

Enfoques metodológicos para la enseñanza de la programación en la escuela secundaria

Federico Amigone¹ Ruben Ramirez² Daniel José Dolz¹
Gerardo Parra¹ Jorge Rodríguez¹

`fede.amigone@fi.uncoma.edu.ar`, `rhubenariel@gmail.com`, `ddolz@fi.uncoma.edu.ar`,
`gparra@fi.uncoma.edu.ar`, `j.rodri@fi.uncoma.edu.ar`

¹*Grupo de Investigación en Lenguajes e Inteligencia Artificial*
Departamento de Teoría de la Computación - Facultad de Informática
UNIVERSIDAD NACIONAL DEL COMAHUE

²*Consejo Provincial de Educación*
MINISTERIO DE EDUCACIÓN DE LA PROVINCIA DE NEUQUÉN

Resumen

La enseñanza de la informática en la escuela secundaria está transitando una profunda transformación. Existe un amplio consenso en incorporar contenidos sobre Ciencias de la Computación en las propuestas curriculares para la educación obligatoria.

El área de conocimiento Algoritmos y Programación ocupa un lugar importante en las iniciativas y propuestas curriculares que se elaboran en este contexto. En los últimos años se han desarrollado entornos dedicados específicamente a la enseñanza de la programación. La definición de enfoques para la enseñanza y aprendizaje de la programación en el ámbito escolar es una pieza clave para el desarrollo y consolidación de los procesos que buscan aproximar las Ciencias de la Computación a la Escuela Secundaria.

En este trabajo se presenta una Línea de Investigación que busca producir conocimiento para favorecer los procesos de enseñanza y aprendizaje de la programación en ambientes escolares. Esta línea se basa en investigaciones recientes acerca de la enseñanza de la programación, en particular en el campo de Embodied Programming, enfoques

que proponen la enseñanza de la programación a partir de la modificación de código existente y estudios sobre cómo facilitar los procesos de transición de lenguajes basados en bloque a lenguajes basados en texto.

Palabras Clave: CIENCIAS DE LA COMPUTACIÓN, ENSEÑANZA DE LA PROGRAMACIÓN.

Contexto

Esta propuesta se ubica en el contexto de las iniciativas promovidas por el Grupo de Investigación en Lenguajes e Inteligencia Artificial de la Facultad de Informática.

Este trabajo se desarrolla en el ámbito del proyecto de investigación *Agentes Inteligentes. Modelos Formales y Aplicaciones para la Educación (04/F015)* que está financiado por la Universidad Nacional del Comahue a través de la Secretaría de Ciencia y Técnica. El proyecto tiene prevista una duración de cuatro años a partir de enero del 2017.

1. Introducción

Aún con diferente grado de avance en los distintos países, existe un amplio consenso

en incorporar contenidos de Ciencias de la Computación en las propuestas curriculares para la educación obligatoria. En este sentido, la enseñanza de la informática en la escuela secundaria está transitando una profunda transformación[3, 6].

En los últimos años se han producido numerosos aportes y se han desarrollado variados entornos dedicados específicamente a la enseñanza de la programación. Es por ello que el área de conocimiento Algoritmos y Programación ocupa un lugar muy importante en las iniciativas y propuestas curriculares que se elaboran en este contexto. La definición de enfoques para la enseñanza y aprendizaje de la programación en el ámbito escolar es una pieza clave para el desarrollo y consolidación de los procesos que buscan aproximar las Ciencias de la Computación a la Escuela Secundaria.

En este trabajo se presenta una Línea de Investigación que tiene como objetivo producir conocimientos y definir enfoques tendientes a favorecer los procesos de enseñanza y aprendizaje del área Algoritmos y Programación en ambientes escolares.

Esta línea se fundamenta y construye sobre investigaciones recientes en el campo de la enseñanza de la programación. En particular, se consideran las perspectivas construidas en el campo de Embodied Programming, los enfoques que proponen la enseñanza de la programación a partir de la modificación de código existente y estudios acerca de cómo facilitar los procesos de transición de lenguajes basados en bloque a basados en texto.

2. Línea de Investigación

En esta sección se presentan los ejes principales que componen la Línea de Investigación.

2.1. Enfoque didáctico disciplinar

La definición de un enfoque requiere algunas definiciones previas, entre ellas, el diseño

del andamiaje didáctico sobre el que se articularán las metodologías concretas.

El modelo propuesto parte de considerar al estudiante como un sujeto vinculado a un contexto que se encuentra atravesado por la aplicación instrumental de los campos disciplinares que se estudian. De esta forma, se persigue replantear el sentido común imperante y percibido del mundo ofreciendo herramientas cognitivas que maximicen la capacidad de análisis de la realidad que lo rodea. Se intenta que el alumno domine el campo disciplinar para que dicho dominio lo habilite a una mejor comprensión de los procesos tecnológicos que urden el funcionamiento de los espacios sociales, laborales y económicos de su contexto.

2.1.1. Punto de partida

El enfoque triangula la subjetividad, el contexto y el campo disciplinar. Desde esa terna se desprenden instancias metodológicas calibradas y funcionales. Existe mucha información que se infiere a partir de la relación ternaria propuesta. En principio, pone al sujeto en una relación contextual con su medio y permite determinar el grado de conocimiento y fricción con las manifestaciones tecnológicas que serán estudiadas.

2.1.2. Recorrido constructivista

Independientemente del grado de conocimiento disciplinar del sujeto, casi siempre podrán establecerse relaciones entre éste y los efectos de la tecnología en su contexto. Esas relaciones de problematización son utilizadas como disparadores y puntos de partidas para el recorrido didáctico. Desde la triangulación como punto de partida, comienza un recorrido que parte de un elevado nivel de abstracción técnica y a través de transposiciones didácticas, se comienza a aproximar acercamientos disciplinares concretos. Se llega al final del recorrido, cuando el sujeto consigue construir un artefacto computacional que sea vinculante a su contexto específico.

2.1.3. Dispositivos didácticos

Son diseñados en función de los niveles de cohesión entre las relaciones sujeto, contexto y disciplina. Son dispositivos pensados a partir de la problematización propuesta, y cuya aplicación permiten desarrollar, con una propuesta constructivista, determinadas nociones del campo de estudio. Estos dispositivos se aplican con un sentido incremental en su acceso concreto a la disciplina.

2.2. Ejes de trabajo

2.2.1. Embodied Programming

Embodied Learning constituye un paradigma de aprendizaje que enfatiza el uso del cuerpo en la práctica educativa, se basa en la premisa de que existe un vínculo inseparable entre cuerpo y mente en el proceso de aprendizaje [5, 2].

Desde este paradigma se supone que el aprendizaje ocurre como fenómeno interactivo entre sujetos, sus cuerpos, los artefactos en el entorno físico y las acciones que realizan con esos artefactos. Este enfoque constituye un recurso innovador para aprender conceptos abstractos como los relacionados al desarrollo de algoritmos[8, 2].

Surrogate Embodiment se refiere a un tipo de interacción donde los movimientos realizados por una persona sobre un escenario son dirigidos explícitamente por los estudiantes. Investigaciones recientes sostienen que existe cierta similitud entre Surrogate Embodiment y las prácticas computacionales que se ponen en juego durante el desarrollo de algoritmos. En el sentido de que la manipulación explícita de un sustituto se asemeja a los procesos de pensamiento de un programador [8].

En el contexto de esta Línea de Investigación se reconoce la importancia de definir un modelo que considere diferentes formas de representación posibilitando la construcción gradual de la capacidad de elaborar abstracciones y manipular conceptos y prácticas fundamentales del área Algoritmos y Programación. Se parte de experiencias concretas

que permitan, más adelante, construir conocimientos abstractos.

Surrogate Embodiment con representación física: las acciones de un sustituto externo, preferiblemente el docente, son dirigidas explícitamente por los estudiantes. La actividad se desarrolla sobre un escenario montado físicamente.

Surrogate Embodiment con representación gráfica: La actividad se desarrolla sobre un escenario representado en forma gráfica, los estudiantes indican como mover un objeto por el escenario.

Surrogate Embodiment con representación imaginaria: la manipulación del sustituto se realiza sobre un escenario imaginario, un modelo mental.

Direct Embodiment con representación imaginaria: las acciones se realizan sobre un modelo imaginario del escenario sin participación de un sustituto.

2.2.2. Enseñanza de la programación a partir de la modificación de código existente

Estudios recientes acerca de la enseñanza de conceptos y prácticas sobre algoritmos y programación sostienen partir de la modificación de código existente constituye un enfoque adecuado para estudiantes sin formación previa en el área de conocimiento[1, 7].

Pair Programming es una práctica originada en la industria del software como parte de las metodologías de Extreme Programming. Es una práctica donde dos programadores, driver y navigator, trabajan colaborativamente en diseño de algoritmos, codificación y prueba de programas.

Pair Programming demuestra ser un método efectivo para contribuir al mejoramiento de la calidad de los procesos de enseñanza y aprendizaje de la programación[9].

En esta Línea de Investigación se propone desarrollar un modelo didáctico que potencie los procesos reelaboración de código a partir de la aplicación de técnicas de Pair Programming.

Se plantea un modelo en espiral de complejidad creciente donde en cada iteración se presenta la necesidad de adecuar una porción de código a una funcionalidad diferente.

En cada ciclo, los estudiantes trabajando en parejas, analizan un programa existente, identifican porciones de código que deben ser modificadas e introducen cambios. Luego de algunas iteraciones, cuando el aprendizaje está consolidado, la pareja está en condiciones de enfrentar desafíos más complejos.

En el modelo propuesto se puede utilizar para estructurar cada iteración la siguiente secuencia de actividades:

- **Conocer y comprender:** Se presenta una pieza de código, las parejas estudian el programa para dar cuenta sobre que problema resuelve y cómo lo hace. Esta actividad, además, se constituye en momento de activación del par de programación.
- **Práctica guiada:** Se realiza con la colaboración del docente quien guía una primer intervención sobre el código.
- **Colección de desafíos:** Se propone al par de programación modificar código existente para resolver una colección de desafíos, donde la complejidad se incrementa al avanzar por el espiral.
- **Evaluación formativa:** Se despliegan actividades tendientes a mejorar la comprensión acerca del logro de los estudiantes respecto las metas.
- **Formalización conceptual:** Los aprendizajes logrados se expresan formalmente e integra lo aprendido a un marco conceptual.
- **Evaluación sumativa:** Se realiza al finalizar las iteraciones con intención de comprobar los resultados obtenidos.

2.2.3. Transición de programación basada en bloques a basada en texto

El enfoque de programación por bloques demuestra ser una opción efectiva para dar los primeros pasos.

En el transcurso de la escuela secundaria los estudiantes deberán programar usando lenguajes basados en texto. El paso a un entorno de programación tradicional, como Netbeans+java, puede resultar frustrante [4].

En esta Línea se propone estudiar entornos que favorezcan y faciliten la transición de programación basada en bloques a programación basada en texto.

En tal sentido se han realizado experiencias satisfactorias con herramientas del tipo mixtas con ambientes tipo de caja de arena.

Por herramientas de tipo mixtas se hace referencia a las que posibilitan trabajar con bloques visualizando en lo inmediato el código producido automáticamente, y viceversa. Se trabaja en la especificación de alguna herramienta que permita este paso. Idealmente dicha herramienta debería contener:

- **Sandbox:** Proponemos que el ámbito de aplicación de la herramienta no sea general, sino específico contenido en un intérprete de preferencia visual, dentro de la misma herramientas.
- **Multilenguaje:** la herramienta debería proveer la posibilidad de generar e interactuar con más de un lenguaje de programación.
- **Bidireccional:** debería ser posible generar código a partir de bloques, y viceversa.
- **Autoevaluativa:** no debería ser necesaria la ayuda de un docente para que el estudiante sepa si su programación resolvió el problema o no; aunque siempre será necesaria una instancia de evaluación de eficiencia, elegancia y posibles alternativas de solución diferentes.

- Instalable: esta línea siempre tiene en cuenta estudiantes sin acceso permanente a internet, con lo que una herramienta puramente web-enabled no estaría al alcance de toda la comunidad educativa.

Se considera importante avanzar en la obtención de una herramienta de estas características, tanto desde la especificación como desde la implementación.

3. Resultados

En el contexto de esta Línea de Investigación se espera obtener los siguientes resultados:

- Definir un modelo didáctico que articule las diferentes tipos de Embodied Programming.
- Proponer un modelo de la enseñanza basado en modificación de código existente en espiral de complejidad creciente.
- Construir especificaciones para herramientas que faciliten la transición de programación basada bloques a texto.
- Avanzar en la articulación de un enfoque general que integre la gama de perspectivas presentadas en este trabajo.

4. Formación de Recursos Humanos

Se espera que el desarrollo de esta Línea de Investigación contribuya a la formación de recursos humanos en el campo de las Ciencias de la Computación en la Educación.

En este sentido, tres de los autores de este artículo cursan maestrías orientadas a conocer, comprender y analizar procesos relacionados con las tecnologías en la educación.

Referencias

- [1] K. Brennan and M. Resnick. 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*, 2012.
- [2] C. L. Fadjó. *Developing computational thinking through grounded embodied cognition*. PhD thesis, Teachers College, 2012.
- [3] K-12 Computer Science Framework Steering Committee. *The K-12 Computer Science Framework*. ACM, 2016.
- [4] M. Kölling, N. C. Brown, and A. Altadmri. Frame-based editing: Easing the transition from blocks to text-based programming. In *Proceedings of the Workshop in Primary and Secondary Computing Education*, pages 29–38. ACM, 2015.
- [5] P. Kosmas and P. Zaphiris. Embodied cognition and its implications in education: An overview of recent literature. *International Journal of Educational and Pedagogical Sciences*, 2018.
- [6] F. Sadosky. *CC – 2016 Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas*. Fundación Sadosky, Argentina, 2013.
- [7] S. Sentance, J. Waite, and M. Kallia. Teachers’ experiences of using primm to teach programming in school. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, pages 476–482. ACM, 2019.
- [8] W. Sung, J. Ahn, and J. B. Black. Introducing computational thinking to young learners: practicing computational perspectives through embodiment in mathematics education. *Technology, Knowledge and Learning*, 22(3):443–463, 2017.
- [9] D. Tsompanoudi, M. Satratzemi, and S. Xinogalos. Distributed pair programming using collaboration scripts: An educational system and initial results. *Informatics in Education*, 14(2):291–314, 2015.