



La programación informática como herramienta para trabajar el álgebra en el aula

Fabienne Venant
Université du Québec à Montréal
Canada
venant.fabienne@uqam.ca

Resumen

El aprendizaje de la programación informática está en auge en las escuelas. En Quebec, por ejemplo, el plan de acción digital para la educación y la enseñanza superior publicado por el Gobierno en 2018 incluye el desarrollo del uso pedagógico de la programación informática en las escuelas como una de las medidas emblemáticas. En este taller, exploraremos si la escritura de programas informáticos puede contribuir al desarrollo de conocimientos algebraicos y de qué manera. Por eso, proponemos realizar y estudiar con los participantes una tarea de generación y resolución de ecuaciones en un contexto lúdico, utilizando la plataforma de programación Scratch. Utilizaremos el marco teórico de la génesis instrumental para analizar los tipos de trabajo implicados.

Palabras clave: Educación Matemática; Educación secundaria; Aprendizaje; Resolución de problemas; Herramientas tecnológicas; Informática; Algorítmica; Programación informática; Álgebra; Québec; Canadá.

Introducción

El aprendizaje de la programación informática está en auge en las escuelas. En Quebec, por ejemplo, el plan de acción digital para la educación y la enseñanza superior publicado por el Gobierno en 2018 incluye el desarrollo del uso pedagógico de la programación informática en las escuelas como una de las medidas emblemáticas. En un informe dirigido a proponer acciones óptimas para promover el uso pedagógico de la programación informática en las escuelas de Quebec, Barma et al. (2020) recomiendan que se añada la programación informática al plan de estudios de Quebec como una nueva habilidad transversal relacionada con el pensamiento

computacional. En este taller, queremos investigar la posibilidad de utilizar la programación informática como herramienta pedagógica y, por tanto, de ponerla al servicio del aprendizaje escolar. Nos interesa la especial relación entre la informática y, más concretamente, los algoritmos y las matemáticas escolares. Exploraremos si la escritura de programas informáticos puede contribuir al desarrollo de estos conocimientos y de qué manera. Nos centraremos especialmente en el álgebra y la resolución de ecuaciones. Nuestro objetivo es iniciar un debate para comprender mejor el trabajo algorítmico y sus vínculos con el trabajo matemático, en particular con el trabajo algebraico. Por eso, proponemos realizar y estudiar con los participantes una tarea de generación y resolución de ecuaciones en un contexto lúdico, utilizando la plataforma de programación Scratch.

Definición y relevancia del tema a desarrollar en el taller

Dualidad objeto/herramienta

Aún queda por explorar el papel preciso que puede desempeñar la programación informática en la movilización del conocimiento y el razonamiento matemáticos (Autor, 2022). Aunque la programación informática está estrechamente vinculada a las matemáticas a través de la actividad algorítmica que la sustenta, el algoritmo sigue siendo un objeto sobre el que se sabe poco. Es difícil prever su papel en la enseñanza y la formación de los profesores de matemáticas. El algoritmo o el programa informático que lo incorpora puede considerarse un método para resolver un problema matemático. Por tanto, se aborda como una herramienta. Sin embargo, como señalan Modeste et al. (2010), un algoritmo no es solo una herramienta para resolver problemas, sino también un objeto matemático por derecho propio. Por tanto, debe abordarse como un objeto de conocimiento, cuyas propiedades matemáticas deben estudiarse en relación con los demás conceptos algorítmicos que entraña. Esta dualidad objeto/herramienta, en el sentido de Douady (Douady, 1986), lleva a Modeste (2012) a defender la idea de que en matemáticas es fundamental no limitarse a cuestiones de programación, escritura y ejecución de algoritmos. Sino que hay que abordar el papel conceptual que los algoritmos pueden desempeñar en el trabajo matemático (Autor, 2018; Autor et al., 2023). Nuestro trabajo ha puesto de manifiesto la dificultad de construir el conocimiento matemático a partir de una situación que requiere conocimientos informáticos. La transposición necesaria para llevar a cabo una tarea algorítmica en clase de matemáticas puede no producirse espontáneamente. Un ajuste forzado o mal concebido puede ser contraproducente para la construcción de significados, tanto matemáticos como algorítmicos. Sumergir objetos matemáticos en un contexto algorítmico obliga a reinterpretarlos y a utilizarlos de nuevo a través de prácticas algorítmicas. Por ejemplo, aunque ambos tienen raíces comunes, las generalizaciones algebraicas y algorítmicas no adoptan la misma forma, sobre todo en lo que respecta a la gestión de la denominación.

Cuestiones didácticas difíciles de identificar.

Desentrañar las cuestiones didácticas específicas del trabajo algorítmico es particularmente crucial en contextos en los que se tiende a una visión puramente instrumental de la programación informática. Los objetos didácticos declarados en la interfaz de las dos disciplinas son difíciles de transferir en su doble referencia (Couderette, 2016). Nuestro trabajo (Autor, 2024) ha mostrado la necesidad de definir con precisión las intenciones didácticas, tanto matemáticas

como algorítmicas, a la hora de construir una tarea algorítmica para el aula de matemáticas. Tener en cuenta la dualidad objeto/herramienta del algoritmo es esencial para explorar un contexto matemático en el que se deben establecer esquemas algorítmicos. Sin embargo, la complejidad matemática puede dificultar el desarrollo algorítmico instrumental. Algunos algoritmos pueden ser más relevantes desde el punto de vista matemático, pero pueden resultar más difíciles de traducir al lenguaje informático. A la inversa, el desarrollo algorítmico avanzado puede bloquear el acceso a algoritmos matemáticos más eficientes, sobre todo en un contexto de prueba. La comparación de diferentes algoritmos que resuelven la misma cuestión y la depuración son actividades que deberían fomentarse explícitamente.

¿Instrumentos algorítmicos para trabajar con álgebra?

A menudo vemos (Barma, 2018; Laval, 2018) un deseo institucional de negar la importancia de la dualidad objeto/herramienta de los algoritmos y de reducir su enseñanza a una herramienta para resolver problemas o construir conocimiento matemático. Sin embargo, la enseñanza de los algoritmos conlleva muchas cuestiones didácticas, por lo que es imposible reducir un algoritmo a un artefacto que facilite la mediación entre los estudiantes y los conceptos matemáticos. Creemos que sería mejor promover la interacción entre las actividades algorítmicas y matemáticas. Esto nos lleva, en línea con Lagrange and Rogalski (2017), a cuestionar los conceptos y habilidades implicados en cada una de estas actividades y sus posibles correlaciones.

En este taller nos centraremos en el campo matemático del álgebra. Las palabras «álgebra» y «algoritmo» tienen una raíz lingüística común. De hecho, la palabra «álgebra» es una deformación occidental del nombre de Al-Khwārizmī, figura central en la formación de la escuela algebraica árabe en el siglo IX. La obra más conocida de Al-Khwārizmī es el Resumen sobre el cálculo por restauración y comparación (*Kitāb al-mukhtaṣar fī ḥisāb al-jabr wa-l-muqābala*), cuyo título, por cierto, dio origen a la palabra «álgebra» (*al-jabr*). En esta obra, Al-Khwārizmī organiza y sistematiza la solución de ecuaciones, en particular a partir de diferentes casos posibles. Los procedimientos que utilizaba y que más tarde fueron calificados de algebraicos tienen un fuerte aspecto algorítmico. Formulaban instrucciones «sencillas» que debían aplicarse sistemáticamente paso a paso, con el fin de transmitir métodos eficaces para obtener resultados precisos. Hoy en día, principalmente bajo la influencia de la informática, la finitud, la iteración, la recurrencia y la recursividad se han convertido en aspectos esenciales del pensamiento algorítmico. Como proceso de cálculo sistemático que puede aplicarse a diversos contextos (numérico, algebraico, geométrico, lógico, etc.), la algorítmica se distingue así más claramente del álgebra. Sin embargo, nos gustaría explorar cómo una actividad de programación puede explotar el potencial de estos estrechos vínculos entre algoritmos y álgebra.

Referencial teórico

Nos interesan las cuestiones didácticas implicadas en la integración de los algoritmos en la enseñanza de las matemáticas. Siguiendo a Lagrange y Rogalski (2017), esto nos lleva a cuestionar los conceptos y habilidades implicados en cada una de estas actividades y sus posibles correlaciones, adoptando la perspectiva teórica de la génesis instrumental (Rabardel, 1995). Este enfoque nos permite abordar el potencial de la programación informática, vista como un artefacto digital, para el aprendizaje (Artigue, 2002; Drijvers, 2000; Guin et al., 2006; Lagrange,

1999), pero también la complejidad del papel del profesor, que debe construir instrumentos didácticos que le permitan orquestar las diferentes génesis instrumentales de los alumnos (Artigue, 2007; Guin y Trouche, 2002; Haspekian, 2014; Restrepo, 2008)

El trabajo del alumno

Tratamos de comprender el trabajo del alumno y como se hace la transición de los objetos virtuales utilizados en clase a los conceptos a los que hacen referencia. Este trabajo se aborda como una actividad humana que adopta la forma de una actividad de resolución de problemas matemáticos. El alumno no debe limitarse a seguir los caminos trazados por los ejercicios escolares, sino actuar como un investigador que explora el problema que se le plantea. Su trabajo no consiste simplemente en adquirir conocimientos, sino en construirlos, y se basa en una cierta coherencia epistemológica (Kuzniak et al., 2016). Los significados también se derivan de los vínculos establecidos entre conceptos diferentes (Noss et al., 1997). El acto de observar relaciones y establecer conexiones, ya sea entre diferentes representaciones funcionales o entre dominios escolares, es un aspecto clave del proceso de aprendizaje (Boaler, 2022). Al igual que el del experto, el trabajo del alumno, cuando se centra en la resolución de problemas, implica fases de descubrimiento, explicación, justificación y aplicación (Giaquinto, 2005; Reichenbach, 1938).

En este taller, proponemos explorar el trabajo que se lleva a cabo durante la programación informática en un contexto algebraico. Para ello, consideramos tres tipos de trabajo.

Trabajo informático

Hablamos de trabajo informático cuando la actividad subyacente es la programación ergonómica, la animación o la contextualización. Este trabajo implica procesos (gestión de movimientos o sensores, consideración de las restricciones de funcionamiento.) y razonamientos (secuenciales, basados en eventos, disyuntivos...) de carácter instrumental o técnico. Conceptos instrumentales como eventos, testes, contadores definen formas particulares de concebir variables (como incrementadores, sensores o acumuladores) u operaciones (como interacciones, comportamientos...).

Trabajo algorítmico

Hablamos de trabajo algorítmico cuando se trata de una actividad sistemática que permite resolver un tipo de problemas siguiendo un conjunto predeterminado de reglas y en un número finito de pasos, hasta llegar a una respuesta en un momento dado (Modeste, 2012). Este trabajo implica procesos (generalización, identificación de regularidades, modelización...) y razonamientos (iterativos, por condiciones, por modularización...) de naturaleza algorítmica. Conceptos algorítmicos como la asignación o las estructuras de control definen formas particulares de entender la igualdad (como una verificación) o las variables (como cantidades computables en un proceso genérico).

Trabajo algebraico.

Hablamos de trabajo algebraico cuando la actividad subyacente implica operaciones de cualquier tipo que se repiten un número finito de veces (Squalli, 2002). Implica procesos (generalización, modelización, operaciones con símbolos...) y razonamientos (analíticos, funcionales, en términos de estructura...) de naturaleza algebraica. Los conceptos algebraicos, como las expresiones numéricas o la equivalencia, definen formas particulares de concebir la igualdad (como equivalencia), las operaciones (en suspenso) y las variables (como objetos conocidos) (Squalli y Jeannotte, 2024)

Existen similitudes interesantes entre el trabajo algebraico y el algorítmico, sobre todo en cuanto a la generalización y la identificación de estructuras o regularidades. Ambos tratan conceptos comunes (como la variable y la igualdad), pero desde enfoques diferentes. Por ello, nos parece interesante estudiar su relación en el contexto de una actividad de programación informática. Abordamos los distintos tipos de trabajo desde la perspectiva de su génesis instrumental, a través del diseño de un programa informático en una plataforma de programación.

Génesis instrumental

Desarrollado inicialmente por Rabardel (1995) desde una perspectiva de ergonomía cognitiva, el modelo instrumental responde a la necesidad de describir el funcionamiento de las herramientas y su relación con el conocimiento matemático movilizado (Soury-Lavergne, 2017). Esta teoría se basa en la distinción fundamental entre un artefacto, diseñado inicialmente para cumplir objetivos específicos, y el instrumento que un sujeto construye a partir de dicho artefacto durante su uso en una actividad situada. Por lo tanto, un artefacto es simplemente una proposición que un sujeto puede desarrollar o no durante una actividad (Trouche, 2005). La génesis instrumental se refiere al proceso mediante el cual un instrumento es construido por un sujeto que participa en una acción en la que interviene un artefacto. Esta construcción implica el establecimiento y desarrollo de esquemas de uso por parte del sujeto implicado en una acción, así como la mejora de las propiedades del artefacto. Se trata de dos niveles de esquemas: esquemas de uso y esquemas de acción instrumentada. Los esquemas de uso corresponden a las interacciones del individuo con el artefacto (Folcher y Rabardel, 2004). Se refieren a la gestión de las características y propiedades del artefacto y se aplican con relativa independencia de los objetivos de la actividad que se esté llevando a cabo. Los esquemas de acción instrumentada se orientan hacia el objeto de la actividad, es decir, la finalidad para la que se utiliza el artefacto. Incorporan patrones de uso.

La génesis instrumental implica dos procesos. La instrumentalización, orientada hacia el artefacto, que preside el descubrimiento o la adaptación de sus funcionalidades y la instrumentación, orientada hacia el sujeto, que preside la aparición y evolución de esquemas de uso. En nuestro caso, los bloques de instrucciones disponibles en la plataforma Scratch y los algoritmos que se pueden construir ensamblándolos son los artefactos en torno a los cuales se despliegan los instrumentos de trabajo matemático, algorítmico e informático. Los tipos de esquemas permiten caracterizar los tipos de trabajos.

Estrategia para desarrollar el taller

Tarea propuesta

Proponemos una reflexión colectiva sobre la relación entre los tres tipos de trabajo — algorítmico, informático y algebraico— a partir de la resolución de una tarea de programación informática utilizando la plataforma Scratch. Hemos elegido esta plataforma porque está muy extendida en las escuelas de Quebec. La tarea consiste en programar un juego de laberinto en el que el personaje «gato» se desplaza utilizando las teclas de flecha. A medida que lo hace, encuentra «tesoros» que debe ganar resolviendo una ecuación. El juego se detiene cuando el gato alcanza el botón verde situado en la parte inferior derecha del laberinto. Entonces, el programa comprueba que se han ganado todos los tesoros y muestra el tiempo que se ha tardado en completar el laberinto

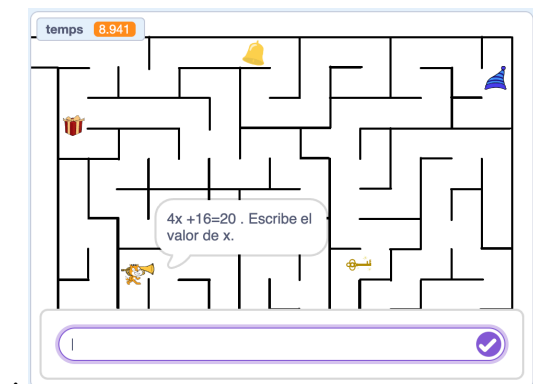


Figura 1. Laberinto algebraico

La tarea es compleja y puede dividirse en tres subtarefas: generación de ecuaciones, resolución de ecuaciones et gestión del laberinto.

Por la generación de ecuaciones, cada uno de los personajes que representan un tesoro debe generar una ecuación de primer grado que cumpla ciertas restricciones sobre el coeficiente, asegurarse de que tiene una solución entera, mostrar la ecuación y pedir la solución al jugador. Por la resolución de ecuación, cada uno de los personajes que representan un tesoro debe calcular la solución de la ecuación generada anteriormente y comparar el valor calculado con el valor introducido por el jugador. Estas dos primeras tareas las repite cada personaje, que genera una nueva ecuación tras cada intento fallido del jugador. La gestión del laberinto incluye la definición de los movimientos del personaje del gato utilizando los comandos de flecha y teniendo en cuenta la topología del laberinto (el gato no puede cruzar paredes...). En esa parte también se trata la gestión del tamaño de todos los personajes, de los sonidos que emiten y de los tiempos de espera. También concierne al número de respuestas correctas (o de tesoros ganados) y a la programación del cronómetro.

Dentro de estas subtarefas se encuentran los tres tipos de trabajo estudiados. El trabajo **informático** se refiere a los aspectos técnicos del diseño del juego. Esto implica, por ejemplo, la

gestión de eventos (como cuando se toca una flecha o un color), instrucciones condicionales y pruebas de igualdad (*si Tesor no es igual a 5*). El trabajo **algorítmico** apoya la implementación de estructuras iterativas más complejas. Incluye la elección y asignación de variables, la definición de las relaciones entre ellas, la determinación de los operadores pertinentes para cumplir las restricciones matemáticas y la elaboración de las pruebas de entrada y salida de las iteraciones. El trabajo **algebraico** permite determinar de forma genérica de las soluciones de las ecuaciones generadas. Se trata también de traducir las restricciones numéricas sobre el coeficiente y la solución de la ecuación que rigen la generación de las ecuaciones a una forma algebraica. Se basa también en la capacidad de generalizar la resolución de ecuaciones de primer grado. Implica un razonamiento analítico (Radford, 2014) que permite tratar las cantidades indeterminadas (coeficientes e incógnitas) como si fueran conocidas y realizar operaciones sobre estos números desconocidos.

Lo que resulta especialmente interesante es que el trabajo algorítmico y el trabajo matemático están estrechamente relacionados. La generalización y el razonamiento iterativo se retroalimentan mutuamente. En particular, esta vinculación implica una fase de denotación que permite asociar variables informáticas a la simbolización de números desconocidos.

Funcionamiento del taller

El facilitador presentará las cuestiones didácticas, los conceptos teóricos y los objetivos del taller durante aproximadamente 10 minutos. A continuación, presentará la tarea de programación que deberán realizar los participantes. Los participantes podrán trabajar individualmente o en equipo. Esta fase durará aproximadamente 50 minutos. A continuación, tendrá lugar una sesión de feedback en grupo. Los diferentes equipos debatirán sobre los problemas/obstáculos encontrados y las soluciones propuestas durante unos 10 minutos. A continuación, el animador presentará su propia versión del programa. Esta versión se podrá comparar con las soluciones propuestas por los participantes.

En la segunda fase de trabajo, los participantes identificarán, individualmente o en equipo, los tres tipos de trabajo que la tarea permite desarrollar. Esta fase se puede hacer descubriendo los esquemas de acción instrumentada, o de forma más intuitiva. A continuación, estos análisis se debatirán en el gran grupo. La atención se centrará en cómo los distintos tipos de trabajo (informático, algorítmico y algebraico) se complementan o se obstaculizan mutuamente. El taller finalizará con un resumen y una conclusión a cargo del responsable del taller.

Referencias y bibliografía

- Artigue, M. (2002). Learning mathematics in a CAS environment: The genesis of a reflection about instrumentation and the dialectics between technical and conceptual work. *International Journal of Computers for mathematical learning*, 7, 245-274. <https://doi.org/10.1023/A:1022103903080>
- Artigue, M. (2007). Digital technologies: a window on theoretical issues in mathematics education. *Proceedings of CERME*.
- Barma, S. (2018). *Réaliser une étude de cas multiple qui vise à affiner les connaissances sur l'usage pédagogique ou didactique de la programmation dans les écoles du Québec - Rapport final*. CRIRES. <https://lel.crires.ulaval.ca/oeuvre/rapport-final-realiser-une-etude-de-cas-multiple-qui-vise-affiner-les-connaissances-sur>

- Couderette, M. (2016). Enseignement de l'algorithmique en classe de seconde: une introduction curriculaire problématique. *Annales de Didactique et de Sciences Cognitives. Revue internationale de didactique des mathématiques*, 21, 267-296. <https://doi.org/10.4000/adsc.834>
- Douady, R. (1986). Jeux de cadre et dialectique objet-outil. *Recherche en didactique des mathématiques*, 7(2), 5-31. <https://revue-rdm.com/1986/jeux-de-cadres-et-dialectique/>
- Drijvers, P. (2000). Students encountering obstacles using a CAS. *International Journal of Computers for mathematical learning*, 5, 189-209. <https://doi.org/10.1023/A:1009825629417>
- Folcher, V. y Rabardel, P. (2004). Usages et appropriation : de l'analyse a priori à l'activité instrumentée. In P. Rabardel et P. Pastré (Eds), *Modèles du sujet pour la conception. Dialectiques Activités Développements* (pp. 129-155). Octarès Éditions. <https://doi.org/10.4000/activites.1885>
- Giaquinto, M. (2005). Mathematical activity. In n Paolo Mancosu, Klaus Froyen Jørgensen & S. A. Pedersen (eds.), *Visualization, explanation and reasoning styles in mathematics* (pp. 75-87). Springer.
- Guin, D., Ruthven, K., & Trouche, L. (2006). *The didactical challenge of symbolic calculators: turning a computational device into a mathematical instrument*. Springer Science & Business Media.
- Guin, D. y Trouche, L. (2002). *Calculatrices symboliques. Transformer un outil en un instrument du travail mathématique: un problème didactique*. La Pensée sauvage éditions.
- Haspekian, M. (2014). Teachers' instrumental geneses when integrating spreadsheet software. In: Clark-Wilson, A., Robutti, O., Sinclair, N. (Eds), *The Mathematics Teacher in the Digital Era. Mathematics Education in the Digital Era*, vol 2. Springer, Dordrecht. https://doi.org/10.1007/978-94-007-4638-1_11
- Kuzniak, A., Tanguay, D., & Elia, I. (2016). Mathematical Working Spaces in schooling: an introduction. *ZDM*, 48(6), 721-737. <https://doi.org/10.1007/s11858-016-0812-x>
- Lagrange, J.-B. (1999). Complex calculators in the classroom: Theoretical and practical reflections on teaching pre-calculus. *International Journal of Computers for mathematical learning*, 4, 51-81.
- Lagrange, J.-B., & Rogalski, J. (2017). Savoirs, concepts et situations dans les premiers apprentissages en programmation et en algorithmique. *Annales de Didactiques et de Sciences Cognitives*, 22, 119-158. <https://doi.org/10.4000/adsc.723>
- Laval, D. (2018). Une ingénierie didactique autour de "l'algorithme de dichotomie" illustrant un enseignement de l'algorithme en première scientifique comme objet d'apprentissage dans un champ des mathématiques: l'analyse. Actes du Séminaire Bational en Didactiques des Mathématiques.
- Modeste, S. (2012). *Enseigner l'algorithmique, pourquoi? Quels apports pour l'apprentissage de la preuve? Quelles nouvelles questions pour les mathématiques?* [Mémoire de thèse, Université de Grenoble].
- Modeste, S., Gravier, S., & Ouvrier-Buffer, C. (2010). Algorithmique et apprentissage de la preuve. *Repères Irem*, 79, 51-72.
- Noss, R., Healy, L., & Hoyles, C. (1997). The construction of mathematical meanings: Connecting the visual with the symbolic. *Educational Studies in Mathematics*, 33, 203-233. <https://doi.org/10.1023/A:1002943821419>
- Rabardel, P. (1995). *Les hommes et les technologies; approche cognitive des instruments contemporains*. Armand Colin.
- Radford, L. (2014). The progressive development of early embodied algebraic thinking. *Mathematics Education Research Journal*, 26, 257-277. <https://doi.org/10.1007/s13394-013-0087-2>
- Reichenbach, H. (1938). *Experience and prediction: An analysis of the foundations and the structure of knowledge*. Chicago, Ill.: The University of Chicago Press
- Restrepo, A. M. (2008). *Genèse instrumentale du déplacement en géométrie dynamique chez des élèves de 6ème* [Thèse de doctorat, Université Joseph-Fourier-Grenoble II].
- Squalli, H. (2002). *Une reconceptualisation du curriculum d'algèbre dans l'éducation de base*. [Thèse de doctorat, Université Laval].
- Squalli, H., y Jeannotte, D. (2024). Un modèle praxéologique de référence de la pensée algébrique élémentaire. *Revue québécoise de didactique des mathématiques*, 66-101.
- Soury-Lavergne, S. (2017). *Duos d'artefacts tangibles et numériques et objets connectés pour apprendre et faire apprendre les mathématiques*. [Mémoire d'HDR, École Normale Supérieure de Lyon].
- Trouche, L. (2005). *Des artefacts aux instruments, une approche pour guider et intégrer les usages des outils de calcul dans l'enseignement des mathématiques*. Actes de l'Université d'été, Le calcul sous toutes ses formes.