

Pensamiento Computacional mediante Programación por Bloques: intervención didáctica usando Pilas Bloques

Hernán C. Ahumada¹, Daniel A. Rivas¹, Nelson A. Contreras¹, Marta del V. Miranda¹, María V. Póliche¹

¹Fac. de Tecnología y Ciencias Aplicadas - Universidad Nacional de Catamarca
Maximio Victoria 55, 4700 Catamarca, Argentina
{hcahumada, darivas, nacontreras, mvmiranda, vpoliche}@tecno.unca.edu.ar

Resumen

Los datos de rendimiento académico de la Facultad de Tecnología y Ciencias Aplicadas indican que la materia Programación I es la que mayor cantidad de cursadas requiere para alcanzar la condición de alumno regular en la carrera Ingeniería en Informática. Tal situación da cuenta de la dificultad de los alumnos para desarrollar las capacidades básicas del Pensamiento Computacional y de los fundamentos de la Programación. Para revertir, en el corto y mediano plazo, la problemática expuesta se implementan prácticas de enseñanza alternativas incorporando como recurso didáctico algunas herramientas de programación por bloques.

La intervención didáctica realizada facilita el abordaje de conceptos relacionados con la Programación e impulsa el desarrollo de habilidades del Pensamiento Computacional. Se observa una mejora en lo cognitivo, en lo actitudinal y motivacional de los alumnos.

Palabras clave: Pensamiento Computacional, Programación por bloques.

1. Introducción

Wing (2006) expresa que “el Pensamiento Computacional es el proceso de pensamiento involucrado en la formulación de problemas y expresión de las soluciones para que las soluciones estén representadas en una forma que pueda ser efectivamente llevada a cabo por un agente de procesamiento de información”.

Para Olabe (2015), “el Pensamiento Computacional es una metodología basada en la implementación de los conceptos básicos de las ciencias de la computación para resolver problemas cotidianos, diseñar sistemas domésticos y realizar tareas rutinarias”.

La Sociedad Internacional para la Tecnología en Educación (ISTE) y la Asociación de Docentes en Ciencias de la Computación (CSTA) (ISTE and CSTA, 2011), establecen una definición operativa del Pensamiento Computacional (PC) como un proceso de solución de problemas que incluye, entre otras, las siguientes características:

- Formular problemas de manera que permitan usar computadoras y otras herramientas para

solucionarlos.

- Organizar datos de manera lógica y analizarlos.
- Representar datos mediante abstracciones, como modelos y simulaciones.
- Automatizar soluciones mediante pensamiento algorítmico (una serie de pasos ordenados).
- Identificar, analizar e implementar posibles soluciones con el objeto de encontrar la combinación de pasos y recursos más eficiente y efectiva.
- Generalizar y transferir ese proceso de solución de problemas a una gran diversidad de éstos.

Según Brennan y Resnick (2012) el Pensamiento Computacional involucra tres dimensiones en las que se puede evaluar si el estudiante adquiere la capacidad de aplicarlas en la resolución de problemas.

Dimensión 1: Conceptos Computacionales. Son aquellos conceptos que permiten implementar la estrategia de solución a problemas. Tales conceptos son: comandos, secuencias de comandos, procedimientos, repetición simple, alternativa condicional, repetición condicional, sensores, variables, parametrización e interactividad.

Dimensión 2: Prácticas Computacionales. Se refiere a la habilidad de aplicar los conceptos computacionales siguiendo criterios de buenas prácticas en Programación. Entre esas prácticas computacionales están: la descomposición en subproblemas, utilización de procedimientos, legibilidad del programa, reutilización.

Dimensión 3: Perspectivas Computacionales. Perspectiva de expresar la estrategia de solución, perspectiva de trabajar colaborativamente para encontrar la solución, perspectiva de preguntar si es posible dar solución a otros problemas desde un punto de vista computacional.

En años recientes se han presentado varias herramientas para motivar y facilitar el aprendizaje inicial en programación a niños y jóvenes. La gran mayoría de ellas adoptan el enfoque de programación por bloques. Entre las más difundidas se encuentran Scratch (Resnick et. al, 2009) y Alice (Dann et. al, 2011).

En entornos de programación por bloques las instrucciones están representadas por bloques. Un programa se construye arrastrando y encastrando bloques en el orden apropiado para un determinado fin. En este modo de programar los conceptos abstractos tienen una representación visual. Además se tiene la ventaja de que el programa escrito está libre de errores de sintaxis tan frecuentes en otros lenguajes de programación. Esto permite a los usuarios focalizarse en la creación del programa. Es por ello que las herramientas de programación por bloques son cada vez más utilizadas en los cursos de iniciación a la programación.

En nuestro país, desde el año 2013 la Fundación Sadosky lleva adelante la iniciativa Program.AR que promueve la inclusión del aprendizaje significativo de Ciencias de la Computación en las escuelas de todo el territorio nacional. Para ello lleva adelante entre otras actividades, cursos de capacitación docente y desarrollo de material y recursos didácticos (Fundación Sadosky, 2013). En tal contexto, han desarrollado la

aplicación *Pilas Bloques* que provee un entorno de programación por bloques. En *Pilas Bloques* se incluyen más de 40 actividades (desafíos) que abordan los principales conceptos de programación con niveles crecientes de dificultad.

Los desafíos incluidos en *Pilas Bloques* han sido diseñados para adoptar una metodología de enseñanza basada en problemas (Torp & Sage, 1998) que aplique una didáctica de la programación por indagación (Sanzo et. al, 2017). Para esta estrategia de enseñanza la indagación autodidacta es fundamental, siendo el docente el que guía y asiste el proceso de aprendizaje del alumno.

En la carrera Ingeniería en Informática que se dicta en la Facultad de Tecnología y Ciencias Aplicadas de la Universidad nacional de Catamarca, la materia Programación I es la que mayor cantidad de cursadas requiere para alcanzar la condición de alumno regular. Lo cual constituye una evidencia de la dificultad para desarrollar las capacidades básicas asociadas al Pensamiento Computacional.

Problemática similar a la aquí planteada ha sido consignada por docentes investigadores de Universidades tanto de nuestro país como del exterior. Tal el caso de la Universidad Nacional de San Luis (Zúñiga, 2014), y de la Escuela de Ingeniería Informática de la chilena Universidad de Valparaíso (Muñoz, 2015). En ambos trabajos se expone a las herramientas de programación por bloques como un recurso útil para enseñanza en los primeros cursos de carreras de grado en Informática.

2. Intervención Didáctica

Se entiende por intervención didáctica

a toda actuación del docente con la intencionalidad de educar y enseñar, desde una postura de mediador y facilitador del aprendizaje del alumno.

Se propone el diseño e implementación de una intervención didáctica, con la intención de subsanar, en el corto y mediano plazo, las dificultades de aprendizaje en los cursantes de Programación I. Se pretende llevar a cabo prácticas de enseñanza alternativas incorporando como recurso didáctico herramientas de programación por bloques, y tomando en cuenta las estrategias y recomendaciones de didáctica de la programación formuladas por la Fundación Sadosky.

A continuación se detallan los diferentes aspectos considerados para llevar a cabo la intervención didáctica.

2.1 Objetivos

Objetivo General

Promover el desarrollo de habilidades cognitivas básicas y específicas de pensamiento computacional y de la programación a alumnos de Programación I (2º año) de la carrera Ingeniería en Informática de la Facultad de Tecnología y Ciencias Aplicadas (FTyCa) de la Universidad Nacional de Catamarca (UNCa).

Objetivos Específicos

- Facilitar el abordaje de conceptos relacionados con la Programación.
- Impulsar el desarrollo de habilidades del Pensamiento Computacional.

- Evaluar el nivel mejora en lo cognitivo, en lo actitudinal y motivacional de los alumnos en base a la intervención didáctica propuesta.

2.2 Contenidos

Contenidos conceptuales: se abordarán algunos conceptos básicos de programación computacional. Tales conceptos son: comandos, secuencias de comandos, procedimientos, repetición simple, alternativa condicional y parametrización.

Contenidos procedimentales: habilidad de aplicar los conceptos computacionales siguiendo criterios de buenas prácticas en Programación. Entre esas prácticas computacionales están: la descomposición en subproblemas, utilización de procedimientos, legibilidad del programa, reutilización.

Contenidos actitudinales: proactividad hacia la resolución de problemas, valoración de la programación como herramienta para resolución de problemas, motivación a la aplicación de conocimientos a fines prácticos, tendencia a explorar nuevos conceptos y herramientas para lograr problemas, actitud crítica y reflexiva sobre la estrategia implementada para dar solución a cada desafío.

2.3 Actividades

En base a los contenidos seleccionados, se determina la secuencia didáctica que le dé soporte. Una secuencia didáctica es el plan actividades mediante el cual se pretende lograr el aprendizaje de los contenidos.

La presente intervención didáctica utiliza actividades incluidas en el entorno

Pilas Bloques. Se seleccionan los desafíos que tengan relación con los contenidos que se pretenden abordar.

Actividades a trabajar con los alumnos, según los contenidos.

Comandos, secuencias de comandos, procedimientos y repetición simple: El alien toca el botón, El gato en la calle, El marciano en el desierto, Tito enciende las luces, El recolector de estrellas, María y las sandías.

Alternativa condicional: El mono y las bananas, La elección del mono, Tres naranjas.

Parametrización: El planeta de Nano, Mamushka cuadrada.

2.4 Interacción Didáctica

Refiere al método de enseñanza que se utiliza en la intervención didáctica. Se seguirán los lineamientos establecidos por el equipo de expertos en didáctica de la programación de la Fundación Sadosky (Fundación Sadosky, 2013) y en las metodologías de enseñanza de programación propuestas por la Universidad Nacional de Quilmes (Martínez López, 2012).

Se utiliza una metodología de enseñanza basada en problemas (Torp & Sage, 1998) aplicando una didáctica de la programación por indagación. En tal sentido los desafíos de Pilas Bloques permiten llevar a cabo la metodología elegida puesto que se tienen situaciones problemáticas representadas visualmente y mediante enunciados escritos, a la vez que se proveen los bloques mediante los cuales es posible construir el programa que dé solución al problema planteado.

En este contexto, el alumno se enfrenta al desafío e intenta resolverlo con los conocimientos previos y herramientas

disponibles. En el caso de que sea necesario un concepto nuevo, el docente deberá esperar que el alumno formule una pregunta o plantee la inquietud para explicarlo. De este modo el rol del docente será de guiar y acompañar el proceso de aprendizaje del alumno.

2.5 Evaluación de aprendizajes

Para realizar una evaluación de los resultados de la intervención didáctica resulta necesario diseñar los instrumentos de recolección de datos que permitan registrar tanto las producciones como apreciaciones de los alumnos. También se tomarán en cuenta las observaciones y opiniones de los docentes que realizan la intervención didáctica.

Se propone como instrumento de recolección a los archivos con los programas generados por cada alumno para cada uno de los desafíos de Pilas Bloques planteados. Luego, será posible analizar si el alumno aplicó los conceptos computacionales adecuados para la elaboración de la solución.

Dado que cada desafío aborda uno o más de los contenidos conceptuales y procedimentales se valorará con una escala categórica ordinal de 0-No Aplicó, 1-Parcialmente, 2-Si Aplicó.

En las Tablas 1 y 2 se muestran los instrumentos de recolección de datos diseñados para analizar el programa que cada alumno realice para dar solución a los diferentes desafíos de *Pilas Bloques*.

Tabla 1. Grilla de evaluación de contenidos conceptuales.

Contenidos Conceptuales	0	1	2

Secuencia de Comandos			
Procedimientos			
Repetición Simple			
Alternativa condicional			
Parametrización			

Tabla 2. Grilla de evaluación de contenidos procedimentales.

Contenidos Procedimentales	0	1	2
Subproblemas			
Legibilidad			
Reutilización			

Para poder conocer la percepción de los alumnos sobre la intervención didáctica, al final de la clase se les solicitará que respondan una encuesta donde para cada pregunta categorice la respuesta con una de las siguientes opciones: NO, PARCIALMENTE, SI.

En la continuación se listan las preguntas de la encuesta dirigida a los alumnos:

1. *¿Te resultó interesante la clase de hoy?*
2. *¿Fue entretenido programar con Pilas Bloques?*
3. *¿Interpretas el concepto de repetición simple y para qué sirve?*
4. *¿Interpretas el concepto de alternativa condicional y para qué sirve?*

5. *¿Interpretas el concepto de parametrización y para qué sirve?*
6. *¿La clase de hoy, te ayudó a comprender los conceptos tratados?*
7. *¿La clase de hoy, te motiva a seguir aprendiendo a programar?*
8. *Otra opinión*

3. Resultados

Asistieron 34 alumnos a la clase que tuvo lugar en el aula 3 del Instituto de Informática de la Facultad de Tecnología y Ciencias Aplicadas. Cada uno de ellos realizó los desafíos de programación de Pilas Bloques en una PC. La clase tuvo una duración de 2 horas y 30 minutos, lapso en el cual los alumnos desarrollaron los once desafíos seleccionados para la secuencia didáctica.

En la revisión de los programas realizados por los alumnos en base a los aspectos y escala indicados en las Tablas 1 y 2, se observa una evolución en la aplicación adecuada de los contenidos tanto conceptuales como procedimentales. En cuanto a los contenidos conceptuales el tema *parametrización* resultó el que mayor dificultad de aplicar correctamente. En tanto que en el aspecto procedimental aplican de manera parcial los conceptos de subproblemas, legibilidad y reutilización.

Para conocer la valoración de los alumnos con respecto a las características de la clase, se puso a disposición en el aula virtual de la cátedra el link para acceder a la encuesta que fue respondida luego de realizadas todas las actividades previstas y antes de retirarse de la sala de

informática. En la Tabla 3 se consignan los porcentajes de cada respuesta para las preguntas formuladas.

Tabla 3. Porcentaje en las respuestas de la encuesta a alumnos

Encuesta a alumnos	NO	Parcial	SI
¿Te resultó interesante la clase de hoy?	0	16	94
¿Fue entretenido programar con <i>Pilas Bloques</i> ?	0	0	100
¿Interpretas el concepto de repetición simple y para qué sirve?	0	6	94
¿Interpretas el concepto de alternativa condicional y para qué sirve?	0	12	88
¿Interpretas el concepto de parametrización y para qué sirve?	0	24	76
¿La clase de hoy, te ayudó a comprender los conceptos tratados?	0	24	76
¿La clase de hoy, te motiva a seguir aprendiendo a programar?	6	6	88

Por los porcentajes que se observan en la Tabla 3, se infiere que la intervención didáctica es ampliamente considerada por los alumnos como positiva tanto en el aspecto cognoscitivo como en lo motivacional.

Es para destacar que las dificultades detectadas en los programas hechos por los alumnos, también son plasmadas en las encuestas respondidas por ellos en especial en las preguntas acerca de si comprenden los diferentes conceptos y su utilidad.

4. Conclusiones

Se realizó una intervención didáctica que incluyó el desarrollo de una secuencia ordenada, progresiva e interrelacionada de actividades utilizando la plataforma educativa de programación por bloques Pilas Bloques.

Se observa que Pilas Bloques facilita el abordaje de conceptos relacionados con la Programación. A la vez que, impulsa el desarrollo de habilidades del Pensamiento Computacional. Se pudo constatar una mejora en lo cognitivo, en lo actitudinal y motivacional de los alumnos en base a la intervención didáctica propuesta.

La experiencia educativa llevada a cabo permitió para que los alumnos de la cátedra Programación I logren aprendizajes significativos de los conceptos principales de la programación.

Se considera que la estrategia didáctica utilizada y la herramienta de programación Pilas Bloques constituyen un novedoso y efectivo enfoque para la enseñanza de fundamentos de programación computacional.

Referencias

ISTE and CSTA (2011). Computer Science Teachers Association and the International Society for Technology in Education. "Pensamiento Computacional, Caja de Herramientas". Eduteka. Disponible en:

<http://www.eduteka.org/modulos/9/272/2062/1>.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada (pp. 1-25).

Dann, W., Cooper S. & Pausch, R. (2011). Learning to Program with Alice. Prentice Hall

Fundación Sadosky (2013). "Informe CC 2016. Una propuesta para refundar la enseñanza de la computación en las escuelas argentinas", Buenos Aires, <http://www.fundacionsadosky.org.ar/wp-content/uploads/2014/06/cc-2016.pdf>.

-Martínez López, P. E. (2013). Las bases conceptuales de la programación. Una nueva forma de aprender a programar. Creative Commons.

Martínez López, P. E., Bonelli, E. A. & Sawady O'Connor, F.A. (2012). El nombre verdadero de la programación. Una concepción de la enseñanza de la programación para la sociedad de la información. Anales del 10mo Simposio de la Sociedad de la Información (SSI'12), dentro de las 41ras Jornadas Argentinas de Informática (JAIIO '12), 1-23. ISSN 1850-2830.

Muñoz, R., Barcelos, T. S., Villarroel, R., Barría, M., Becerra, C., Noel, R., & Frango Silveira, I. (2015). Uso de Scratch y Lego Mindstorms como apoyo a la docencia en Fundamentos de programación. In Actas de las XXI Jornadas de la Enseñanza Universitaria de la Informática (pp. 248-254). Universitat Oberta La Salle.

Olabe, X. B., Basogain, M. Á. O., & Basogain, J. C. O. (2015). Pensamiento Computacional a través de la Programación:

Paradigma de Aprendizaje. Revista de Educación a Distancia, (46).

Sanzo, A., Schapachnik, F., Factorovich, P., & O'Connor, F. S. (2017). Pilas bloques: A scenario-based children learning platform. In Learning Technologies (LACLO), 2017 Twelfth Latin American Conference on (pp. 1-6). IEEE.

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(11), 60-67.

Torp, L., & Sage, S. (1998). El Aprendizaje Basado en Problemas. (E. Litwin, Ed.). Buenos Aires: Amorrortu.

Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35.

Wing, J. (2014). Computational thinking benefits society. 40th Anniversary Blog of Social Issues in Computing, 2014.

Zúñiga, M. E., Rosas, M. V., Fernández, J., & Guerrero, R. A. (2014). El desarrollo del pensamiento computacional para la resolución de problemas en la enseñanza inicial de la programación. In XVI Workshop de Investigadores en Ciencias de la Computación