

Referencia del documento:

González Padrón, J. P. (2023). Procedimientos instruccionales para el pensamiento computacional en educación. Tecnológico de Monterrey. <https://hdl.handle.net/11285/650712>



Tecnológico
de Monterrey



Institute
for the Future
of Education



Interdisciplinary
Research
Group

E4C&CT: Ecosystem for scaling up computational thinking and reasoning for complexity

Procedimientos instruccionales para el pensamiento
computacional en educación

José Guadalupe González Padrón

Mayo 2023

Pensamiento computacional y educación

El pensamiento computacional es una habilidad crítica para la era digital, que permite a las personas abordar y resolver problemas complejos a través del razonamiento lógico y el pensamiento algorítmico. Para ello, existen diversos métodos y estrategias para enseñar el pensamiento computacional a estudiantes de todas las edades. Al incorporar estos enfoques en los entornos educativos, los profesores pueden fomentar las habilidades de pensamiento computacional y capacitar a los estudiantes para sobresalir en el mundo del siglo XXI.

El pensamiento computacional abarca un conjunto de habilidades de resolución de problemas que implican dividir problemas complejos en pasos manejables, formular algoritmos y aprovechar las herramientas computacionales para analizarlos y resolverlos. Estas habilidades son valiosas en varias disciplinas, desde informática hasta matemáticas, ingeniería y más. Las estrategias y métodos de enseñanza efectivos juegan un papel crucial en el desarrollo de habilidades de pensamiento computacional en los alumnos.

Modelo de entorno de aprendizaje

El modelado del entorno de aprendizaje (LEM: *Learning Environment Modeling*) se refiere al proceso de creación de un entorno educativo que apoye el aprendizaje efectivo y fomente la adquisición de conocimientos, habilidades y actitudes. Se trata de diseñar y estructurar el entorno de aprendizaje para optimizar la experiencia de aprendizaje de los estudiantes. Los componentes del modelado del entorno de aprendizaje (Figura 1) incluyen información, diálogo, retroalimentación, práctica y evidencia, que son cruciales para crear un entorno de aprendizaje dinámico y atractivo.

Figura 1

Componentes del modelado del entorno de aprendizaje



La información es un componente fundamental del modelado del entorno de aprendizaje. Abarca el contenido, los recursos y los materiales de instrucción que se proporcionan a los alumnos. Esto incluye libros de texto, conferencias, presentaciones multimedia, recursos en línea y otros materiales relevantes. La información debe organizarse, ser accesible y presentarse de una manera que sea significativa y atractiva

para los alumnos. La entrega eficaz de información garantiza que los alumnos tengan acceso a los conocimientos y conceptos necesarios para desarrollar su comprensión del tema.

El diálogo juega un papel crucial en el modelado del entorno de aprendizaje. Implica una comunicación significativa e interactiva entre alumnos e instructores, así como entre compañeros. El diálogo promueve la participación activa, el pensamiento crítico y el intercambio de ideas. Puede tomar varias formas, como discusiones en clase, trabajo en grupo, debates y proyectos colaborativos. A través del diálogo, los alumnos pueden aclarar su comprensión, hacer preguntas, compartir perspectivas y desarrollar sus habilidades de comunicación.

La retroalimentación es un componente esencial del modelado del entorno de aprendizaje. Se trata de proporcionar a los alumnos información oportuna y constructiva sobre su desempeño y progreso. La retroalimentación ayuda a los alumnos a comprender sus fortalezas y áreas de mejora, guiándolos hacia el logro de sus objetivos de aprendizaje. Puede ser proporcionado por instructores, compañeros o mediante autoevaluación. La retroalimentación efectiva es específica, procesable y ayuda a los alumnos a hacer los ajustes necesarios en sus estrategias y comportamientos de aprendizaje.

La práctica es otro componente crítico del modelado del entorno de aprendizaje. Implica oportunidades para que los alumnos apliquen sus conocimientos, habilidades y conceptos en contextos significativos y auténticos. La práctica puede tomar la forma de ejercicios, simulaciones, estudios de casos, experimentos, proyectos o aplicaciones del mundo real. Al participar en la práctica, los alumnos desarrollan y refuerzan su comprensión, mejoran sus habilidades y generan confianza en sus habilidades.

La evidencia se refiere a la recopilación y el análisis de datos e información para evaluar el progreso de los alumnos, la eficacia de la instrucción y el entorno general de aprendizaje. Implica el uso de evaluaciones, pruebas, proyectos, observaciones y otras medidas para recopilar evidencia de los resultados del aprendizaje. Las prácticas basadas en evidencia permiten a los instructores identificar áreas de mejora, adaptar estrategias de instrucción y brindar apoyo específico a los alumnos. También permite a los alumnos monitorear su propio progreso y tomar posesión de su aprendizaje.

Así, el modelado del entorno de aprendizaje implica el diseño y estructuración del entorno educativo para optimizar el aprendizaje. Sus componentes trabajan juntos para crear un entorno de aprendizaje atractivo y efectivo. Al incorporar estos componentes en el diseño educativo, los educadores pueden mejorar la experiencia de aprendizaje, facilitar la adquisición de conocimientos y promover el desarrollo del pensamiento crítico, la resolución de problemas y otras habilidades esenciales.

Métodos y estrategias para la enseñanza del pensamiento computacional

El aprendizaje basado en problemas (PBL) involucra a los estudiantes en problemas del mundo real, promoviendo la colaboración, el pensamiento crítico y las habilidades de pensamiento computacional. Los estudiantes trabajan en proyectos que requieren la deconstrucción de problemas, el diseño de algoritmos y la programación, lo que les permite aplicar el pensamiento computacional en contextos auténticos (Brennan & Resnick, 2012). El uso de juegos como herramienta educativa puede mejorar el compromiso y motivar a los alumnos a desarrollar habilidades de pensamiento computacional. Los entornos de aprendizaje basados en juegos brindan oportunidades para que los estudiantes resuelvan acertijos, diseñen algoritmos y superen desafíos, fomentando el pensamiento computacional (Kiili et al., 2019).

Las actividades desconectadas (unplugged activities) implican enseñar conceptos de pensamiento computacional sin la necesidad de computadoras o dispositivos digitales. A través de ejercicios prácticos, los estudiantes aprenden los principios fundamentales del pensamiento computacional, como la descomposición, el reconocimiento de patrones, la abstracción y el pensamiento algorítmico. Las actividades desconectadas pueden ser particularmente beneficiosas para los estudiantes más jóvenes (Barr et al., 2011).

Los entornos de aprendizaje colaborativo promueven la interacción entre pares, la comunicación y la resolución de problemas. Las actividades y discusiones grupales permiten a los estudiantes aprender unos de otros, compartir diversas perspectivas y aplicar colectivamente estrategias de pensamiento computacional para resolver problemas (Nicolaidou et al., 2013).

La integración del pensamiento computacional en varias áreas temáticas, como las ciencias, las matemáticas y las humanidades, puede mejorar la comprensión interdisciplinaria y la aplicación del pensamiento computacional. Al incorporar conceptos de pensamiento computacional en los planes de estudio existentes, los educadores brindan a los alumnos oportunidades para explorar conexiones entre diferentes campos y desarrollar habilidades para resolver problemas. (Weintrop et al., 2016).

Enseñar el pensamiento algorítmico implica presentar a los alumnos el concepto de algoritmos, sus características y el proceso paso a paso de resolución de problemas. Los profesores pueden usar diagramas de flujo, pseudocódigo y ejercicios estructurados para ayudar a los estudiantes a practicar la creación y el análisis de algoritmos (Wing, 2006).

El modelado computacional implica representar fenómenos del mundo real utilizando herramientas computacionales. Los estudiantes aprenden a identificar patrones, crear simulaciones y analizar datos usando modelos computacionales. Este método permite a los alumnos aplicar habilidades de pensamiento computacional a problemas complejos y obtener una comprensión más profunda del tema (Wilensky & Reisman, 2006).

La abstracción implica identificar las características esenciales de un problema mientras se filtran los detalles innecesarios. Enseñar a los alumnos a abstraer y generalizar les permite desarrollar estrategias de resolución de problemas aplicables a una amplia gama de escenarios. Al enfatizar estas habilidades, los educadores cultivan la capacidad de los alumnos para analizar problemas en diferentes niveles de complejidad (Grover & Pea, 2013).

Finalmente, una opción para conjuntar todo lo anterior es el aprendizaje basado en escenarios, un enfoque de enseñanza que implica presentar a los alumnos escenarios o situaciones realistas y contextualizados en los que deben aplicar sus conocimientos y habilidades para tomar decisiones y resolver problemas. Es un método centrado en el alumno que fomenta la participación activa, el pensamiento crítico y la aplicación del conocimiento en contextos prácticos.

Referencias

- Barr, Valerie & Stephenson, Chris. (2011). Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?. *ACM Inroads*, 2. 10.1145/1929887.1929905.
- Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*. Vancouver: American Educational Research Association., 1, 25.
- Dabbagh, N., & Kitsantas, A. (2012). Personal Learning Environments, social media, and self-regulated learning: A natural formula for connecting formal and informal learning. *The Internet and Higher Education*, 15(1), 3-8.
- Garrison, D. R., & Anderson, T. (2003). *E-learning in the 21st century: a framework for research and practice*. Routledge.
- Grover, S., & Pea, R. (2013). Computational Thinking in K–12: A Review of the State of the Field. *Educational Researcher*, 42(1), 38–43.
<https://doi.org/10.3102/0013189X12463051>
- Hmelo-Silver, C. E., Duncan, R. G., & Chinn, C. A. (2007). Scaffolding and achievement in problem-based and inquiry learning: A response to Kirschner, Sweller, and Clark (2006). *Educational Psychologist*, 42(2), 99-107.
- Kiili, K., Tuomi, P. (2019). Teaching Educational Game Design: Expanding the Game Design Mindset with Instructional Aspects. In: Liapis, A., Yannakakis, G., Gentile, M., Ninaus, M. (eds) *Games and Learning Alliance*. GALA 2019. *Lecture Notes in Computer Science()*, vol 11899. Springer, Cham. https://doi.org/10.1007/978-3-030-34350-7_11

Mayer, R. E. (2009). *Multimedia learning* (2nd ed.). Cambridge.

Nicolaidou, Iolie & Nicolaou, Charalambos & Constantinou, Costas. (2005). Implementing a Virtual Learning Environment to Enhance Modeling skills and Collaboration in a Pre-service Teacher Education Science Program: Students' Ideas about a Blended e-Learning Approach. 560-567.

Siemens, G., & Tittenberger, P. (2009). *Handbook of emerging technologies for learning*. University of Manitoba.

Weintrop, D., Beheshti, E., Horn, M. et al. Defining Computational Thinking for Mathematics and Science Classrooms. *J Sci Educ Technol* 25, 127–147 (2016).
<https://doi.org/10.1007/s10956-015-9581-5>

Wilensky, U., & Reisman, K. (2006) Thinking Like a Wolf, a Sheep, or a Firefly: Learning Biology Through Constructing and Testing Computational Theories—An Embodied Modeling Approach, *Cognition and Instruction*, 24:2, 171-209, DOI: 10.1207/s1532690xci2402_1

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3), 33–35.
<https://doi.org/10.1145/1118178.1118215>.