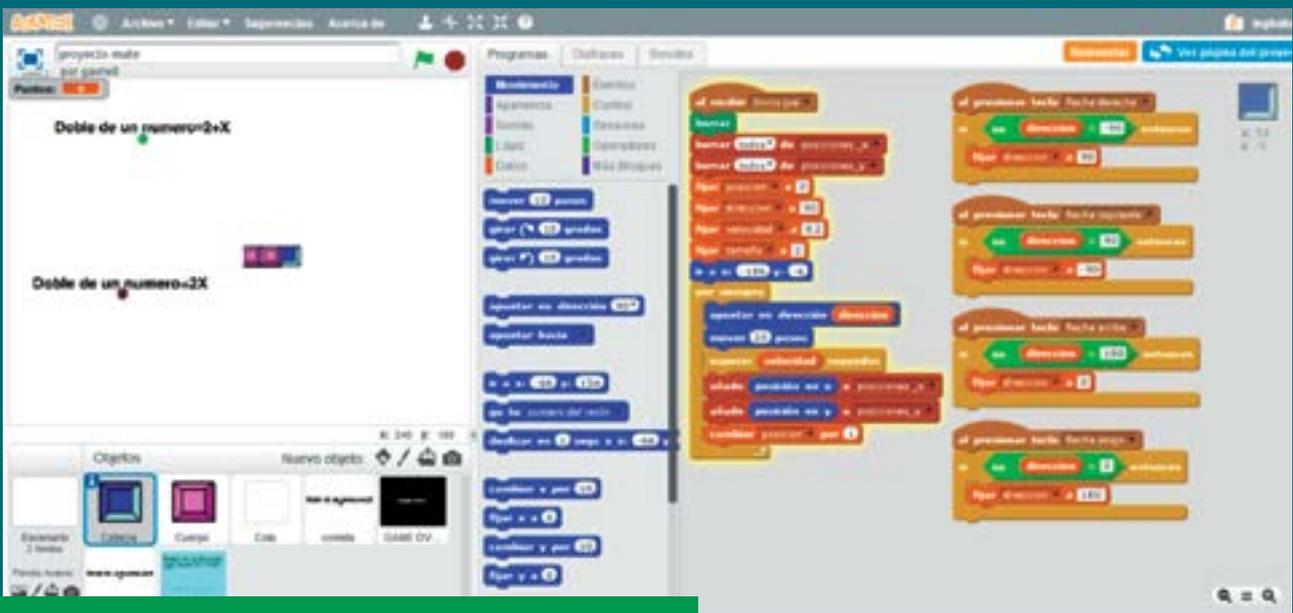
Uno de los mayores problemas al aprender proposiciones es el de poder determinar si una oración es una proposición y si esta es verdadera o falsa. Con esta problemática se escribió un programa para que los usuarios tengan que movilizarse por un laberinto hasta encontrar su salida, pero hay ciertas paredes que se pueden romper para poder cruzar de una manera más rápida. El momento que rompe las paredes, el programa le hace una pregunta, presentando proposiciones, las cuales deben ser contestadas correctamente para que le permita cumplir con el objetivo y así poder llegar a encontrar la salida.



Interfaz del programa Scratch presentado por los estudiantes.

IV. CONJUNTOS

Una de las formas más divertidas de aprender a sumar, restar, multiplicar y dividir, es jugando. El usuario inicia el programa y éste le presenta una pregunta, a través de las flechas de dirección del teclado de la computadora se tiene que movilizar a un muñeco animado hasta el lugar correcto de la respuesta que se encuentran situadas en unas cajas. Se utilizó una interfaz muy flexible y divertida, aplicable para cualquier niño o joven que desee practicar las operaciones básicas.



Interfaz del programa Scratch presentado por los estudiantes.

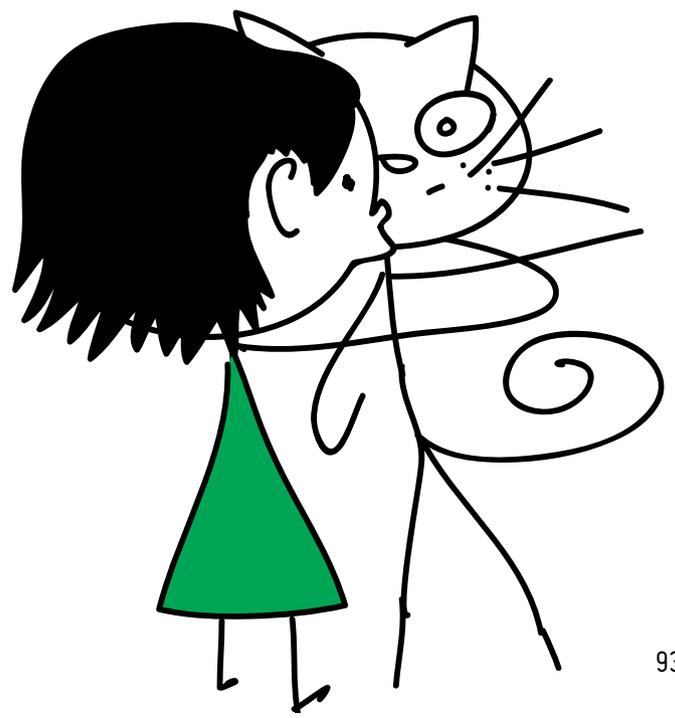
V. LENGUAJE ALGEBRAICO

El poder traducir de lenguaje verbal a lenguaje algebraico, es uno de los retos principales que tenemos al momento de resolver problemas. Por lo general estos siempre son presentados en un lenguaje natural y nosotros tenemos que presentarlos en lenguaje algebraico en un formato de ecuación, una vez realizado esto, resolvemos la ecuación y encontramos la respuesta. Por ser esta traducción lo más importante para encontrar una solución, ha sido motivo para desarrollar este programa cuyo objetivo es permitir al usuario pasar de un lenguaje verbal o natural a lenguaje algebraico.

El programa inicia con una serpiente conformada por tres cubos, su cabeza es representada por un cubo de color azul y a través de las flechas de dirección de la computadora empieza a recorrer por la pantalla, teniendo que dirigirnos hacia un punto que al momento de hacer contacto con la cabeza de la culebra se detiene y realiza una pregunta en lenguaje natural o verbal y el usuario tiene que contestar en lenguaje algebraico, en caso de ser correcta la respuesta la culebra crece, caso contrario, finaliza el juego. Es muy interesante ya que el reto es hacer que la serpiente crezca y poder determinar el ganador.

Todos los programas se realizaron en función de juegos para que sea atractivo para los usuarios, ya que actualmente los niños y jóvenes utilizan muchos estos recursos en su vida diaria y el objetivo siempre va a ser que sea una herramienta de ayuda para los docentes que permita practicar al estudiante de una manera divertida.

Los programas desarrollados además de ayudar a aprender a otros usuarios han permitido aprender a sus creadores (los estudiantes) los temas tratados en la asignatura de Matemática de una manera activa y colaborativa, uno de los principios del modelo pedagógico de la UNAE.



DISCUSIÓN Y CONCLUSIONES

Cualquier persona puede aprender a programar. Todas las personas para realizar una tarea realizan un conjunto de instrucciones que las ejecutan paso a paso para conseguir su fin, a este conjunto de instrucciones en programación les conocemos como algoritmo, es decir, tenemos desde nuestra niñez desarrollado el pensamiento computacional o algorítmico, razón por la cual, al enfrentar la necesidad de aprender a programar en un lenguaje de programación, lo podemos hacer sin mayor complicación.

La programación es una actividad importante para el proceso de enseñanza-aprendizaje. Uno de los principales retos es que los estudiantes aprendan de una manera autónoma, grupal y práctica. Los beneficios que ofrece la programación deben ser aprovechados en los procesos de aprendizaje de las diferentes áreas disciplinares.

Scratch podría convertirse en el eje transversal de los aprendizajes de niños y jóvenes. Hemos visto que Scratch nos permite desarrollar habilidades que no solo necesitan tener los estudiantes de matemática, sino que son habilidades que todo ser humano requieren en su vida diaria, entonces, se debería utilizar en todas las asignaturas que tienen el currículo de una manera transversal y solo se requiere la creatividad del docente para encontrar su aplicabilidad.

Los docentes de los nativos digitales deberían aprender a programar. Los docentes deben afrontar que sus estudiantes manejan con fluidez la tecnología digital, es decir que ellos, en su mayoría son usuarios y que un momento se cansarán de serlo y querrán pasar a crear, la pregunta es ¿quién les enseñará? Claro, por supuesto, son los futuros y actuales docentes que deberán aprender a programar para enseñar. Por lo tanto, las universidades que forman profesionales de la educación deberán incluir en sus competencias el manejo de lenguajes de programación.

REFERENCIAS

- DiSessa, A. (2000). *Changing Minds: Computers, Learning, and Literacy*. Cambridge: MIT Press
- Prensky, M. (2001) "Digital natives, digital immigrants". *On the Horizon* 9, 5, 1–6.
- Resnick, M. (2016). *Programamos*. Recuperado de <http://programamos.es/aprende-a-programar-programa-para-aprender/>
- Resnick, M. (2014). *Videoconferencia Aprender a Programa Programar para Aprender*. Santiago de Cali: Universidad Icesi.
- Resnick, M. (2007). "Sowing the seeds for a more creative society". *Learning and Leading with Technology*. 18–22.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., Kafai, Y. (2009). *Scratch: Programming for All. Communications of the ACM*, 52, 60–67. Recuperado de <http://doi.org/10.1145/1592761.1592779>
- Scratch. (2016). *Scratch*. Recuperado de <https://scratch.mit.edu/>
- Wing, J. (2006). "Computational thinking". *Commun. ACM* 49, 3, 33–35.

Hugo Abril Piedra

Docente Universidad Nacional de Educación del Ecuador UNAE

hugo.abril@unae.edu.ec