

Pensamiento Computacional mediante Programación por Bloques

Hernán C. Ahumada¹, Daniel A. Rivas², Nelson A. Contreras³, María V. Póliche⁴

(1) Departamento de Formación Básica, Facultad de Tecnología y Ciencias Aplicadas, UNCa.

hcahumada@tecno.unca.edu.ar

(2) Departamento de Informática, Facultad de Tecnología y Ciencias Aplicadas, UNCa.

darivas@tecno.unca.edu.ar

(3) Departamento de Informática, Facultad de Tecnología y Ciencias Aplicadas, UNCa.

nacontreras@tecno.unca.edu.ar

(4) Departamento de Informática, Facultad de Tecnología y Ciencias Aplicadas, UNCa.

vpoliche@tecno.unca.edu.ar

Fecha de recepción del trabajo: 13/11/2018

Fecha de aceptación del trabajo: 08/10/2019

RESUMEN: Este trabajo presenta el diseño e implementación de una intervención didáctica, cuyo objetivo es subsanar las dificultades de aprendizaje que se observan en los alumnos que cursan asignatura Fundamentos de Informática de las carreras de Ingeniería de la Facultad de Tecnología y Ciencias Aplicadas de la Universidad Nacional de Catamarca.

Se busca generar prácticas de enseñanza alternativas incorporando como recurso didáctico la herramienta de programación Pilas Bloques, desarrollada por la Fundación Sadosky en el marco de la iniciativa Program.AR. Se observa que la intervención didáctica realizada facilita el abordaje de conceptos relacionados con la Programación e impulsa el desarrollo de habilidades del Pensamiento Computacional, obteniendo una mejora en lo cognitivo, en lo actitudinal y motivacional de los alumnos.

PALABRAS CLAVES: pensamiento - computacional – programación – bloques – secuencia - didáctica

COMPUTATIONAL THINKING THROUGH BLOCK PROGRAMMING

ABSTRACT: This work introduces the design and implementation of a didactic intervention, which aims at correcting the learning difficulties in students attending the subject Fundamentals of Computer Science of the Engineering careers at the Faculty of Technology and Applied Sciences, National University of Catamarca

The idea is to generate alternative teaching practices incorporating as a didactic resource the programming tool Pilas Bloques developed by Sadosky Foundation within the Program.AR initiative. It is observed that the didactic intervention makes the approach to programming-related concepts easier as it promotes the development of Computational Thinking skills. In this way, improvements not only in students' cognitive development are obtained, but also in attitudinal and motivational ones.

KEYWORDS: thinking - computational - programming - blocks - sequence - didactics

1 INTRODUCCIÓN

1.1. Pensamiento Computacional

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

En tanto que Olabe (2015) sostiene que “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 objetivo de encontrar la combinación más eficiente y efectiva de pasos y recursos.
- Generalizar y transferirse proceso de solución de problemas a una gran diversidad de situaciones problemáticas.

Para 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 de procedimientos.

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.

1.2. Programación por Bloques

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 los 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.

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 las escuelas de la República Argentina. Paralelo 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 planteados en Pilas Bloques han sido diseñados para llevar adelante 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.

1.3. Problemática

En la carrera Ingeniería en Informática que se dicta en la Facultad de Tecnología y Ciencias Aplicadas de la Universidad Nacional de Catamarca, los resultados académicos en la cátedra *Fundamentos de Informática (1º año)* y *Programación I (2º año)* dan cuenta de las dificultades de los estudiantes para desarrollar las capacidades básicas asociadas al Pensamiento Computacional. Dicha problemática se ve reflejada en el desarrollo de los trabajos prácticos y en los resultados de las evaluaciones parciales de las mencionadas cátedras. Lo que se observa es que la mayoría de los estudiantes no logra interpretar y aplicar adecuadamente principios de algoritmia y programación.

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 indica que las herramientas de programación por bloques resultaron un recurso útil para enseñanza en los primeros cursos de carreras de grado en Informática.

1.4. Contexto

El presente trabajo se enmarca dentro del proyecto de investigación acreditado por SECyT - UNCA dentro del Programa de Incentivos por Resolución Rectoral 375/2018 “*Estudio del impacto de enseñar mediante técnicas de programación por bloques las nociones básicas de programación a alumnos de las cátedras Fundamentos de Informática (1º año) y 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)*”, que se desarrolla en el Departamento de Informática de la Facultad de Tecnología y Ciencias Aplicadas (Universidad Nacional de Catamarca).

2 ANTECEDENTES

Según el Relevamiento Anual 2016 publicado por el Ministerio de Educación, Ciencia y Tecnología de Catamarca funcionan en total 148 unidades educativas de nivel secundario en la provincia de Catamarca. La matrícula en 2016 fue de 44766 de estudiantes secundarios. Sólo en 3 de las 148 escuelas secundarias de la provincia (1 en departamento Capital y 2 del departamento Santa María) se incluyen en el Diseño

Curricular asignaturas específicamente dedicadas a la enseñanza de conceptos y lenguajes de programación. Las demás escuelas cuentan con el espacio curricular Tecnologías de la Información y la Comunicación en el que se incluyen como contenidos mínimos el uso de programas ofimáticos.

La situación descrita se manifiesta en las dificultades que presentan los alumnos de Ingeniería en Informática en las cátedras Fundamentos de Informática y Programación I. Dada la falta de conocimientos y práctica en la resolución de problemas mediante algoritmia básica, la iniciación en tales temáticas presenta dificultades en lograr los aprendizajes esperados. Es así que se llega a que la tasa de aprobados del Parcial 1 de Programación I es de apenas el 20%.

En el año 2017, docentes del Departamento de Informática de la Facultad de Tecnología y Ciencias Aplicadas impulsaron, coordinaron y supervisaron la realización por primera vez en la provincia de Catamarca del evento mundial “La Hora del Código”. Se realizaron 32 eventos en los que aproximadamente 650 alumnos de nivel primario, secundario y terciario realizaron actividades relacionadas con programación. Se utilizaron entornos de programación por bloques en computadoras y celulares. Participaron escuelas de 7 de los 16 departamentos de la provincia. Esta actividad fue Declarada de interés académico por la Facultad de Tecnología y Ciencias Aplicadas - Universidad Nacional de Catamarca mediante Resolución 258-17 y se le dio difusión por medios digitales e impresos provinciales.

También en el año 2017, la Universidad Nacional de Catamarca se presentó y resultó seleccionada en la convocatoria realizada por la Fundación Sadosky para el dictado del curso de capacitación docente “La programación y su didáctica”. El equipo responsable del dictado del curso se integra con docentes de la carrera Ingeniería en Informática de la Facultad de Tecnología y Ciencias Aplicadas. El curso estuvo destinado a docentes de nivel primario, secundario o terciario en las disciplinas de Informática, Educación Tecnológica y Matemáticas principalmente, habiendo participado también docentes de Lengua y Literatura, inglés y Teatro. La cantidad de pre-inscriptos fue de 170, de los cuales y por cuestiones de cupo fueron seleccionados 70 docentes de Catamarca Capital y de diferentes localidades del interior provincial.

Como condición para la aprobación del curso, cada docente debía diseñar el plan de clases para el dictado de contenidos del mismo a sus alumnos. Del total de 8hs. cátedras a dictar, 4 de ellas fueron con la presencia de los docentes universitarios capacitadores. Bajo tal circunstancia, se visitaron 24 Establecimientos Educativos en distintos puntos de la Provincia. Finalizaron y aprobaron la capacitación un total de 42 docentes de diferentes niveles educativos (primaria,

secundaria y terciario). Las clases dictadas por los docentes asistentes a la capacitación permitieron que más de 800 estudiantes tomen contacto con conceptos de programación fundamentales para el desarrollo del Pensamiento Computacional.

La realización de “La hora del código” y el dictado del curso de capacitación docente “La programación y su didáctica” permitieron corroborar el interés que despierta en docentes y estudiantes la propuesta de trabajar aspectos del pensamiento computacional a través de actividades que involucren el uso de herramientas de programación por bloques.

En base a estos antecedentes de enseñanza de fundamentos de programación mediante software educativo de programación basada en bloques, es que se elabora la propuesta de incorporar actividades que ayuden al desarrollo y afianzamiento de capacidades características del Pensamiento Computacional en alumnos de los primeros años de la carrera Ingeniería en Informática. En las siguientes secciones del presente trabajo se detallan objetivos y características de la intervención didáctica.

3 MATERIALES Y MÉTODOS

Intervención Didáctica

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

En este trabajo se presenta el diseño e implementación de una intervención didáctica, cuyo objetivo es subsanar, en el corto y mediano plazo, las dificultades de aprendizaje en los alumnos que cursan la asignatura *Fundamentos de Informática*. Se pretende implementar prácticas de enseñanza alternativas incorporando como recurso didáctico herramientas de programación por bloques, y tomando en cuenta las estrategias y recomendaciones didácticas formuladas por la Fundación Sadosky para introducir conceptos fundamentales de la programación. La intención de la propuesta es realizar un abordaje inicial de las tres dimensiones del pensamiento computación planteadas por Brennan y Resnick (2012), referidas a *Conceptos, Prácticas y Perspectivas Computacionales*.

El público objetivo de la actividad es, en este caso, el alumnado de la cátedra Fundamentos de Informática. El momento de realización de la intervención didáctica es en paralelo al dictado de los contenidos correspondientes a nociones de programación.

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

3.1 Objetivos

3.1.1 Objetivo General

Promover el desarrollo de habilidades cognitivas básicas y específicas de pensamiento computacional y de la programación a alumnos de Fundamentos de Informática (1º año) y de Programación I (2º año) de la carrera Ingeniería en Informática de la FTyCa de la UNCa.

3.1.2 Objetivos Específicos

1. Facilitar la presentación y aprendizaje de conceptos relacionados con la Programación.
2. Impulsar el desarrollo de habilidades del Pensamiento Computacional.
3. Evaluar las mejoras en lo cognitivo, en lo actitudinal y motivacional de los alumnos en base a la intervención didáctica propuesta.

3.2 Contenidos

3.2.1 Contenidos conceptuales

En relación a la *Dimensión 1 - Conceptos Computacionales*, se abordarán los siguientes conceptos básicos de programación computacional: comandos, secuencias de comandos, procedimientos, repetición simple, alternativa condicional y repetición condicional.

3.2.2 Contenidos procedimentales

En relación a la *Dimensión 2 -Prácticas Computacionales*, se busca lograr la habilidad de aplicar los contenidos conceptuales siguiendo criterios de buenas prácticas en Programación, tales como: planteamiento de la estrategia de solución, la descomposición en subproblemas, utilización de procedimientos, legibilidad del programa, reutilización.

3.2.2 Contenidos actitudinales

En cuanto a la *Dimensión 3-Perspectivas Computacionales* se busca incentivar en los alumnos la proactividad hacia la resolución de problemas, la 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.

3.3 Secuencia didáctica

Una secuencia didáctica es el plan actividades mediante el cual se pretende lograr el aprendizaje de los contenidos.

En base a los contenidos seleccionados en el apartado 3.2, se determina la secuencia didáctica que le dé soporte.

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.

Los contenidos conceptuales presentes en toda la intervención didáctica son: comandos, secuencias de comandos y procedimientos. En tanto que con diferentes grupos de desafíos permiten trabajar las estructuras de repetición simple, alternativa condicional y repetición condicional en orden correlativo y creciente de complejidad.

En la Tabla 1 se lista la secuencia de actividades (desafíos) de Pilas Bloques a trabajar con los alumnos, según los contenidos conceptuales.

Tabla1. Actividades según contenido conceptual.

Contenido conceptual	Actividad
Repetición simple	El gato en la calle
	El marciano en el desierto
	Tito enciende las luces
	El alien y las tuercas
	El recolector de estrellas
Alternativa condicional	María y las sandías
	El mono y las bananas
	La elección del mono
	Tres naranjas
	Tito recargado
Repetición condicional	Laberinto largo
	Super Tito 1
	Super Tito 2
	Laberinto con queso
	Fútbol para robots

3.4 Interacción didáctica

Se entiende por *interacción didáctica* al método de enseñanza que se utiliza en la intervención didáctica. En este trabajo se siguen 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 plantean 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 aplicar 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.

3.5 Valoración 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 resolver cada uno de los desafíos de Pilas Bloques planteados. Luego, mediante la observación y análisis de los mismos será posible determinar si el alumno aplicó los conceptos computacionales adecuados para la elaboración de la solución.

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

En las Tablas 2 y 3 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. Cada tabla permite registrar respectivamente el grado de aplicación de los contenidos conceptuales y procedimentales involucrados en los desafíos propuestos.

En la Tabla 2 se muestra la grilla diseñada para registrar el grado de aplicación de los contenidos conceptuales por parte de cada alumno en las diferentes actividades realizadas. Los aspectos considerados en esta tabla componen la Dimensión 1 del pensamiento computacional para Brennan y Resnick (2012).

A continuación, se explica el criterio de evaluación de cada aspecto considerado en la Tabla 2:

- Secuencia de comandos: NO (no resuelve ningún punto del problema), Parcial (resuelve algunas partes del problema), Si (resuelve la totalidad del problema).
- Procedimientos: NO (no define ningún procedimiento), Parcial (define algunos procedimientos), Si (define adecuadamente todos los procedimientos).
- Para dar una valoración sobre los contenidos conceptuales Repetición simple, Alternativa condicional y Repetición condicional, la escala utilizada refleja el caso en que No aplicó el concepto, o si lo aplicó en parcialmente (no en todo el problema), o que el concepto fue aplicado

en todas las situaciones que correspondía hacerlo en el problema.

Tabla2. Grilla de evaluación del grado de aplicación de contenidos conceptuales.

Contenido conceptual	No	Parcial	Si
Secuencia de comandos			
Procedimientos			
Repetición simple			
Alternativa condicional			
Repetición condicional			

En la Tabla 3 se muestra la grilla que permite registrar el grado de aplicación de los contenidos procedimentales por parte de cada alumno en las diferentes actividades realizadas. Los aspectos considerados en esta tabla componen la Dimensión 2 del pensamiento computacional para Brennan y Resnick (2012).

Tabla3. Grilla de evaluación del grado de aplicación de contenidos procedimentales.

Contenido procedimental	No	Parcial	Si
Subproblemas			
Legibilidad			
Reutilización			

A continuación, se explica el criterio de evaluación de cada aspecto considerado en la Tabla 3:

- Subproblemas: NO (no identifica ningún subproblema), Parcial (divide parcialmente en subproblemas), Si (plantea correctamente la totalidad de subproblemas).
- Legibilidad: NO (no asigna nombres descriptivos a procedimientos), Parcial (asigna nombres descriptivos a algunos procedimientos), Si (asigna nombres descriptivos a todos los procedimientos).
- Reutilización: NO (no aplica reutilización de procedimientos), Parcial (aplica reutilización de algunos procedimientos), Si (reutiliza procedimientos en todos los casos posibles).

Por otro lado, para poder conocer la percepción de los alumnos sobre la intervención didáctica, al final de la clase se les solicitará que respondan un formulario web a modo de 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 resultaron interesantes las clases?
- 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 repetición condicional y para qué sirve?
- 6.¿Las clases te ayudaron a comprender los conceptos tratados?
- 7.¿Las clases te motivan a seguir aprendiendo a programar?
8. Otra opinión.

4 RESULTADOS OBTENIDOS

La intervención didáctica tuvo lugar en las aulas 1 y 3 del Instituto de Informática de la Facultad de Tecnología y Ciencias Aplicadas. Se desarrolló durante 3 clases de 2 horas duración cada una, lapso en el cual los alumnos intentaron resolver los quince desafíos de Pilas Bloques seleccionados para la secuencia didáctica. En la primera clase se trabajó sobre el concepto Repetición simple, en la siguiente clase se hizo foco en el concepto Alternativa condicional y en la tercera clase se abordó el concepto Repetición condicional.

Participaron de la clase 30 alumnos, trabajando un alumno por computadora realizando los desafíos de programación de Pilas Bloques en una PC. En cada clase se desarrollaron todas las actividades previstas del concepto trabajado, en el orden indicado en la Tabla 1. A lo largo de las 3 clases los alumnos realizaron los 15 desafíos seleccionados para la secuencia didáctica.

Al finalizar cada clase, se analizaron los programas realizados por los alumnos en base a los aspectos y escala indicados en las Tablas 2 y 3, de modo tal de valorar y cuantificar el aprendizaje alcanzado en las dimensiones del pensamiento computacional consideradas.

En la Figura 1, se grafican los porcentajes de aplicación del contenido conceptual *Secuencia de comandos* en los desafíos de Pilas Bloques

En la Figura 2, se grafican los porcentajes de aplicación del contenido conceptual *Procedimientos* en los desafíos de Pilas Bloques.

En las Figuras 1 y 2, el orden de realización de los desafíos es de abajo hacia arriba, y además plantean un orden creciente de dificultad en su resolución.

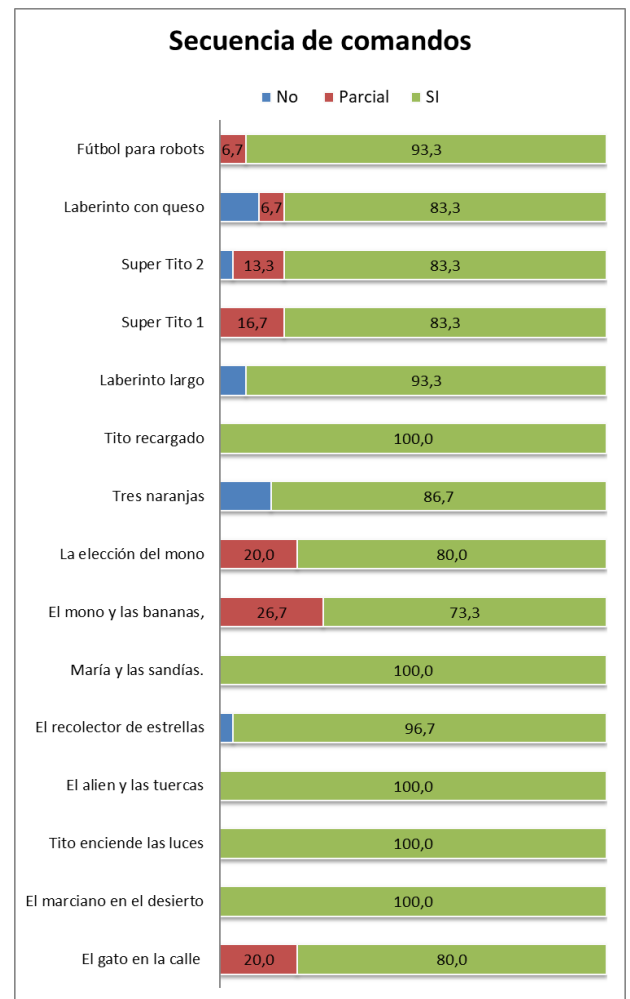


Figura 1. Porcentajes de aplicación de la secuencia de comandos en desafíos de Pilas Bloques.

En ambos gráficos, se observa una evolución en la aplicación adecuada de los contenidos tanto conceptuales.

Similar progreso se manifiesta en dimensión de los contenidos procedimentales. En cuanto a los contenidos conceptuales los temas alternativa condicional y repetición condicional resultaron los que mayor dificultad de aplicar correctamente por parte de los alumnos. En tanto que en el aspecto procedimental se observa que, en la primera clase, la mayoría de los alumnos aplican de manera parcial los conceptos de subproblemas, legibilidad y reutilización. En las siguientes clases, esta dimensión del pensamiento computacional se fue afianzando en su desarrollo por parte de los alumnos al resolver los desafíos planteados.

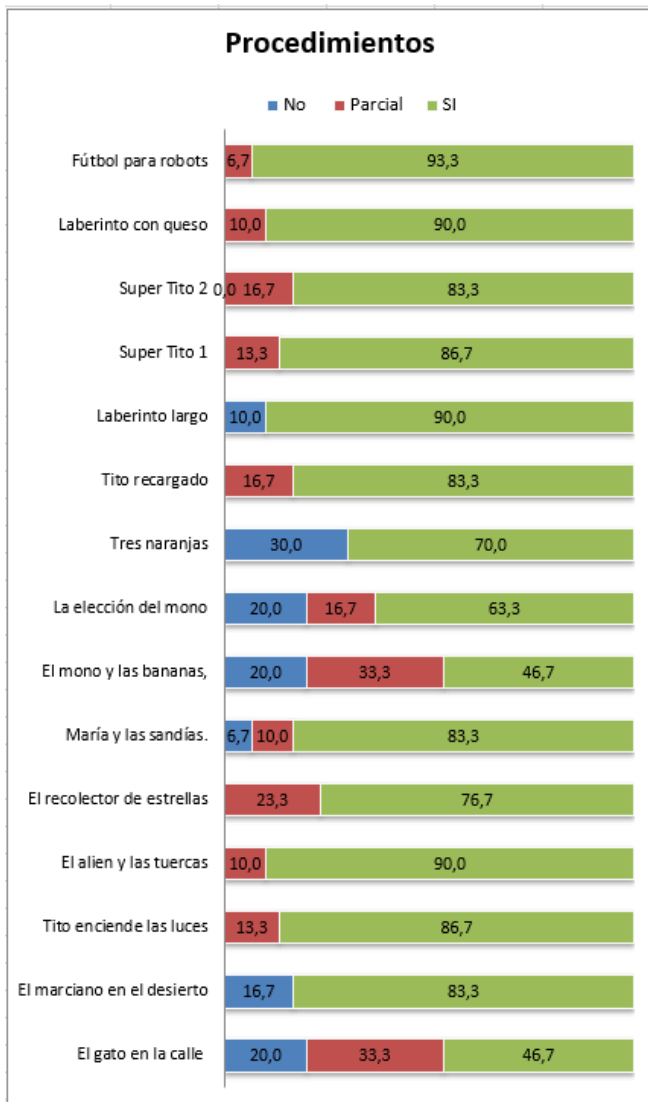


Figura 2. Porcentajes de aplicación del contenido conceptual Procedimientos

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 al final de la tercera clase. En la Tabla 4 se listan las preguntas realizadas y los porcentajes de respuesta en las categorías posibles.

En base a las respuestas dadas, 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.

Tabla4. Porcentajes de respuestas en la encuesta de valoración de las clases por parte de los alumnos.

Pregunta	No	Parcial	Si
¿Te resultaron interesantes las clases?	0	6	94
¿Fue entretenido programar con Pilas Bloques?	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	11	89
¿Interpretas el concepto de repetición condicional y para qué sirve?	0	28	72
¿Las clases te ayudaron a comprender los conceptos tratados?	0	22	78
¿Las clases te motivan a seguir aprendiendo a programar?	5	6	89

5 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 que los alumnos de la cátedra Fundamentos de Informática 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.

Se detecta que las herramientas de programación por bloques reducen la complejidad de resolver problemas desde el punto de vista computacional, puesto que permite al estudiante focalizarse en diseñar e implementar la estrategia de cómo resolver un problema y plasmarla mediante bloques, lo que evita la dificultad adicional que representa conocer y dominar la sintaxis de un lenguaje de programación.

2. REFERENCIAS

- 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. . Learning to Program with Alice. Prentice Hall. 2011.
- Fundación Sadosky. "Informe CC 2016. Una propuesta para refundar la enseñanza de la computación en las escuelas argentinas", Buenos Aires. Disponible en: <http://www.fundacionsadosky.org.ar/wp.content/uploads/2014/06/cc-2016.pdf>. 2013.
- ISTE and CSTA. 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>. 2011.
- Martínez López, P. E. Las bases conceptuales de la programación. Una nueva forma de aprender a programar. Creative Commons. 2013.
- Martínez López, P. E., Bonelli, E. A. & Sawady O'Connor, F.A. 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. 2012.
- Muñoz, R., Barcelos, T. S., Villarroel, R., Barría, M., Becerra, C., Noel, R., & Frango Silveira, I. . 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. 2015.
- Olabe, X. B., Basogain, M. Á. O., & Basogain, J. C. O.. Pensamiento Computacional a través de la Programación: Paradigma de Aprendizaje. *Revista de Educación a Distancia*, (46). 2015.
- Sanzo, A., Schapachnik, F., Factorovich, P., & O'Connor, F. S. Pilasbloques: A scenario-based children learning platform. In Learning Technologies (LACLO), 2017 Twelfth Latin American Conference on (pp. 1-6). IEEE. 2017.
- Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., & Kafai, Y. Scratch: programming for all. *Communications of the ACM*, 52(11), 60-67. 2009.
- Torp, L., & Sage, S. El Aprendizaje Basado en Problemas. (E. Litwin, Ed.). Buenos Aires: Amorrortu. 1998.
- Wing, J. M.. Computational thinking. *Communications of the ACM*, 49(3), 33-35. 2006.
- Wing, J. Computational thinking benefits society. 40th Anniversary Blog of Social Issues in Computing, 2014.
- Zúñiga, M. E., Rosas, M. V., Fernández, J., & Guerrero, R. A. 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. 2014.